

**ROBUST DESIGN METHODOLOGY FOR COMMON CORE GAS
TURBINE ENGINES**

A Thesis
Presented to
The Academic Faculty

by

Jonathan Stephen Sands

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
Daniel Guggenheim School of Aerospace Engineering

Georgia Institute of Technology
May 2015

Copyright © 2015 by Jonathan Stephen Sands

ROBUST DESIGN METHODOLOGY FOR COMMON CORE GAS TURBINE ENGINES

Approved by:

Dr. Dimitri Mavris, Advisor
Daniel Guggenheim School of Aerospace
Engineering
Georgia Institute of Technology

Dr. Brian German
Daniel Guggenheim School of Aerospace
Engineering
Georgia Institute of Technology

Dr. Jeffrey Schutte
Daniel Guggenheim School of Aerospace
Engineering
Georgia Institute of Technology

Dr. Jechiel Jagoda
Daniel Guggenheim School of Aerospace
Engineering
Georgia Institute of Technology

Dr. Alexander Karl
Rolls-Royce Associate Fellow - Robust
Design and Systems Engineering
Rolls-Royce

Date Approved: April 3, 2015

To my greatest blessings, my wife and children. They keep me grounded, constantly reminding me of what is truly important in this life.

ACKNOWLEDGEMENTS

Looking back, I am very fortunate to have been given so many opportunities. In particular, my academic advisor, Professor Dimitri Mavris has provided me with so much. He serves as an outstanding role model, showing dedication to advancing the state of the art in our industry, while remaining humble, helpful, and approachable to all around him.

My thesis advisory committee has offered me valuable support and advice these past years, not only on my research, but in my personal development as a researcher and engineer. I am thankful to Professor Mavris and Professor Brian German for offering me guidance, keeping me focused on the important aspects of the design problem, and through their expertise drawing real world implications of design choices and their associated consequences. I'd also like to thank Dr. Jeffrey Schutte for all of his guidance throughout my thesis development, and also for his friendship. His technical assistance when battling in the trenches during model and method development was invaluable.

I'd also like to thank the members of my reading committee, both Professor Jechiel Jagoda and Dr. Alexander Karl of Rolls-Royce. Their broad range of experience made them great resources and I am very appreciative of the time they gave up and offered me during this process.

I am honored to have been a part of a great propulsion and energy division within ASDL at Georgia Tech. Dr. Jimmy Tai, Dr. Jeffrey Schutte, Mr. Russ Denney, and Mr. Chris Perullo have provided me great support over these past six years, along with all the prestigious students I've worked with. In particular, Mr. Metin Ozcan has been a huge help and a great friend. Always selfless in offering his assistance and producing great quality work, he set a terrific example for all of us at ASDL.

Finally, I can't thank my wife Carla enough for all of her unwavering love and support. I know she and our children are ready for the next adventure on our journey together.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iv
LIST OF TABLES	ix
LIST OF FIGURES	xiii
LIST OF SYMBOLS OR ABBREVIATIONS	xxii
SUMMARY	xxviii
I INTRODUCTION	1
1.1 Research Objective	2
1.2 Gas Turbine Engine Program Initial Development	5
1.3 Design for Maximum Utilization	9
II BACKGROUND	14
2.1 The CFM56 Engine Family	14
2.1.1 Initial Development - The CFM56-2	16
2.1.2 Boeing Application - The CFM56-3	17
2.1.3 Airbus Application - The CFM56-5	19
2.1.4 Boeing Next Generation - The CFM56-7 and Beyond	20
2.2 Gas Turbine Engine Design	21
2.2.1 Traditional Design Process	21
2.2.2 Development of Engine Variants	39
2.2.3 Development Cost Considerations	45
2.2.4 Probabilistic Design Methods	47
2.2.5 Enhanced Robust Design Simulation for Gas Turbine Engine Design	50
2.2.6 Bayesian Belief Network for Engine Upgrade Option Selection . . .	56
2.2.7 Multi-Objective Optimization of Engine Designs	57
III TECHNICAL APPROACH	61
3.1 Enhanced Robust Design Simulation (ERDS)	67
3.1.1 ERDS: Step-by-Step Procedure	76
3.1.2 ERDS Case Study: Robust Design of Two-Spool Axi-Centrifugal Turboshaft Engine	83

3.1.3	ERDS Summary	105
3.2	Modeling of Geometrically Common Core Variant Designs	105
3.2.1	Modeling and Simulation of Common Core Application	105
3.2.2	Geometrically Common Core Applications: Variant Design Rules	107
3.3	Integrated Common Engine Core Evaluation (COMMENCE) Method	110
3.3.1	Establishing the Target Market	111
3.3.2	Benchmark Engines for Applications Considered	113
3.3.3	Engine Variant Design Options and Restrictions	114
3.3.4	Common Core Variant Design Evaluation	120
3.3.5	Overall Engine Program Exploration and Evaluation	124
3.4	Summary of COMMENCE Method Development	127
IV	EXPERIMENTAL IMPLEMENTATION AND RESULTS	130
4.1	Training of Deterministic and Probabilistic Surrogate Models for Experiments	132
4.1.1	Step 1: Physics based model preparation	132
4.1.2	Step 2: Deterministic assessment and regression	137
4.1.3	Step 3: Probabilistic assessment and regression	142
4.1.4	Step 4: Robust design exploration and selection	144
4.2	Experiment 1: Enhanced Robust Design Simulation Method Applied to Single Application Turbofan Design Problem	147
4.2.1	Overview	147
4.2.2	Requirements and Technology Scenario	148
4.2.3	Experiment 1a: Deterministic Design Selection	150
4.2.4	Experiment 1b: Robust Design Probabilistic Assessment and Can- didate Design Selection	157
4.2.5	Conclusions	177
4.3	Experiment 2: Engine Variant Upgrade Option Exploration	179
4.3.1	Overview	180
4.3.2	Scenarios Considered	182
4.3.3	Common Core Engine Variant Design - Modeling and Simulation Environment	184
4.3.4	Common Core Technology Infusion	187
4.3.5	Utilization of Existing Surrogate Models	188

4.3.6	Variant Engine Design Space Exploration	189
4.3.7	Geometrically Fixed Core - Design Space Explorations for Various Technology Scenarios	191
4.3.8	Modified Common Core - Design Space Explorations for Various Technology Scenarios	197
4.3.9	Conclusions	206
4.4	Experiment 3: Common Engine Core Design for Multiple Applications . .	209
4.4.1	Utilization of Existing Surrogate Models	210
4.4.2	Multi-Application Common Core Design - Problem Setup	211
4.4.3	Common Core Engine Design for Multiple Applications	213
4.4.4	Conclusions	239
4.5	Experiment 4: Common Core Design for Multiple Applications Introduced Over Time	241
4.5.1	Sets of Requirements for Multiple Applications	243
4.5.2	Technology Scenarios for Multiple Time Periods	245
4.5.3	Common Core Engine Design for Multiple Applications at Various Time Periods	248
4.5.4	Conclusions	292
V	CONCLUSIONS	295
5.1	Research Questions and Hypotheses	296
5.2	Summary of Contributions Made	300
5.2.1	Possible Use Cases	302
5.2.2	Recommendation for Future Method Enhancement	303
5.2.3	Regarding Various Overall Engine Architectures	303
APPENDIX A — MODELING AND SIMULATION OF A GEOMETRI- CALLY COMMON CORE VARIANT ENGINE - NPSS CODE . . .		305
APPENDIX B — MONTE CARLO ANALYSIS FOR PROBABILISTIC SURROGATE MODEL TRAINING DATA GENERATION		330
APPENDIX C — DETERMINISTIC SURROGATE MODEL FIT STATIS- TICS		355
APPENDIX D — PROBABILISTIC SURROGATE MODEL FIT STATIS- TICS		371

APPENDIX E — EXPERIMENT 1C: ROBUST CYCLE AND TECHNOLOGY DESIGN	384
REFERENCES	391
VITA	400

LIST OF TABLES

1	Gas Turbine Engine Core Design Parameters	8
2	Engine Core Design Parameter Categories	13
3	Characteristics of the CFM56-2 and CFM56-3 Engines	18
4	Typical Commercial Transport Engine Design Requirements and Constraints	30
5	NASA Subsonic Transport Vehicle Level Metric Goals	58
6	Overall Objective and Corresponding Research Questions to be Addressed .	63
7	Beta distribution shape parameter lookup table based on domain of normalized modal value and confidence level of modal value.	69
8	ERDS Case Study - Shaft horsepower requirements at various flight conditions.	84
9	ERDS Case Study - Installation Uncertainty Distributions	89
10	ERDS Case Study - Component Efficiency Uncertainty Distributions	89
11	ERDS Case Study - Component Weight Uncertainty Distributions	89
12	ERDS Case Study - Miscellaneous Uncertainty Distributions	89
13	Design variables and corresponding ranges considered for ERDS case study.	91
14	Engine model output metrics for which surrogate models were trained as functions of control and noise variables.	91
15	Case Study - Comparison of robust and deterministic candidate cycle ranges (T41=3,450°R).	101
16	Engine Variant Design Options - Matrix of Alternatives.	116
17	NPSS model components and their corresponding DESIGN or OFF-DESIGN status during Design and Variant engine simulations.	118
18	Typical control variables and corresponding ranges for a clean sheet turbofan engine design exploration.	126
19	Control variables and corresponding ranges for variant engine exploration. .	126
20	Core Defining Design Engine - Design Variable Ranges	133
21	Core Defining Design Engine - Input Variables for Component Efficiency Impacts	134
22	Core Defining Design Engine - Input Variables for Component Weight Factors	134
23	Core Defining Design Engine - Remaining Input Variables with Possible Uncertainty	135
24	Common Core Variant Engine - Design Variable Ranges	135

25	Common Core Variant Engine - Input Variables for Component Efficiency Impacts	136
26	Common Core Variant Engine - Input Variables for Component Weight Factors	136
27	Common Core Variant Engine - Remaining Input Variables with Possible Uncertainty	137
28	Experiment 1 - Engine Design Points	149
29	Experiment 1 - Engine Design Requirements	149
30	Experiment 1 - Baseline Cycle, Technology Level, and Corresponding Performance	150
31	Experiment 1 - Design Variable Ranges	150
32	Experiment 1 - Most likely engine technology improvements with respect to baseline engine.	151
33	Experiment 1 - Temperature limits imposed on candidate designs.	151
34	Experiment 1a - Candidate design selection with top TOPSIS score that maintained feasible temperature levels within imposed limits.	157
35	Experiment 1 - Technology Impact Distributions - Component Efficiency . .	159
36	Experiment 1 - Technology Impact Distributions - Component Weight . . .	160
37	Experiment 1 - Technology Impact Distributions - Installations	161
38	Experiment 1 - Technology Impact Distributions - Component Loading . .	162
39	Experiment 1 - Technology Impact Distributions - Component Cooling . . .	162
40	Experiment 1 favorable design regions that provided top scoring engines for the deterministic assessment, probabilistic assessment with 50% CI metric estimates, and probabilistic assessment with 95% CI metric estimates. . . .	174
41	Experiment 1b Candidate design selections from probabilistic assessment with top 95% likely TOPSIS score that maintained a 95 % likelihood of having feasible temperature levels within imposed limits, compared to Experiment 1a Deterministic Selection. Prob Select 1 within original temperature limits, Prob Select 2 within relaxed temperature limits.	175
42	Experiment 2 - Variant engine upgrade scenarios.	181
43	Experiment 2 - Baseline cycle used to size the common engine core.	183
44	Experiment 2 - Major NPSS model components and their DESIGN or OFF-DESIGN status during Design and Variant engine simulations.	186
45	Experiment 2 - Low pressure system technology infusion - scenario impacts on variant design.	188
46	Experiment 2 - High pressure system technology infusion - scenario impacts on variant design.	189

47	Experiment 2 - Geometrically Fixed Core Scenarios - Feasible design selections achieving maximum takeoff thrust growth.	196
48	Experiment 2 - Modified Core Scenarios - Feasible design selections achieving maximum takeoff thrust growth.	200
49	Experiment 3 - Engine Requirements and cycle temperature limits for each of the three applications considered in the present common core engine design problem.	212
50	Experiment 3 - Power and air extraction requirements uncertainty distributions assumed for the present multiple application experiment.	213
51	Experiment 3 - Technology impact uncertainty distributions assumed for the present technology scenario.	215
52	Experiment 3 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for each of the three applications considered.	217
53	Experiment 3 - Common Core Variant Engine Cycle Selections and corresponding 95% confidence interval performance levels for each of the three applications considered.	237
54	Experiment 3 - Common core variant and overall program evaluation based on comparisons to new centerline benchmark cycles independently designed for each application considered.	238
55	Experiment 4 - Engine Applications considered for the present multiple application common core design problem.	243
56	Experiment 4 - Application Installation Requirements Uncertainty Distributions assumed for each of the seven Common Core Variant Applications considered in the present experiment.	246
57	Experiment 4 - Technology Impact Uncertainty Distributions assumed for the three technology levels considered in the present multiple application experiment.	247
58	Experiment 4 - HPC exit temperature and LPT inlet temperature limits imposed on each of the seven engine applications considered.	248
59	Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the low technology level designs: Applications A and B.	250
60	Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the mid technology level designs: Applications C and D.	250
61	Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the high technology level designs: Applications E, F, and G.	251

62	Experiment 4 - Core Defining Engine Cycle Selection.	260
63	Experiment 4 - Core Performance Mitigation - Adjustments made to imposed limits of core design limiting applications in order to improve overall engine program performance.	263
64	Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for Applications A and B after Core Performance Mitigation.	264
65	Experiment 4 - Common Core Design Cycle Selections before and after Core Performance Mitigation.	267
66	Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications A and B.	275
67	Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications C and D.	280
68	Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications E, F, and G.	286
69	Experiment 4 - Initial-term common core variant design responses compared to each corresponding benchmark engine cycle selection.	288
70	Experiment 4 - Mid-term common core variant design responses compared to each corresponding benchmark engine cycle selection.	289
71	Experiment 4 - Late-term common core variant design responses compared to each corresponding benchmark engine cycle selection.	291
72	Experiment 4 - Common core variant and overall program evaluation based on comparisons to new centerline benchmark cycles independently designed for each application considered.	293
73	Experiment 1c - Baseline and Pushed Efficiency Improvement Distributions	385
74	Experiment 1c - Component efficiency improvement distribution shape parameter settings for each of the technology scenarios considered.	386
75	Experiment 1c - Highest scoring candidate cycle designs for each technology scenario considered and corresponding 95% likely performance levels.	390

LIST OF FIGURES

1	Standard Turbofan Engine Flow Station Designations	7
2	CFM International’s CFM56-2 Engine	15
3	CFM56 Engine Family Thrust Capabilities	22
4	CFM56 Engine Family Evolution	23
5	Oates engine/airframe matching process.	25
6	Mattingly’s Gas Turbine Engine Design Process	27
7	Mattingly’s Preliminary Engine Design Process	29
8	Impact of BPR on Specific Thrust	33
9	Impact of BPR on the Thrust per unit Core Air Flow	34
10	Engine Program Commonality Levels	39
11	Product variety tradeoff chart	41
12	Performance deviations due to commonality	41
13	Simpson’s NCI and PDI measurements visualized on Fellini’s chart of performance variation across design settings for various design applications. . .	42
14	Core size requirements for multiple types of engine applications	43
15	Common Core Flowpath Variety	44
16	Engine Family Selection Process	45
17	Probability Density Function	49
18	Cumulative Distribution Function	49
19	Generic form of an Artificial Neural Network (ANN) surrogate model. . . .	53
20	Pareto Front and Corresponding TOPSIS Measurements	59
21	Common Engine Core Evaluation (COMMENCE) Method.	62
22	Identification of required advancements to be made and integrated into the overall COMMENCE process.	65
23	Integrated Product and Process Development decision-making process . . .	66
24	Progression of probability density function for impact of technology being matured.	70
25	Robust Design Simulation method modified for robust engine and technology design.	71
26	Cumulative distribution function for likely performance of a candidate design. 73	73

27	Monte Carlo probabilistic analysis process.	74
28	Two spool axi-centrifugal turboshaft engine architecture.	83
29	Centrifugal compressor tip speed constraints due to material stress limits.	86
30	Axi-centrif engine performance limited by material stress limits.	88
31	Artificial Neural Network surrogate model fit quality - cruise ESFC.	92
32	Probabilistic surrogate model fit quality - likely cruise ESFC.	95
33	Feasible design centrifugal compressor pressure ratio settings for various max turbine inlet temperatures.	96
34	Feasible centrifugal compressor design pressure ratio settings for various design power levels.	97
35	Feasible centrifugal compressor specific speed settings for various design compressor pressure ratios.	97
36	Feasible centrifugal compressor specific speed settings for various design overall pressure ratios.	98
37	Feasible design overall pressure ratio settings for various design centrifugal compressor pressure ratios.	99
38	95% likely ESFC levels for various centrifugal compressor pressure ratio designs.	99
39	95% likely engine weight levels for various centrifugal compressor pressure ratio designs.	100
40	Evaluation of robust and deterministic candidate cycles against technology realization.	104
41	Common Core Variant Engine - Modeling and Simulation Environment Representation	106
42	Common Core Variant Engine - Design Map Operating Point and Defining Parameters	108
43	COMMENCE Method - Establishing Market Placement	112
44	COMMENCE Method - Benchmark Engine Generation for Each Application Considered	114
45	COMMENCE Method - Engine Variant Design Options	116
46	NPSS model components making up the physics based separate flow turbofan engine model, highlighting the engine core which remains geometrically common between design and variant engine applications.	118
47	Fit quality achieved for deterministic surrogate model predicting TSFC at ADP for the core defining design engine.	140
48	Fit quality achieved for deterministic surrogate model predicting minimum cruise TSFC for the common core variant engine.	141

49	Fit quality achieved for probabilistic surrogate model predicting HPC exit corrected flow at ADP for the core defining design engine.	145
50	Fit quality achieved for probabilistic surrogate model predicting minimum cruise TSFC for the common core variant engine.	145
51	High bypass, two shaft boosted turbofan engine architecture.	148
52	Experiment 1a - Constrained scatterplot of deterministic simulation data. .	153
53	Experiment 1a - Constrained scatterplot of deterministic simulation data, colored by TOPSIS score.	155
54	Experiment 1a - Constrained parallel coordinate chart of deterministic assessment with highest TOPSIS scores.	156
55	Experiment 1b - Constrained scatterplot of 50% likely performance data. .	164
56	Experiment 1b - Constrained scatterplot of 95% likely performance data. .	166
57	Experiment 1b - Constrained scatterplot of 95% likely performance data, HPC exit temperature limit increased by 50 degrees.	167
58	Experiment 1b - Constrained scatterplot of 95% likely performance data, LPT inlet temperature limit increased by 50 degrees.	168
59	Experiment 1b - Constrained scatterplot of 50% likely performance data, colored by TOPSIS score.	170
60	Experiment 1b - Constrained scatterplot of 95% likely performance data, colored by TOPSIS score, HPC exit and LPT inlet limits both increased by 50 degrees.	171
61	Experiment 1b - Constrained parallel coordinate chart of 50% likely performance data with highest TOPSIS scores.	173
62	Experiment 1b - Constrained parallel coordinate chart of 95% likely performance data with highest TOPSIS scores, HPC exit and LPT inlet limits both increased by 50 degrees.	173
63	NPSS model components making up the physics based separate flow turbofan engine model. Gray components make up the engine core which remains geometrically common between design and variant engine applications. . . .	185
64	Experiment 2 - Common core modeling and simulation environment representation.	186
65	Scenario 2.1 multivariate scatterplot of feasible candidate variant engine designs.	193
66	Scenario 2.2 multivariate scatterplot of feasible candidate variant engine designs.	194
67	Scenario 2.3 multivariate scatterplot of feasible candidate variant engine designs.	195

68	Scenario 2.4 multivariate scatterplot of feasible candidate variant engine designs.	198
69	Scenario 2.5 multivariate scatterplot of feasible candidate variant engine designs.	199
70	Scenario 2.3 multivariate scatterplot of feasible candidate variant engine designs. High technology variant engine design space with fixed geometry core.	202
71	Scenario 2.5 multivariate scatterplot of feasible candidate variant engine designs. High technology variant engine design space with flared HPC core. . .	203
72	Scenario 2.3 parallel coordinate plot of feasible candidate variant engine designs. High technology variant engine design space with fixed geometry core.	204
73	Scenario 2.5 parallel coordinate plot of feasible candidate variant engine designs. High technology variant engine design space with flared HPC core. . .	205
74	Scenario 2.3 parallel coordinate plot of best candidate variant engine designs. High technology variant engine design space with fixed geometry core. . . .	205
75	Scenario 2.5 parallel coordinate plot of best candidate variant engine designs. High technology variant engine design space with flared HPC core.	206
76	Scenario 2.3 parallel coordinate plot of best candidate variant engine cycles. High technology variant engine design space with fixed geometry core. . . .	207
77	Scenario 2.5 parallel coordinate plot of best candidate variant engine cycles. High technology variant engine design space with flared HPC core.	207
78	Representation of multiple application common core design problem considered in Experiment 3. Each engine variant performance estimate is dependent upon that variant’s input variable settings as well as the design engine input settings which establish the common engine core definition.	220
79	Simplified representation of multiple application common core design problem considered in Experiment 3.	221
80	Experiment 3 - Common core design candidates that meet thrust requirements of all three applications while also satisfying all temperature constraints.	224
81	Experiment 3 - Feasible common core design candidates and corresponding cruise TSFC levels for each engine variant application.. . . .	225
82	Experiment 3 - Feasible common core design candidates which achieve high performance among the samples collected within the common core application design spaces.	226
83	Experiment 3, Application A - Feasible common core variant engine cycle candidates among the sample population.	227
84	Experiment 3, Application B - Feasible common core variant engine cycle candidates among the sample population.	227

85	Experiment 3, Application C - Feasible common core variant engine cycle candidates among the sample population.	228
86	Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine takeoff thrust and HPC design pressure ratio after simultaneous exploration of core and variant engines.	230
87	Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine top-of-climb thrust and HPC design pressure ratio after simultaneous exploration of core and variant engines.	230
88	Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio after simultaneous exploration of core and variant engines.	231
89	Experiment 3 - Side-by-side comparison of contour profilers representing several core design planes.	232
90	Experiment 3 - Common core variant exploration for Application A: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant A feasibility.	233
91	Experiment 3 - Common core variant exploration for Application B: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant B feasibility.	235
92	Experiment 3 - Common core variant exploration for Application C: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant C feasibility.	236
93	Experiment 4 - Common Core Variant Representations, Thrust Requirements and Maximum Fan Diameter Limits, each application colored by its corresponding technology level assumed.	244
94	Experiment 4 - Representation of the multiple application, multiple technology level common core design problem, and the dependencies present in the problem.	255
95	Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine takeoff thrust and HPC design pressure ratio during simultaneous exploration of core and variant engines.	256
96	Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine top-of-climb thrust and HPC design pressure ratio during simultaneous exploration of core and variant engines.	258
97	Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio during simultaneous exploration of core and variant engines.	258

98	Experiment 4 - Common core design exploration: Side-by-Side contour profilers identifying the feasible common core design space across the key contributing design planes of the core defining design engine.	259
99	Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio during simultaneous exploration of core and variant engines, labeled to identify contours of constant HPC exit corrected flow and the favorable design region that would allow for a more compact common core definition.	262
100	Experiment 4 - Common core design exploration: Constraint contours and design core selection settings before and after Core Performance Mitigation.	266
101	Surface plot showing the exponential decay of the likelihood of a multiple application sample being feasible as the number of applications considered increases for various levels of the probability of a single application being feasible.	269
102	Experiment 4 - Common Core, Variant A Design Exploration: Constraint contours and Variant A design selection settings before and after Core Performance Mitigation.	272
103	Experiment 4 - Common Core, Variant B Design Exploration: Constraint contours and Variant B design selection settings before and after Core Performance Mitigation.	273
104	Experiment 4 - Common Core, Variant C Design Exploration: Constraint contours and Variant C design selection settings before and after Core Performance Mitigation.	277
105	Experiment 4 - Common Core, Variant D Design Exploration: Constraint contours and Variant D design selection settings before and after Core Performance Mitigation.	278
106	Experiment 4 - Common Core, Variant E Design Exploration: Constraint contours and Variant E design selection settings before and after Core Performance Mitigation.	281
107	Experiment 4 - Common Core, Variant F Design Exploration: Constraint contours and Variant F design selection settings before and after Core Performance Mitigation.	283
108	Experiment 4 - Common Core, Variant G Design Exploration: Constraint contours and Variant G design selection settings before and after Core Performance Mitigation.	284
109	Core Defining Engine - Deterministic model fit statistics - LPC inlet corrected flow at ADP.	358
110	Core Defining Engine - Deterministic model fit statistics - LPC inlet corrected flow at TKO.	359

111	Core Defining Engine - Deterministic model fit statistics - HPC exit corrected flow at ADP.	359
112	Core Defining Engine - Deterministic model fit statistics - HPC exit total temperature (T3) at TKO.	360
113	Core Defining Engine - Deterministic model fit statistics - LPT inlet total temperature (T45) at TKO.	360
114	Core Defining Engine - Deterministic model fit statistics - HP shaft power at ADP.	361
115	Core Defining Engine - Deterministic model fit statistics - HP shaft power at TKO.	361
116	Core Defining Engine - Deterministic model fit statistics - Bypass ratio at ADP.	362
117	Core Defining Engine - Deterministic model fit statistics - Fan diameter. . .	362
118	Core Defining Engine - Deterministic model fit statistics - TSFC at ADP. .	363
119	Core Defining Engine - Deterministic model fit statistics - Engine pod weight.	363
120	Common Core Variant Engine - Deterministic model fit statistics - LPC inlet corrected flow at ADP.	364
121	Common Core Variant Engine - Deterministic model fit statistics - LPC inlet corrected flow at TKO.	365
122	Common Core Variant Engine - Deterministic model fit statistics - HPC exit total temperature (T3) at TKO.	365
123	Common Core Variant Engine - Deterministic model fit statistics - LPT inlet total temperature (T45) at TKO.	366
124	Common Core Variant Engine - Deterministic model fit statistics - Net thrust at TKO.	366
125	Common Core Variant Engine - Deterministic model fit statistics - Net thrust at TOC.	367
126	Common Core Variant Engine - Deterministic model fit statistics - HP shaft power at ADP.	367
127	Common Core Variant Engine - Deterministic model fit statistics - HP shaft power at TKO.	368
128	Common Core Variant Engine - Deterministic model fit statistics - Bypass ratio at ADP.	368
129	Common Core Variant Engine - Deterministic model fit statistics - Fan diameter.	369
130	Common Core Variant Engine - Deterministic model fit statistics - TSFC at ADP.	369

131	Common Core Variant Engine - Deterministic model fit statistics - Minimum TSFC at CRZ.	370
132	Common Core Variant Engine - Deterministic model fit statistics - Engine pod weight.	370
133	Core Defining Engine - Probabilistic model fit statistics - LPC inlet corrected flow at ADP.	372
134	Core Defining Engine - Probabilistic model fit statistics - LPC inlet corrected flow at TKO.	373
135	Core Defining Engine - Probabilistic model fit statistics - HPC exit corrected flow at ADP.	373
136	Core Defining Engine - Probabilistic model fit statistics - HPC exit total temperature (T3) at TKO.	374
137	Core Defining Engine - Probabilistic model fit statistics - LPT inlet total temperature (T45) at TKO.	374
138	Core Defining Engine - Probabilistic model fit statistics - HP shaft power at ADP.	375
139	Core Defining Engine - Probabilistic model fit statistics - HP shaft power at TKO.	375
140	Core Defining Engine - Probabilistic model fit statistics - Bypass ratio at ADP.	376
141	Core Defining Engine - Probabilistic model fit statistics - Fan diameter. . .	376
142	Common Core Variant Engine - Probabilistic model fit statistics - LPC inlet corrected flow at ADP.	377
143	Common Core Variant Engine - Probabilistic model fit statistics - LPC inlet corrected flow at TKO.	378
144	Common Core Variant Engine - Probabilistic model fit statistics - HPC exit total temperature (T3) at TKO.	378
145	Common Core Variant Engine - Probabilistic model fit statistics - LPT inlet total temperature (T45) at TKO.	379
146	Common Core Variant Engine - Probabilistic model fit statistics - Net thrust at TKO.	379
147	Common Core Variant Engine - Probabilistic model fit statistics - Net thrust at TOC.	380
148	Common Core Variant Engine - Probabilistic model fit statistics - HP shaft power at ADP.	380
149	Common Core Variant Engine - Probabilistic model fit statistics - HP shaft power at TKO.	381

150	Common Core Variant Engine - Probabilistic model fit statistics - Bypass ratio at ADP.	381
151	Common Core Variant Engine - Probabilistic model fit statistics - Fan diameter.	382
152	Common Core Variant Engine - Probabilistic model fit statistics - Minimum TSFC at CRZ.	382
153	Common Core Variant Engine - Probabilistic model fit statistics - Engine pod weight.	383
154	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, base efficiency distributions.	387
155	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, fan efficiency distribution push.	388
156	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, LPC efficiency distribution push.	388
157	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, HPC efficiency distribution push.	388
158	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, HPT efficiency distribution push.	389
159	Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, LPT efficiency distribution push.	389

LIST OF SYMBOLS OR ABBREVIATIONS

A_{25}	Flow area at the entrance of the high pressure compressor.
A_3	Flow area at the exit of the high pressure compressor.
A_4	Flow area at the exit of the combustor.
A_{41}	Flow area at the entrance of the high pressure turbine.
A_{43}	Flow area at the exit of the high pressure turbine.
ADP	Aerodynamic Design Point, the engine operating conditions at which engine components are designed for maximum efficiency. Typically defined at cruise conditions and engine maximum power.
AN^2	Turbine loading parameter, usually measured in <i>billions of in² · rpm²</i> .
ANN	Artificial Neural Network.
BBN	Bayesian Belief Network.
BPR	Bypass Ratio, the ratio of the amount of airflow traveling through the engine's bypass duct to the amount of airflow entering the engine's core compression system.
CDF	Cumulative Distribution Function.
CDS	Cycle Design Space.
CFMI	CFM International, a joint partnership of Snecma of France and General Electric of the United States.
Ch_{gHPT}	HPT Chargeable Cooling Flow, expressed as a fraction of total HPC inflow.
Ch_{gLPT}	LPT Chargeable Cooling Flow, expressed as a fraction of total HPC inflow.
COMMENCE	COMMMon ENgine Core Evaluation.
CRZ	Cruise conditions.
CVs	Control Variables. Within the Enhanced Robust Design Simulation process, the set of variables for which settings will be selected that result in a technology scenario and robust design selection for the particular technology scenario considered.
ΔH	Total enthalpy rise.
DoE	Design of Experiments.
\dot{E}_{fuel}	Rate of thermal energy addition by the combustor.

EPR	Engine Pressure Ratio.
$\dot{E}_{propulsive}$	Rate of energy addition by the engine.
ER	Extraction Ratio, the ratio of total pressure exiting the bypass nozzle to the total pressure exiting the core nozzle.
ERA	Environmentally Responsible Aviation, a NASA project aiming to simultaneously reduce vehicle fuel burn, noise, and NOx emissions within the N+2 time frame.
ERDS	Enhanced Robust Design Simulation.
ER_{Var}	Extraction Ratio of the variant engine.
ESFC	Effective Specific Fuel Consumption, usually measured in units of $lb_{m, fuel} / (eshp \cdot hr)$.
η_{Fan}	Fan efficiency achievable at aerodynamic design point.
η_{HPC}	High pressure compressor efficiency achievable at aerodynamic design point.
η_{HPT}	High pressure turbine efficiency achievable at aerodynamic design point.
η_{LPC}	Low pressure compressor efficiency achievable at aerodynamic design point.
η_{LPT}	Low pressure turbine efficiency achievable at aerodynamic design point.
η_O	Overall efficiency.
η_P	Propulsive efficiency.
η_T	Thermal efficiency.
η_{Tr}	Transmission efficiency.
FADEC	Full Authority Digital Electronics Control.
F_N	Net thrust produced by the engine, expressed in lbf .
FPT	Free Power Turbine, responsible for providing the power required by the turboshaft engine.
GE	General Electric.
HPC	High Pressure Compressor, the compressor stages that experience the highest temperatures in the compression system. Powered by the high pressure turbine.
HPT	High Pressure Turbine, the turbine stages providing power to the high pressure compressor stages.

JPDM	Joint Probabilistic Decision-Making Method.
$\dot{K}E_{core}$	Total available kinetic energy addition rate of the engine core.
$\dot{K}E_{engine}$	Total available kinetic energy addition rate of the engine.
LHS	Latin Hypercube Sampling.
LPC	Low Pressure Compressor, the compressor stages spinning at the same rotational speed as the fan and powered by the low pressure turbine.
LPT	Low Pressure Turbine, the turbine stages providing power to the low pressure compression system.
MDP	Multiple Design Point.
\dot{m}_f	Fuel mass flow rate.
MFE	Model Fit Error, the error present in a surrogate model when predicting the performance of data used in the training of the model.
MRE	Model Representation Error, the error present in a surrogate model when predicting the performance of data not used in the training of the model.
N	Physical rotational speed of turbomachinery, usually measured in rpm.
N_C	Corrected rotational speed of the high speed turbomachinery shaft powering the engine core, usually measured in rpm.
$NChg_{HPT}$	HPT NonChargeable Cooling Flow, expressed as a fraction of total HPC inflow.
$NChg_{LPT}$	LPT NonChargeable Cooling Flow, supplied by HPC interstage bleed and expressed as a fraction of total HPC inflow.
$NChg_{LPT,Var}$	LPT NonChargeable Cooling Flow of the variant engine, supplied by HPC interstage bleed and expressed as a fraction of total HPC inflow.
$N_{C,HPT,Var}$	Corrected map speed of the variant engine's high pressure turbine.
NCI	Non-Commonality Index.
N_{FPT}	Design mechanical speed of free power turbine.
NOx	Nitrous Oxide, harmful emissions that engine designers try to minimize with advanced combustors.
NPSS	Numerical Propulsion System Simulation, an aerothermodynamic engine cycle simulation tool originally developed by NASA.
$N_{s,HPC}$	Specific speed of high pressure centrifugal compressor.

$nStg_{HPC}$	Number of rotor stages that make up the high pressure compressor.
$nStg_{HPT}$	Number of rotor stages that make up the high pressure turbine.
NVs	Noise Variables. Within the Enhanced Robust Design Simulation process, the set of variables with which variability due to uncertainty is assumed present, represented by a distribution of likely variable settings.
PDF	Probability Density Function.
PDI	Performance Deviation Index.
π_{Fan}	Fan Pressure Ratio, the ratio of the total pressure exiting an engine fan stage to the total pressure entering the fan.
$\pi_{Fan,Var}$	Fan Pressure Ratio of the variant engine.
π_{HPC}	High Pressure Compressor Pressure Ratio.
$\pi_{Overall}$	Overall Pressure Ratio, the ratio of the total pressure exiting an engine's high pressure compressor to the total pressure entering the first compression element of the engine.
$\pi_{Overall,Var}$	Overall Pressure Ratio of the variant engine.
$P_{S,core}$	Specific core power.
Q	Volumetric flow rate.
R^2	Coefficient of determination, a statistical measure of how well a model explains the variation present in a metric response.
RDS	Robust Design Simulation.
SFC	Specific Fuel Consumption, general term used interchangeably with either thrust-specific or power-specific fuel consumption.
s_{HP}	Design horsepower growth margin.
SLS	Sea-level Static conditions.
$SSA_{Des,Det}$	Surrogate Set A, the deterministic surrogate models trained to represent the core defining design engine of the design-variant physics based modeling and simulation environment. This set of surrogate models can also be used for the deterministic design exploration of a single engine design problem.
$SSB_{Var,Det}$	Surrogate Set B, the deterministic surrogate models trained to represent the common core variant engine of the design-variant physics based modeling and simulation environment.

$SSC_{Des,Prob}$	Surrogate Set C, the probabilistic surrogate models trained to represent the core defining design engine of the design-variant physics based modeling and simulation environment. This set of surrogate models can also be used for the probabilistic design exploration of a single engine design problem.
$SSD_{Var,Prob}$	Surrogate Set D, the probabilistic surrogate models trained to represent the common core variant engine of the design-variant physics based modeling and simulation environment.
SVM	Support Vector Machine, a kernel based classification model.
sW_{25R}	Variant design core flow scale factor, the factor by which the design core airflow of the variant engine is increased with respect to the design engine core flow.
$T_{3,max}$	Maximum allowable stagnation temperature at engine flow station 3, the exit of the high pressure compressor.
T_4	Gas stagnation temperature at engine flow station 4, the exit of the combustor.
T_{41}	Gas stagnation temperature at engine flow station 41, the entrance of the high pressure turbine. The gas at this flow station includes both combustor exit gas and non-chargeable cooling gas.
$T_{41,max}$	Maximum allowable stagnation temperature of gas entering the high pressure turbine.
T_{45}	Stagnation temperature of gas entering the low pressure turbine.
$T_{45,max}$	Maximum allowable stagnation temperature of gas entering the low pressure turbine.
$T_{4,max}$	Maximum allowable stagnation temperature of gas exiting the combustor.
$T_{4,Var}$	Gas stagnation temperature at engine flow station 4, the exit of the variant engine combustor.
TIT	Turbine Inlet Temperature, the temperature of the gas entering the high pressure turbine after exiting the engine's combustor.
TKO	Takoeff conditions.
TOC	Top of Climb conditions.
TOPSIS	Technique for Order Preference by Similarity to Ideal Solution.
TSFC	Thrust Specific Fuel Consumption, usually measured in units of $lb_{m,fuel}/(lb_{f,thrust} \cdot hr)$.
VAATE	Versatile Affordable Advanced Turbine Engines.

W_{23R}	Corrected flow entering core, the corrected amount of air flow at the entrance of the low pressure compressor (booster).
W_{25}	Core airflow, the amount of air flow traveling through the entrance of the high pressure compressor.
W_{25R}	Corrected core flow, the corrected amount of air flow at the entrance of the high pressure compressor.
W_{3R}	HPC exit corrected flow. Used as a geometric similarity parameter for comparison of gas turbine engine cores.
WATE	Weight Analysis of Turbine Engines, a NASA tool developed for gas turbine weight and disk life estimationan.

SUMMARY

The initial development of a clean sheet gas turbine engine program is a multi-billion dollar undertaking[80]. In order to distribute this immense development cost among as large a number of products as possible, a family of engines should employ the resultant engine core design across a variety of engine applications. However, initial common core design selections made during a program's initial development phase can limit both its utility for future variant applications as well as the resultant application performance levels.

A method has been developed to simultaneously consider immediate and future common core engine variant applications when exploring and evaluating candidate gas turbine engine core designs, while minimizing the added computational burden on the designer. The Common Engine Core Evaluation (COMMENCE) method serves to allow core designers to consider any number of initial or future engine applications, each under unique uncertainty scenarios, enabling simultaneous exploration of the multiple application common core design space. By estimating the implications of core design decisions on current and future applications, the method aims to increase the likelihood of the designer identifying an engine core definition and technology development strategy that offers feasible common core variant solutions able to achieve competitive performance levels for the engine applications considered.

This work assumes that for an engine core to be geometrically common, its core casing geometry and flow path must remain fixed when utilized for a new engine application. When necessary, upgrade options that increase core flow capacity are allowed to achieve aggressive growth application requirements. Core component performance characteristics are maintained across common core variant designs and projected advances in technologies are taken advantage of through the infusion of improvements into the baseline common core definition. This work establishes a set of common core variant design rules, which impose

design restrictions representing geometrically common core applications with or without increased core flow capacity. These rules enable relationships to be established between the core definition and corresponding capabilities of various common core variant engine design options.

In addition to the common core variant design rules, another key enabler of the COMMENCE method is the integration of the Enhanced Robust Design Simulation (ERDS) method, which enables candidate designs to be evaluated under various uncertainty scenarios, allowing for the selection of robust candidate designs that achieve the highest likely performance. The method produces probabilistic surrogate models that instantaneously and accurately predict confidence interval performance levels of candidate designs throughout a wide variety of uncertainty scenarios. The form of these performance prediction models, when integrated into the COMMENCE method, enables the simultaneous robust design exploration of the multi-application common core design space in a highly efficient manner. The projected requirements of all variant applications can be mapped directly onto the core design variables. This allows for rapid identification of feasible core design regions for the set of engine applications considered, allowing for required core design margins to be directly determined, preventing the selection of a core definition of excessive size.

In order to test the individual enabling developments of the larger COMMENCE method, initial experimentation is performed. First the ERDS method is put to the test, comparing candidate robust design selections to those selected using a traditional deterministic approach. Next, the rules and environment used to simulate common core variant designs is tested. Various variant upgrade options are explored to determine thrust growth levels achievable at various levels of upgrade complexity. The third experiment tests the method's ability to simulate multiple common core applications simultaneously, imposing geometric commonality and searching for system feasibility across applications. The final experiment demonstrates the full range of capabilities enabled by the fully integrated COMMENCE method. A large number of engine applications are considered simultaneously for a common core engine design problem, each having unique sets of technology and requirements

uncertainty distributions. Common core design considerations are made, and simultaneous exploration of the common core design space is performed. Geometric commonality is implicitly enforced across all applications, eliminating the need for data filtering or design matching of variant applications. A feasible region of the engine core design space is identified, and a core definition is established to provide a high likelihood of achieving feasible variant applications for all sets of requirements considered.

While exploring the common core design space for a projected engine family with multiple applications to be released over time, several projected technology limits were identified to be limiting the required core size to higher flow capacity levels in order to meet all application requirements, resulting in compromised performance levels. It was found that through the accelerated maturation of technologies aiming to improve those component limits, a more compact core definition was able to be applied across the applications. The resultant common core variant selections maintained a high likelihood in meeting their requirements while achieving improved performance levels. By simultaneously considering all projected engine program applications, technology development options, and likely sources of uncertainty when exploring a common core design space, the designer is able to identify desirable design regions that avoid potential problems earlier in the development process, while also ensuring that decisions are made that maximize engine family performance.

Through development, testing, integration, and implementation of the COMMENCE method, the objective of this work was achieved. Interesting multi-application common core robust design studies can now be performed, offering unique observations to be made that previously had not been able to be observed. The COMMENCE method enables high order, multiple application common core robust design studies to be performed with greatly reduced computational burden compared to existing robust design methods. With minimal additional effort required upfront in the process, highly useful confidence interval probabilistic information is provided with minimal increases in the computational budget required to perform such a study. These high combinatorial design explorations enabled by the COMMENCE method provide significant returns in valuable program guiding information, allowing for more informed initial common core design decisions to be made.

CHAPTER I

INTRODUCTION

The days of trial and error design have long been over, but the stakes involved in the development of aerospace systems have never been higher. The development of a clean sheet gas turbine engine program can be a multi-billion dollar undertaking[80], where the decision to take on such a program can place the company at great risk. Furthermore, airframers require more and more capabilities from candidate engines for new vehicles while concurrently requiring high efficiency, competitive pricing, low maintenance costs, and reliable operation.

In order to spread capital risk among a large quantity of products, a gas turbine engine core is typically utilized on many engine variants. These common core engine variant designs are tailored to achieve various sets of customer requirements. These variants cannot simply be technically feasible options for the various requirements imposed, but they must maintain performance levels that make them the most competitive and economically viable options available to possible customers. With such a significant capital risk involved, there must be an equally proportional high level of confidence in the design decisions made in the initial development of a new engine program. A common core design selection must provide a wide variety of competitive current and future applications in order to remain successful well into the future.

This thesis describes the formulation and implementation of a methodology which seeks to identify and select a common core engine design based on the performance of likely derivative engine applications. This aims to increase the useful competitive life and overall versatility of a common core engine family. The core of a gas turbine engine contains the most expensive engine components, including the high pressure compressor, the combustor, and the high pressure turbine. A common core engine program aims to utilize the engine core design for many engine variant applications throughout the program's life. By designing a common core to provide a wide range of performance capabilities over time instead of

designing for a single set of requirements, the method aims to better account for future applications of an engine family in the initial development phase. Projecting implications of initial design selections on later common core engine applications will allow for the designer to evaluate whether certain decisions will improve or compromise application feasibility and/or performance levels of future engine applications.

1.1 Research Objective

This method, called the Common Engine Core Evaluation (COMMENCE) method aims at the selection of a turbine engine core definition that allows for a robust ability to employ competitive variants with a high associated level of confidence. Design rules will be established for the core development process, establishing a framework for an engine core design that enables competitive derivative engines throughout a large range of performance capabilities, while enforcing geometric similarity between the various common core variant applications considered. Instead of making design decisions with a single engine design in mind, a gas turbine engine core design cycle and corresponding size should be selected that will, from a business perspective, offer a maximum return on investment of the initial development program.

A turbine engine core design is typically utilized on many engine variants. This is due primarily to the high cost of its initial development. In 1990, General Electric (GE) estimated the development cost of the GE90 program to be on the order of \$2 billion[80]. In order to distribute the massive initial investment, the design must be employed on as many aircraft as possible. To accomplish this, variants of the engine are generated, creating unique engines tailored to various sets of customer requirements. These variants cannot simply be technically feasible options for the various requirements imposed, but they must maintain performance levels that make them the most competitive and economically viable options available. With the initial investment involved in gas turbine engine designs, the amount of confidence in the design decisions made in the initial development of a new engine program must be proportional to the capital risk involved.

Engines are designed to meet various requirements, and in terms of commercial turbofan

engines requirements typically consist of thrust levels at various operating conditions, specific fuel consumption (SFC) limitations, and extractions of bleed air and horsepower for the aircraft subsystems. Although the designer usually has a general idea of the range of thrust capabilities that the engine program should aim to gain customers, it is still extremely challenging to forecast future engine requirements. Projections of thrust requirements, extracted power demands, geometry, and regulatory requirements all must be made in the initial development of a successful engine program. Also present is the need to account for projected impacts that future technologies will have on the proposed engine applications. Actual performance impacts of technologies under development are not fully known until they are utilized and integrated into the engine system. In addition to these sources of uncertainty, there lies another source in the actual design, the variation of performance levels that will actually be achieved when the engine design under development is finally built and tested.

Successfully accounting for future requirements and the major sources of uncertainty can give the developer a strategic advantage in the commercial engine market. By taking these factors into account as early as possible, a robust core design can be selected and employed across the spectrum of engines needed in the aircraft fleet. This allows for the distribution of the program development costs across a greater number of product applications, reducing the acquisition cost of each unit offered to customers.

For commercial gas turbine engines, an engine's architecture, cycle, and corresponding size is typically selected to achieve required thrust levels while allowing for the best performance in terms of TSFC, engine weight, design mission fuel burn, noise levels, and/or other emission metrics such as NO_x emissions. If someone is simply attempting to search for the ideal solution for that particular aircraft, corresponding design mission, and a fixed set of technology assumptions, a traditional deterministic design method is sufficient. However, if one considers a commercial engine design from a practical standpoint, where recovery of the capital investment is one of if not the primary metric against which the engine design will be weighed, many more design considerations must be included. Analysis of the gas turbine engine market must be performed, identifying current and future capabilities and

engine family should be able to provide. Major sources of uncertainty present in the engine applications considered must be accounted for to ensure a high likelihood of successfully meeting the projected requirements of the various engine applications considered. The uncertainty present in the design of a single engine application is not only present for every other engine variant being considered for the program, variant applications being projected for future production have even greater levels of uncertainty present in their requirements and technology capabilities. Moving from the deterministic design of a single engine to the robust design of a multiple application common core engine family greatly increases the complexity and dimensionality of the design problem. This necessitates improvements to be made to the design process for the modeling, simulation, evaluation, and selection of candidate common core designs.

In the past, probabilistic design methods have been employed when attempting to achieve a robust single engine design. A framework must be utilized that takes advantage of the various methods that have been shown to be promising, and in addition must achieve advancements that allow for the simultaneous exploration and design of an entire family of engines without significant increases in the computational burden required. It is important for the resultant design method to be repeatable, transparent, and methodical in its approach, ensuring a academically sound approach to the challenges that must be overcome in the present work.

The overall engine architecture is typically consistent throughout an engine family, so the present work assumes a fixed overall engine architecture. Also, the high pressure system of the engine, typically called the engine *core*, consists of the high pressure compressor (HPC), the combustor, and the high pressure turbine (HPT). In order to minimize development, testing, production, and maintenance costs of an engine family, it is desirable to have a geometrically fixed core geometry throughout a family of applications, or at the very least, geometrically similar throughout the life of an engine program. Aerodynamic and material updates are made as technology level permits, but the overall casing and flowpath size should remain as geometrically similar as possible. Therefore, when deciding the size of the common core definition, the designer should take into account all of the future projected

growth thrust and power demands likely to be desired from the engine family architecture.

Primary Research Question: How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?

Exploration of various engine core designs must be performed, estimating their impacts on the ability of projected variant applications to meet projected requirements, while also estimating corresponding application performance levels. Within the scope of the current work, common engine core designs will be simultaneously evaluated for multiple applications that require a wide range of capabilities, while having various assumed technology limits.

Research Objective: Develop a gas turbine engine design and decision making process that aims to increase the useful competitive life and overall versatility of a common core engine family. The process should consider current and future competitive engine family performance, utilizing current and eventual technology improvements without the need for a core re-design.

The objective identifies the need of a design framework that incorporates the impacts of major sources of uncertainty into the common core design process for multiple engine family applications. The framework should enable studies to be performed that evaluate candidate core definitions under variations in design requirements, current and future technology impacts, and corresponding performance levels of all applications considered. The framework should also be able to evaluate the common core engine program's ability to achieve a wide range of thrust and power levels with a high likelihood of success. The core design decisions will be aided by the proposed framework, bringing knowledge of design implications on possible future product applications earlier in the core design process. This will enable the designer to determine how the core design decisions will impact the performance levels of initial and future applications of the engine program.

1.2 Gas Turbine Engine Program Initial Development

Early in the design process of a gas turbine engine program, core design decisions are made which begin to solidify the core definition and corresponding capabilities of the engine

core. In order to make these design decisions, it is necessary to identify where the core requirements are derived from. An engine core's primary role is power generation, tasked to do work on and add energy to the gas flowing through the engine. The core flow performs work on the low pressure turbine in order to produce thrust and/or power. Typically, there are additional horsepower extraction and bleed air requirements that tax the engine in order to power aircraft subsystems. In the case of a high bypass ratio turbofan engine, the energy added to the gas by the engine core is used primarily to power the engine's propulsor, or fan. The thermal energy exiting the high pressure turbine (HPT) is converted into mechanical work by the low pressure turbine (LPT) which powers the engine's fan and when present, the low pressure compressor (LPC), referred to as the booster. A majority of the thrust achieved by a high bypass ratio engine is produced by the bypass stream, which is compressed by the fan and expanded by the bypass nozzle, increasing the flow velocity and the momentum flux of the bypass stream. The engine core must be sized so that sufficient energy can be supplied to the LPT which allows for the engine to achieve the thrust levels desired, while staying within imposed temperature limits, throughout the design mission under consideration.

The current or projected technology level determines the temperature limits allowed at various locations within the engine. In order to provide a consistent standard for identifying various locations within gas turbine engines, SAE provides the standard for gas turbine engine flow station numbering[2][3]. The standard flow station numbers for the two-shaft turbofan engine architecture considered in the present work can be found in Figure 1. Typically, there is a limit imposed at the exit of the HPC, referred to as the $T_{3,max}$ limit. There are also limits most commonly applied to the inlet of the HPT and the LPT inlet, referred to as a $T_{41,max}$ limit and $T_{45,max}$ limit, respectively. These limits are set based on the thermal properties of the turbomachinery materials and various life cycle considerations. These temperature limits impact the maximum cycle pressures and temperatures, and also impact the amount of turbine cooling flows required. Knowing the core temperature limits and the predicted thrust demand of the engine at various points of the design mission, the designer can size a core that will supply the LPT with sufficient energy to produce required thrust

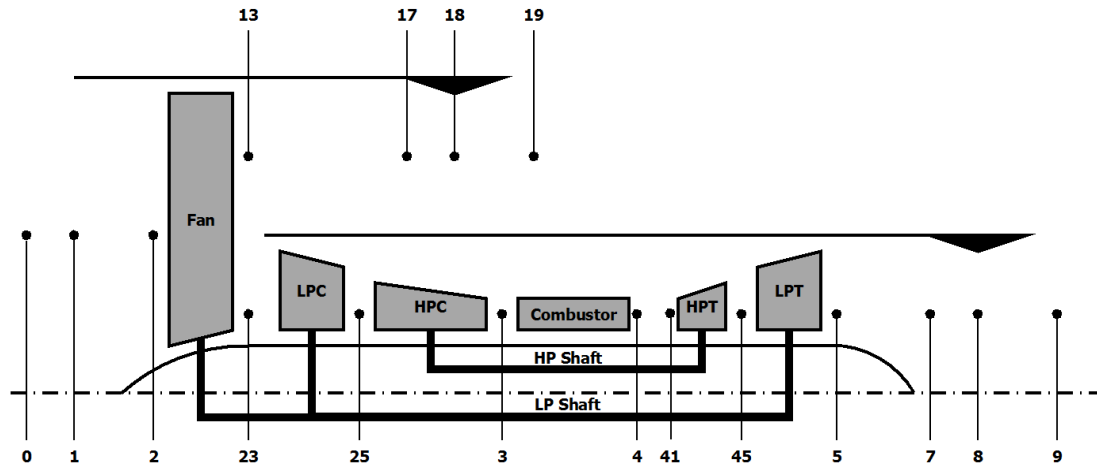


Figure 1: Standard Turbofan Engine Flow Station Designations[2][3]

levels throughout the design mission without exceeding the technology limits imposed. The HPC can be sized to provide a specified design pressure ratio, requiring a certain number of compression stages to supply the pressure while remaining below stage loading limits. Based on the amount of flow capacity of the core and the determination of power required by the HPC, the HPT can then be sized.

All of the core sizing decisions made should not only be based on the single design engine being sized, but should also consider engine variants that in the future will utilize the current engine core definition currently in development. Table 1 contains a list of core design parameters that values must be assigned for a common core engine program. For example, the actual geometric size of the core may need to be increased not for the current engine under development, but for a future variant that will require higher thrust levels through increased core working fluid. Material and aerodynamic improvements will almost certainly be made in the future, so the core design may benefit from higher design loadings. The higher design loading would likely cost the initial engine design in component efficiency, but would enable future derivative engines with aerodynamic improvements to utilize the same number of stages and flow capacity of the initial core design, requiring minimal design alterations for future variants while improving the overall engine family performance.

A question now arises that must be answered if the designer desires an engine core that

Table 1: Gas Turbine Engine Core Design Parameters

Parameter	Description
$nStg_{HPC}$	HPC Stage Count
$nStg_{HPT}$	HPT Stage Count
A_{25}	HPC Entrance Area
A_3	HPC Exit Area
A_{41}	HPT Entrance Area
A_{43}	HPT Exit Area
Chg_{HPT}	HPT Chargeable Cooling Fraction
$NChg_{HPT}$	HPT NonChargeable Cooling Fraction
Chg_{LPT}	LPT Chargeable Cooling Fraction
$NChg_{LPT}$	LPT NonChargeable Cooling Fraction

will offer the best overall performance for an entire family of engines. Should every projected variant carry the same weight when making design decisions, or should preference be given to different regions of the requirements space in order to achieve the best performance possible in those regions? If there is a high level of confidence that particular projected engine variants will have more customers, then those variant designs should carry more weight to ensure those variants perform well. In addition, the initial engine family members must perform well against the current competition in order to ensure entry into the market, preventing the engine program from premature failure. By adding preference to the performance of certain regions of the engine program application space, the designer may be able to improve the level of competitiveness of applications that aim to serve more projected customers, aiming at achieving the maximum possible fleet utilization. By increasing the number of units sold by the engine program, the initial development cost is able to be distributed among more engines and the individual unit cost will likely be reduced.

Competitive utilization of an engine core across a variety of applications usually requires modifications of the engine. Methods utilized to create engine derivatives must be considered. De-rating and throttling fixed engine designs is an inexpensive way to tailor an engine to specific customer requirements, but have negative impacts on both the performance and life of the engine. De-rating an engine decreases the operating temperatures the engine components are exposed to, extending the product life. However, significant

de-rating causes the components of the engine to no longer operate at near-peak efficiency levels, inducing performance penalties. Throttling the engine and elevating the temperature levels and operating speeds produces more core power, but reduces the life of the engine.

Resizing the low pressure system while having the engine core fixed is a more expensive development, but may pay off in performance for future variants that require significantly different thrust levels than an LP system was initially sized for. Also, geometric limitations can require a modified LP system. The typical placement of the engine under the aircraft's wing usually imposes a fan diameter constraint that can become active, requiring a smaller fan to be employed for a particular engine application. If the designer aims to develop an engine core that achieves the best overall engine family performance, then projected customer requirement variations and corresponding engine variants must be modeled and evaluated simultaneously in order to predict the likely engine performance with the selected common core design.

Research Question 2: For a given gas turbine engine core, how should a common core engine variant design be simulated? What parameter(s) must be held to consistent values in order to maintain geometric and aerothermodynamic commonality between engine applications?

1.3 Design for Maximum Utilization

Maximum return on investment for engine design is achieved by satisfying two primary goals. The first is to minimize the capital investment in the development of a product line. The second is ensuring the product line provides a wide range of competitive solutions. In other words, the program must offer products that achieve high performance at a low cost in order to attract customers. The designer must make sure their product line performs well for a variety of possible requirements, including the range of possible thrust levels, power and bleed demands of the aircraft, and any geometric constraints imposed. With consideration of the current aircraft fleet distribution as well as projected vehicle retirements, the designer can accommodate projected customer needs in the initial development program of their engine design. Accommodating the fleet would enable the designer to design an engine

core according to estimates of the thrust range which will offer the maximum number of possible customers, while also placing preference on certain thrust and/or power levels in the requirements space where more demand is projected to exist.

Research Question 3: What design options should be considered for common core engine variant applications in order to distribute development capital across the engine program by taking advantage of commonality, while also offering more design freedom when needed for more demanding applications?

With or without the consideration of current or projected fleet distributions, the engine designer should make design decisions that maximize the competitive thrust spectrum of the engine program. When designing the engine core, the range of thrust and power levels an engine line can competitively achieve should be maximized. The designer must determine what range of thrust levels the core will aim to provide competitive applications, as well as the impact of technology maturation on the achievable engine program capabilities. At a given technology level and corresponding temperature and efficiency limits, there is a limit to how much energy can be added to a fixed amount of core flow. This limits the amount of work the core flow can perform on the engine's propulsor. Therefore, the engine core size selected for an engine program is highly correlated with the maximum amount of thrust achievable with fixed core flow levels. On the other hand, as the size of the engine core increases, the ability to efficiently provide low thrust engine program applications is diminished.

Research Question 4: What range of capabilities can various common core design options achieve without significant compromises made in application performance?

Making core design selections requires evaluation of how well a specific decision achieves the designer's goals. Metrics must be used that will assist in determining whether an engine family provides a wide variety of *competitive* capabilities at various technology levels. Proper evaluation of engine designs under consideration is crucial in order to make well informed design decisions. Otherwise, the engine family design performance could be compromised by not considering the correct metrics for evaluating an engine family design.

Research Question 5: How should a common core engine program consisting of multiple variant design applications be evaluated?

How well an engine core provides a wide range of successful engine derivatives should be determined within the context of performance and economic levels. Performance of candidate commercial engines are traditionally determined by estimating TSFC and corresponding mission fuel burn levels. However, if candidate engine performance was the only aspect considered, then unique engine designs, including core designs, would be employed for every new set of customer requirements, which would be extremely costly. Therefore, consideration should also be made on economic aspects of design decisions made for the engine family. Not necessarily to estimate the acquisition costs of engines, but to place the engine core and its corresponding derivatives on a cost scale. The size of the engine components as well as the temperature levels the engine operates at will provide physical relationships to the relative cost of a particular engine. In order to achieve higher temperature levels, engine material and cooling technologies must be developed and integrated, requiring significant costs. A cost scale should consider temperature levels of the design cycle, the number of component stages and corresponding loading levels, and the complexity involved in achieving high performance variants throughout the thrust spectrum. By placing an engine application on the cost scale, the designer will not estimate the dollar amount required for an application, but will be able compare two proposed variants for a particular application.

Utilization of a common core engine design requires performance compromises to be made. The engine core size must be selected in order to provide ample power throughout the range of possible requirements. Therefore, for the sectors of the requirements space that would require lower power generation from the core, the common core design would likely be over-sized for that particular application. An over-sized engine core is heavier than necessary, and may operate at lower efficiency levels than a core sized specifically for an application's specific requirements. Evaluations of candidate common core designs and possible engine family members for a commercial engine program should be carried out with consideration of at least some the following characteristics of the engine design:

- TSFC levels at various flight conditions
- Fuel burn levels when *flown* on board various aircraft throughout design missions
- Thermal, propulsive, transmission, and overall efficiency levels achieved by the resultant engine design
- Core air flow levels at the upper and lower desired design thrust levels and associated thermodynamic flow properties
- Complexity of engine derivative design modifications required to achieve high performance throughout the thrust spectrum
- Number of unique low pressure systems required to achieve high performance throughout the thrust spectrum
- Core turbomachinery stage counts and corresponding loadings at various technology levels
- Engine core growth capabilities with increases in technology level and various degrees of design freedom allowed

In order to evaluate a common core design, maintaining a common engine core, in terms of core architecture and size, upon which different low pressure systems will be applied to will be one challenge that must be overcome. The geometry of the core must remain fixed in order to simulate a common core engine model with a geometrically fixed design core. However, technology updates to the materials, aerodynamics of the turbomachinery, and cooling network modification must be able to be applied to the common core variant applications, enabling higher temperatures and power outputs of the engine core, while still maintaining a geometrically fixed engine core. A set of engine core parameters that entirely defines a geometrically fixed or similar engine core must be established and maintained throughout any engine family design exploration exercise. Otherwise, the evaluation of the common core engine family design would be invalid. Table 2 contains an entire list of engine core design parameters, along with categorizations of which ones remain constant and which ones can vary within a common core engine program.

Table 2: Engine Core Design Parameter Categories

Parameter	Description	Category
W_{25R}	Design corrected core flow	Application Specific
W_{3R}	HPC design exit corrected core flow	Fixed for Program
π_{HPC}	Design HPC pressure ratio	Application Specific
N_C	Design core shaft speed	Application Specific
$T_{3,max}$	Design maximum HPC exit temperature	Technology Limited
$T_{41,max}$	Design maximum HPT inlet temperature	Technology Limited
$T_{4,max}$	Design burner exit temperature	Technology Limited
$nStg_{HPC}$	HPC stage count	Minor Modifications Allowed
$nStg_{HPT}$	HPT stage count	Fixed for Program
A_{25}	HPC entrance area	Minor Modifications Allowed
A_3	HPC exit area	Fixed for Program
A_4	Burner exit area	Fixed for Program
A_{43}	HPT exit area	Fixed for Program

Employing the Common Engine Core Evaluation (COMMENCE) method, decision makers will be able to select an initial engine core design with better knowledge on how different core design parameters will affect the future capabilities and limitations of the core design under consideration. They will be better informed on how robust their engine program is to the variation in possible customer requirements and projected technology impacts, ensuring their initial design will provide competitive solutions for their customers now and in the near and distant future with a high associated level of confidence. The major sources of uncertainty affecting future applications, particularly the capabilities required of the engine program and the technology limits achievable will be accounted for, allowing the designer to bring the projected impacts of these up front in the design process. This will allow the designer to select a common core design that will be robust to future requirements and constraints, creating the ability to build the proper growth potential into the engine program's core design.

CHAPTER II

BACKGROUND

In order to understand and improve upon the traditional gas turbine engine design process from a practical standpoint, it is important to have a historical perspective on existing successful engine programs. One such program is the CFM56 family of engines. CFM International (CFMI) has been able to successfully utilize their engine family since the 1970s, offering a great example of a robust common core engine program. It is also important to understand the traditional academic approach to turbine engine design, allowing for the identification of regions of the process that must be refined for robust common core design. Specifically, traditional approaches to the determination of the engine architecture and design cycle should be understood, as well as the requirements and constraints considered during this selection process.

This chapter provides historical lessons learned from the CFM56 product family. It then provides traditional approaches to commercial engine design, exploring how requirements and technology limits affect the engine cycle performance and the selected cycle of an engine design. The latest approaches to accounting for major sources of uncertainty and ways of optimizing the design will also be presented. The goal is to provide a clear basis for which the proposed common core design methodology will enhance the designer's ability to select an engine core design that is robust to requirements and technology limits of current and future applications.

2.1 The CFM56 Engine Family

CFM International (CFMI), a joint partnership of two aircraft engine manufacturers, Snecma (SAFRAN Group) of France and General Electric (GE) of the United States formed in September 1974 in order to compete with Pratt & Whitney's JT3D and JT8D engines in the "10 ton" thrust class. The resultant engine, the CFM56-2 first flew in February 1977 and was certified in November 1979. The CFM56-2 was selected to fly on the stretched

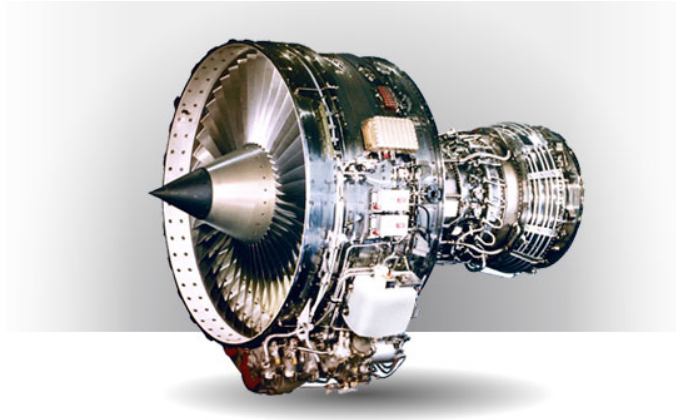


Figure 2: CFM International's CFM56-2 Engine[1]

DC-8-70's and then on KC-135R tankers. Once confidence was gained in CFMI through the introduction of the CFM56 to the fleet on the DC-8's, the CFM56 shown in Figure 2, became a popular and viable option for new aircraft. In 1981, Boeing's 737-300 was first launched, mounted with CFM56-3 engines on board. Then in 1984, Airbus selected the CFM56-5A for their launch of the A320[12].

By making their way onto the 737-300 and A320 aircraft early in each of their respective programs, CFMI ensured they would maintain relationships with both major commercial airframers. Not only did this encourage the use of CFM56 engines on later 737 and A320 variants, but it also made CFMI an attractive company for the airframers to work with in the development of their latest single aisle aircraft designs. The CFM56 engine family is a model of how to gain entrance into a competitive market, and to nearly guarantee their products' use well into the future. Below is a quote from the CFMI website.

“CFM's product line includes six engine models that are tailored to the aircraft applications they power. From the initial CFM56-2 to today's Tech Insertion CFM56-5B and CFM56-7B, our engines are the most sought-after in the industry. Spanning a thrust range of 18,500 to 34,000 pounds. CFM engines are the power behind more than 8,000 commercial and military aircraft. CFM's vast field experience of nearly 500 million flight hours is built into every engine we

make. Blending proven reliability with cost-effective technology, CFM engines lead the industry in reliability, durability, performance retention, and time on wing.” [1]

In order to gain insight into how the program became so successful, several questions should be asked:

- How was the CFM56 engine architecture and corresponding cycle and size initially determined?
- From an engine design perspective, how did CFMI ensure that their CFM56 design would allow for as much growth as the program has seen?
- How were competitive variants of the CFM56 designed and selected?
- How did CFMI decide when to introduce technology updates to their engine program?

2.1.1 Initial Development - The CFM56-2

In order for CFMI to have a successful engine program, they needed to enter a competitive market with an engine that performed exceptionally well while also providing growth capability for future customers. They knew higher bypass ratio engines than those currently flying would offer the competitive edge in the market at the time. They also knew that they needed to build significant growth potential into the engine program. Shortly after the GE/Snecma partnership was officially formed, in a February 1975 issue of Aviation Week, the following statement was made concerning the growth potential of the CFM56:

“The CFM56 has a built-in and planned growth potential to 24,000 lb. thrust and then to 25,000 lb. with increased airflow and higher operating temperatures. Also planned is a 27,500-lb.-thrust version that will require definite modifications from the preceding member of the engine family.” [117]

In order to enter and gain the competitive edge in the single aisle market, CFMI focused primarily on achieving higher fuel efficiency, but also placed significant focus in the reduction of maintenance costs. The design philosophy of the CFM56 began with the use

of leading edge technology which enabled a compact core and advanced cycle. They also considered room for improvement, enabling the infusion of new technology when it had reached maturity. The CFM56 was deliberately designed and tested at maximum severity conditions, then was de-rated when entering into service[12]. This allowed for the evolution of the CFM56 over time as growth of the engine was demanded by the market.

The testing program of the CFM56 consisted of “extra-severity” testing. Core testing was performed at ratings more severe than what was planned during commercial operation. The testing was done at growth power levels of temperature, speed, and pressure. The purpose was to find potential problems earlier and at a faster rate than the typical testing program. This allowed significantly more time for corrective actions before commercial service began[78].

For example, the CFM56-2 engine was regularly tested at thrust levels in excess of 26,000 pounds of thrust, and was certified at a lower level of 24,000 pounds. The DC-8's on which the -2 first operated on in 1982 however, required less than 22,000 pounds of thrust with additional de-ratings around 19,000 pounds[32]. This not only guaranteed reliability by designing and testing at temperatures and pressures well above what they would experience during commercial operation, but it was also an ingenious way to build immediate growth potential into the initial engine. Not only was CFMI able to achieve TSFC levels 10 to 20% better than the competing low bypass ratio engines[12], but they were able to lay a foundation for which to launch major thrust growth variants in conjunction with regular evolutionary technology updates to the product family.

2.1.2 Boeing Application - The CFM56-3

Due to the attractive performance of the CFM56 during it's initial introduction to the fleet, Boeing selected it to be on board its new 737-300, which entered service in 1984. The use of the CFM56-2 showed that high bypass engines that were being used on the larger aircraft could be applied to offer substantial performance improvements on the smaller single aisle vehicles. CFMI developed the CFM56-3 engine for the 737-300. The CFM56-3 was smaller than the -2, due to geometric constraints imposed on the engine diameter due to the low

Table 3: Characteristics of the CFM56-2 and CFM56-3 Engines[32]

	CFM56-2	CFM56-3
Thrust @ SLS	24,000 lbs	20,000 lbs
Fan Diameter	68.3 in.	60 in.
Bypass Ratio	6	5.1
Weight	4,612 lbs	4,278 lbs
Length	95.7 in.	93.0 in.
Certification	Nov. 1979	Sept. 1983

wing of the 737. Some of the defining characteristics of both the -2 and -3 engines are shown in Table 3 for comparison. It required less thrust and therefore had a smaller fan diameter and bypass ratio than the -2. The smaller fan design also gave the engine proper clearance when mounted under the 737-300 wing. This constraint also required the gearbox and accessories, normally found underneath the engine for easy access, to be placed on the side of the engine. The most attractive aspect of the CFM56-3 was that it offered an uninstalled TSFC improvement of roughly 20% compared to the competing low bypass ratio engines on similar single aisle aircraft at the time[32]. In addition, about 90% of the parts were common with the CFM56-2, allowing for much of the testing and maintenance experience to be applied to the -3 engine. Specifically, the entire core as well as the low pressure turbine (LPT) of the -3 engine were the same as what was in the -2 engine, which had already been significantly vetted through the extra severity testing program[32].

In the same manner that the CFM56-2 was tested at performance levels beyond its certification levels, the CFM56-3 was tested to achieve thrust up to 23,000 lbs while it was planned to be certified for 20,000 lbs of thrust and operated at even lower levels. In the normal operation of the 737-300 at stage lengths of 500 miles or less, the takeoff thrust would be de-rated between 10 and 15 percent below the engine's certification level, with exhaust gas temperature (EGT) levels on the order of 300° F below it's certification redline[32].

A majority of the differences between the CFM56-3 and its parent, the CFM56-2, were contained within the low pressure compression system. In fact, the -3 engine employed the same exact core and low pressure turbine as the CFM56-2[32]. The -3 engine was designed

to operate at the same low pressure rotor speeds as the -2 engine. This offered the smaller fan to have lower tip speeds than the larger fan design on the -2 engine. The fan for the CFM56-3 was a scaled CF6-80 design, which offered high efficiencies at the operational tip speeds. The smaller fan of the CFM56-3 provided an efficiency improvement of almost 5-1/2 percent over the tip shrouded -2 design, offering almost 3% of improvement in cruise TSFC. The reduced loading of the fan also enabled a reduction in the number of struts from 12 to 8. In addition to allowing for a structural improvement, the aerodynamics of the new fan required a redesign of the low pressure compressor (LPC) for the CFM56-3[32].

2.1.3 Airbus Application - The CFM56-5

The next member of the CFM56 family came with the launch of the Airbus A320. The aircraft was designed to carry 150 passengers over ranges up to 3,000 nautical miles[47]. In 1984, the A320 launched with the CFM56-4, but an improved version was offered shortly after in 1988: the CFM56-5A1, a 25,000 lb thrust class engine. In the design of this advanced technology derivative, aerodynamic improvements were applied to virtually all of the components, including the engine core components and the 68 inch diameter fan. Full Authority Digital Electronics Control (FADEC) was a major addition to the new design as well. The -5A1 not only emphasized high performance at the design point(s), but also maintaining high performance during off-design operation. The engine reportedly provided cruise TSFC improvements greater than 11% when compared to the CFM56-2[12].

Significant upgrades were made to the common engine core architecture of the -5A1. Aerodynamic improvements were made along with FADEC stator schedules to improve the HPC performance. Also, significant attention was paid to the interaction between the HPT and LPT performance, requiring full scale rig testing. This, in addition to aerodynamic improvements made to the LPT provided about 40% of the -5A1 performance improvement with respect to the -2 engine[12].

The design of the -5A1 offered significant growth potential, which later became a key factor in its selection for the long range Airbus A340. The CFM56-5C2, a high thrust derivative of the -5A1, was designed to provide up to 36,000 lbs of thrust by utilizing a

72 inch high flow fan, as well as improving the aerodynamic components of the -5A1. The -5C2 entered service in 1993 providing 31,200 lbs of thrust for the A340. It utilized the same core as the -5A1, with the addition of a fourth booster stage, a long duct mixed flow nacelle and a fifth LPT stage to provide ample power to the new high flow fan[12].

In 1989, focus was then put on the design of a high thrust derivative for the stretched A320: the CFM56-5B. Again, the common core of the -5A1 was utilized. The only modifications needed from the -5A1 was a four stage booster and new 68 inch high flow fan in order to achieve 29,000 to 31,000 lbs of thrust with the -5B. Everything aft of the low pressure compressor was consistent with the 1992 version of the -5A1 engine.

2.1.4 Boeing Next Generation - The CFM56-7 and Beyond

With the proven track record of the CFM56-3 on board the 737-300, -400 and -500, it was an easy decision for Boeing to select CFMI for their Next-Generation 737. Development of the CFM56-7B began in early 1994. The design took advantage of the latest technology of the -5B, while also outfitting the engine with an advanced 61 inch fan. The engine is able to achieve growth thrust up to the 27,300 lb thrust capability of the -7B27.

Figure 3 displays the various thrust levels achieved by members of the CFM56 engine family and each member's corresponding weight. This shows that a single common core engine program was able to gain customers with competitive solutions producing max takeoff thrust ratings ranging almost 15,000 lbs, offering more than a 50% increase in rated thrust levels from the initial design over the life of the program. CFMI showed that building in accommodations for new technology in their modular design approach enables rapid and competitive design responses for future customer requirements. The modular, compact and highly loaded core employed in the CFM56 allowed for continuous technology insertions as the new materials and manufacturing improvements were made. Not only did this allow for rapid development of new derivatives, but also allowed for advanced technology insertions into their existing fleet. The growth potential built into each base variant by testing and certification at higher thrust levels than initially required allowed for extensive thrust growth with very minimal modifications. Figure 4 shows the evolution of the CFM56 engine family,

including the technology insertions made to earlier models employing the technological developments of the later engine models.

2.2 Gas Turbine Engine Design

The gas turbine has been theoretically explored since the late 1700s, with the first related patent by John Barber in 1791[8]. The general process for the design of the gas turbine engine is well established, allowing for an acceptable process to which additions will be added to improve upon the more practical process of selecting a common engine core design for a family of gas turbine engines. Taking advantage of recent developments in engine design will enable simultaneous consideration of a variety of requirements and constraints when making decisions and down-selections that define the core design.

2.2.1 Traditional Design Process

The design and development of a new gas turbine engine begins with market research accompanied with or resulting in customer requirements. If market research determines a need for a new engine, whether there is a lack of available engines for a desired thrust class, or the advancement of technology permits a new entry into the market, an engine company may start exploring their options for the development of a new engine. Conversely, if a customer, usually an airframer, needs a new engine developed, the engine company will also perform some preliminary studies to offer the airframer with a proposed competitive solution.

The requirements of the engine are what determine the engine's architecture, resultant cycle, and overall size. Typical engine capabilities account for mission requirements, such as thrust requirements at various conditions as well as power extraction requirements for aircraft subsystems. From an aircraft's perspective, mission fuel burn and overall engine weight must be minimized. There are also geometric requirements, typically due to fan diameter constraints imposed by the typical engine placement under the wing of the aircraft. Additionally, a key design requirement often overlooked from an academic perspective is the minimization of the various system costs.

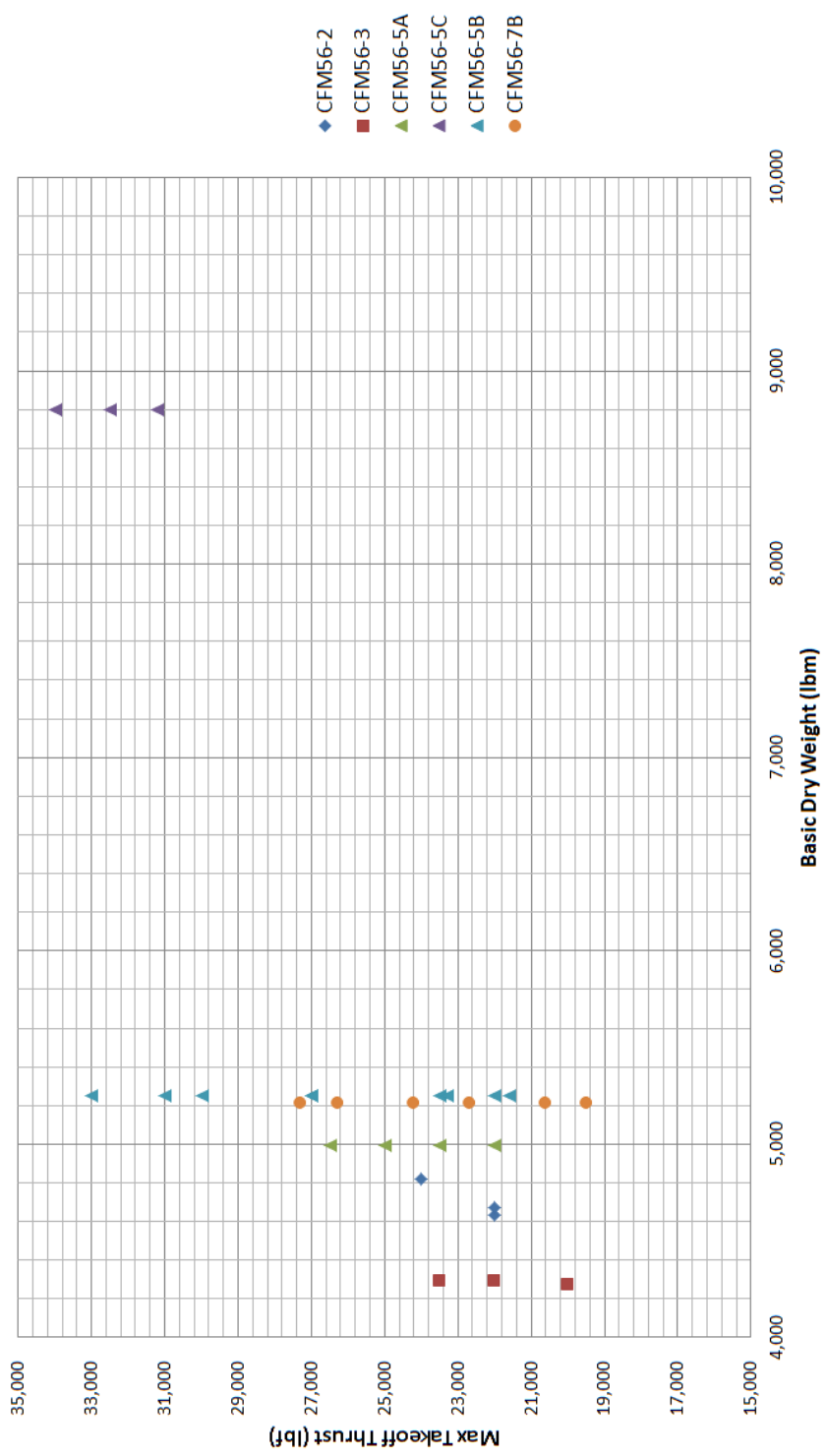


Figure 3: CFM56 Engine Family Thrust Capabilities

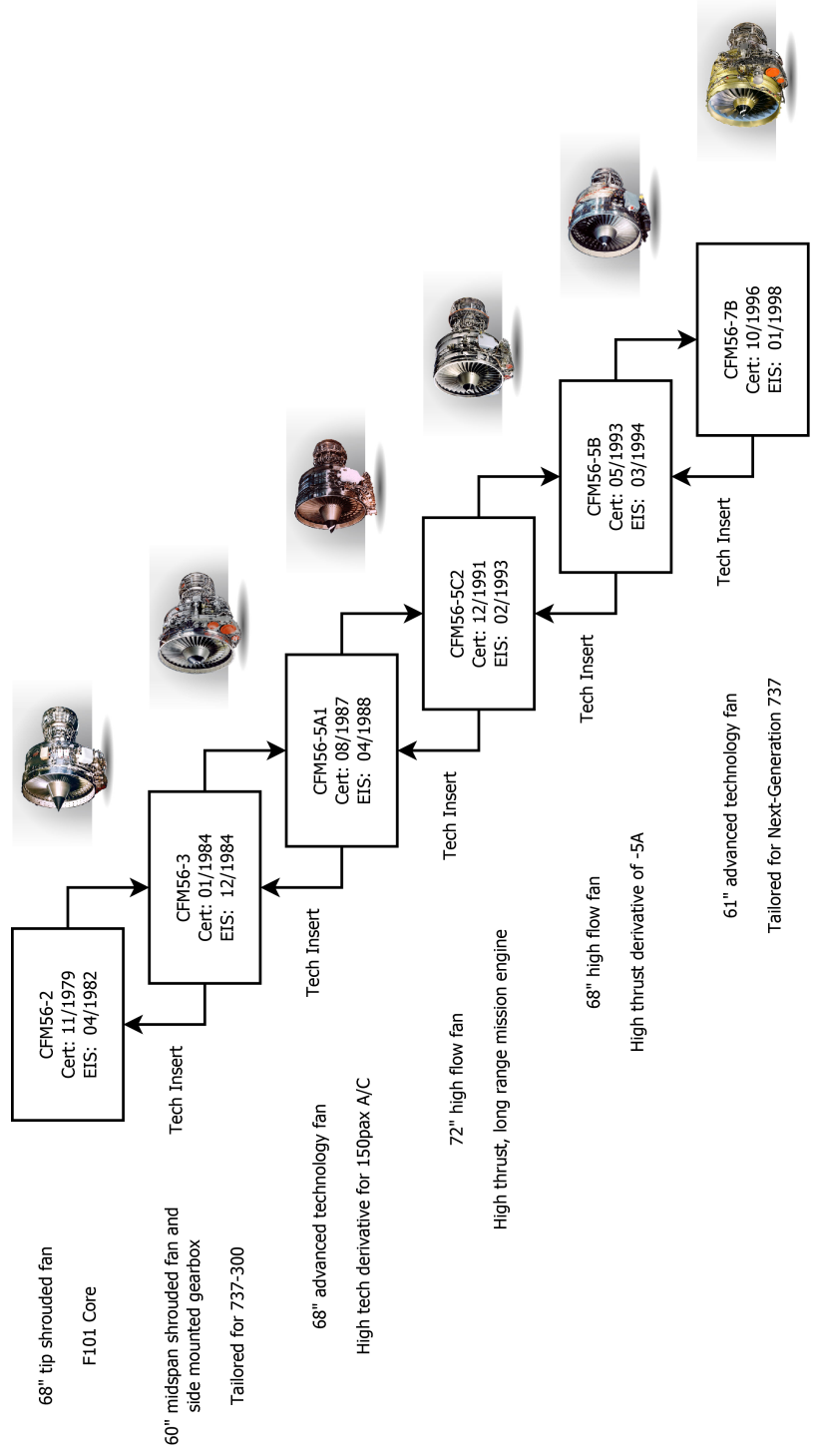


Figure 4: CFM56 Engine Family Evolution

The high temperatures that the turbomachinery components experience significantly affect the amount of maintenance required over the engine's life cycle. The overall parts count also has an impact on maintenance costs. The various costs involved in the development, testing, production, certification, operation, and maintenance is of the utmost importance to possible engine customers. A candidate design that is not economically viable, regardless of its likely performance levels, will not be attractive to possible customers.

The consequences of making incorrect design decisions can be very significant, particularly when poor initial design selections are made that limit the design freedom as a development program matures. Expensive mitigation actions may be necessary to meet missed requirements or improve an application performance characteristics. In addition to the added costs involved, mitigation of a design based on a posteriori knowledge would likely have accompanying compromises in achievable performance, compared to a more informed initial design selection. Worse yet, if design decisions are made that greatly compromise an engine family's initial engine applications, the entire engine program may be at risk of premature termination. The immense consequences of incorrect design decisions require great care to be taken when simulating and evaluating, and selecting candidate designs.

Oates provides the traditional procedure for matching of an engine and airframe for a projected aircraft mission[85]. This framework, displayed in Figure 5, aims to converge upon an aircraft design that results in an engine that is sized to provide the needs of the aircraft system. The objective of the process is a design selection that is very likely to meet mission requirements with the best performance. Oates notes the advantage of using a digital computer to quickly and efficiently assess a large number of candidate configurations. However, he notes that the quality of the final design selection depends greatly on the engineering judgment as well as the accuracy of the analytical methods used to simulate and evaluate candidate designs. This highlights several aspects that must be present in a gas turbine design and selection process. First, an engine must be able to meet all requirements while achieving competitive performance levels. Second, the design tool used must accurately represent the real systems being simulated, allowing for accurate performance estimates to be made for candidate designs. Finally, judgment of the engineer should not

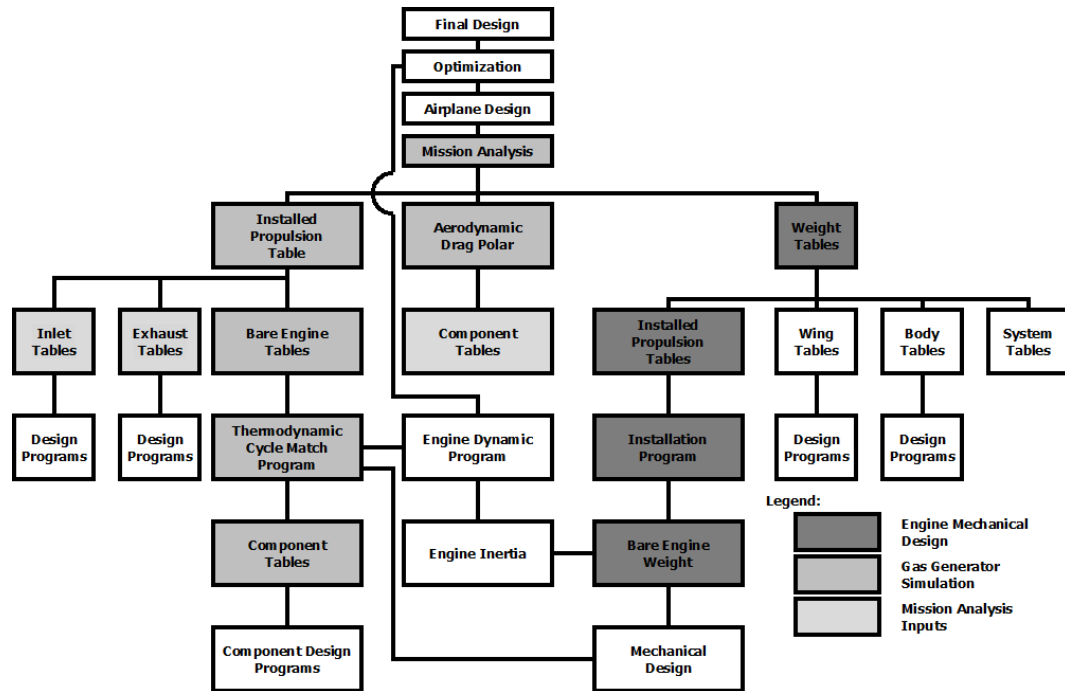


Figure 5: Oates engine/airframe matching process.[85]

be outside of the process. No matter what a design tool suggests, an designer must make the final selection based on all the information available to him.

Relating Oates' design process to the objective of the present work, his observations highlight several requirements of a common core design tool. Evaluating a candidate design based on its ability to provide assumed requirements is of primary concern. Second, the method must maintain accuracy in performance estimates of candidate designs. Lastly, the tool must provide the decision maker as much information about candidate designs as possible. In terms of a multiple application common core design problem, this challenge of maintaining model accuracy while providing substantially more useful information to the engineer must be overcome. This is of chief concern in the present work which aims to evaluate common core designs through the simulation of all projected applications under various likely sources of uncertainty.

Once engine design requirements are specified or projected, preliminary studies are performed. These studies help in the selection of the engine architecture, as well as the

selection of the engine cycle that creates a competitive solution that meets or exceeds all requirements. Engine cycle design space exploration assists in the determination of which values of design cycle parameters allow for sufficient performance at various design points of the engine[103]. If it is determined that a particular engine cycle satisfies all performance requirements, it is considered a candidate engine cycle.

For each candidate engine cycle, the predicted flowpath is then generated. In other words, dimensions are added to the otherwise dimensionless cycle analysis. This assists in the estimation of the engine component dimensions, component stage counts necessary under the assumed loading levels achievable, and the predicted weight build up of the propulsion system.

Next an engine deck is generated, consisting of throttle hook data for flight conditions throughout the possible flight envelope of a particular aircraft mission. A typical engine deck contains engine performance characteristics throughout all possible flight altitude, Mach number, and throttle levels the engine may experience during the design and alternate missions.

Mission analysis is then performed, simulating the mission in order to determine how well the vehicle performs. This analysis also determines if engine or vehicle re-sizing is necessary or desired. If the vehicle is resized, then the new required thrust levels are fed back to the cycle analysis process and the entire vehicle design loop is performed again, until convergence is reached. This iterative process, from engine cycle analysis to mission analysis, results in the overall vehicle performance of the candidate engine cycle under consideration.

Mattingly[68] offers a process flow for gas turbine engine design, depicted in Figure 6. The dashed regions of the engine design process represent the primary focus of the present work. Take note that the consideration of *uprated and modified versions* of the engine design occurs far downstream in the process, well after the engine design has entered the detailed design phase.

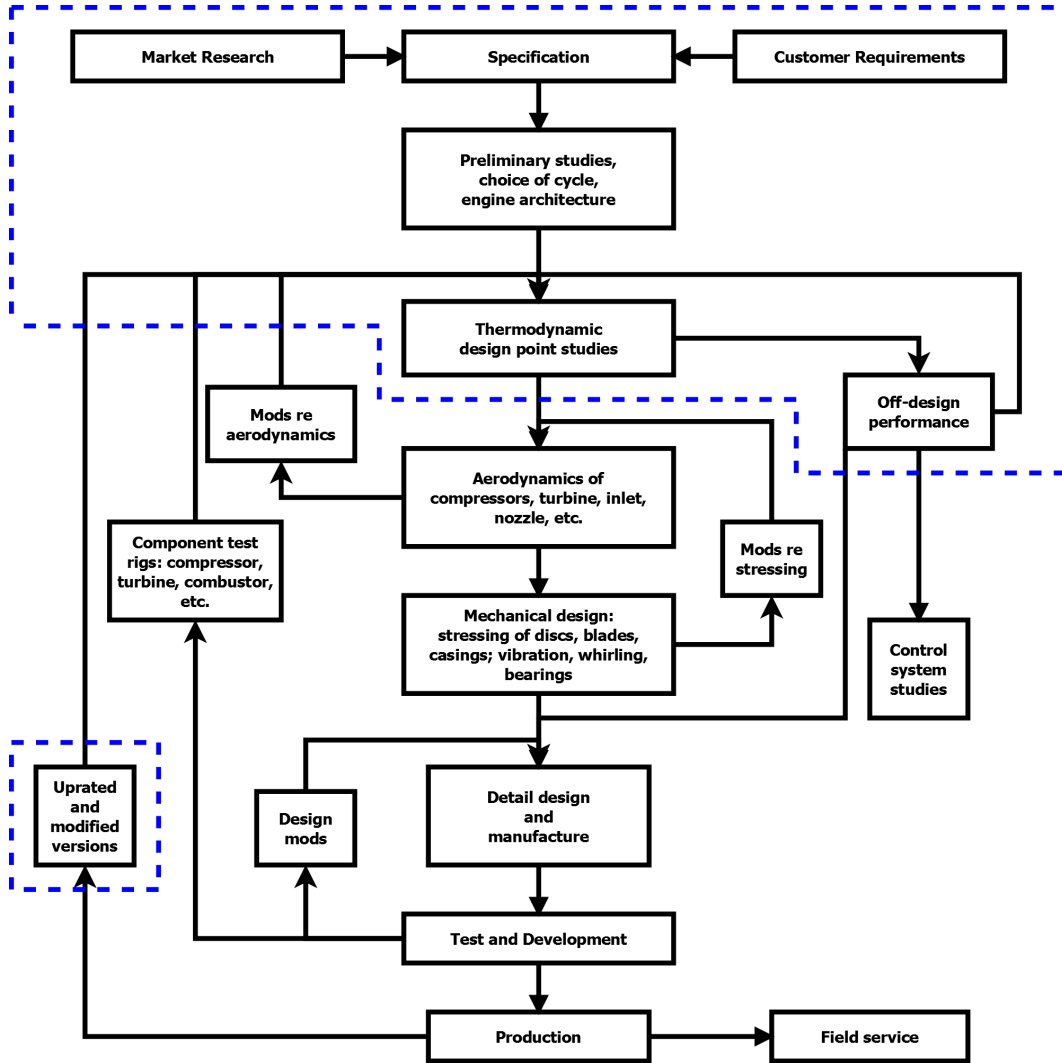


Figure 6: Mattingly's Gas Turbine Engine Design Process[68]

2.2.1.1 Preliminary Design

The primary goal of the preliminary design of a commercial turbofan engine is the establishment of a thermodynamic cycle and corresponding engine size which is optimized to minimize block fuel burn for a specified airframe and design mission, while meeting a variety of restrictions, or constraints[59]. Preliminary design of propulsion systems begins with requirements, such as specified performance requirements as well as requirements established through constraint and mission analysis[36]. Design parameter values converge, resulting in an engine design that meets all of the requirements and satisfies all constraints under consideration.

The engine cycle is analyzed at various design points under consideration and design cycle parameters are selected to provide a technically feasible design that maximizes the figures of merit used to compare different candidate engine cycles. Sizing of the engine is performed, which determines the geometry of the engine that meets thrust and power requirements specified by the customer. Aircraft performance can then be predicted, determining whether resizing of the engine is required. Mattingly et al.[69] presents the preliminary design sequence for propulsion systems, which is shown in Figure 7.

2.2.1.2 Requirements and Constraints

Requirements and constraints together determine the feasible design space of a propulsion system. Requirements of the propulsion system consist of performance targets and performance limits. Performance targets, such as required thrust levels at various flight conditions of a design mission or horsepower demands of the airframe must be matched. These targets can be treated as equality constraints, shaping the engine cycle design space (CDS)[102]. Performance limits usually consist of requirements that must be sustained at minimum levels. These can be treated as inequality constraints, setting the boundaries of the feasible CDS. Fowler[36] compiled a list of typical design requirements, consisting of vehicle requirements affecting the propulsion system design. In addition to performance limits, technology limits also constrain the engine CDS[102]. These constraints consist of material temperature and component loading limits, as well as achievable turbomachinery

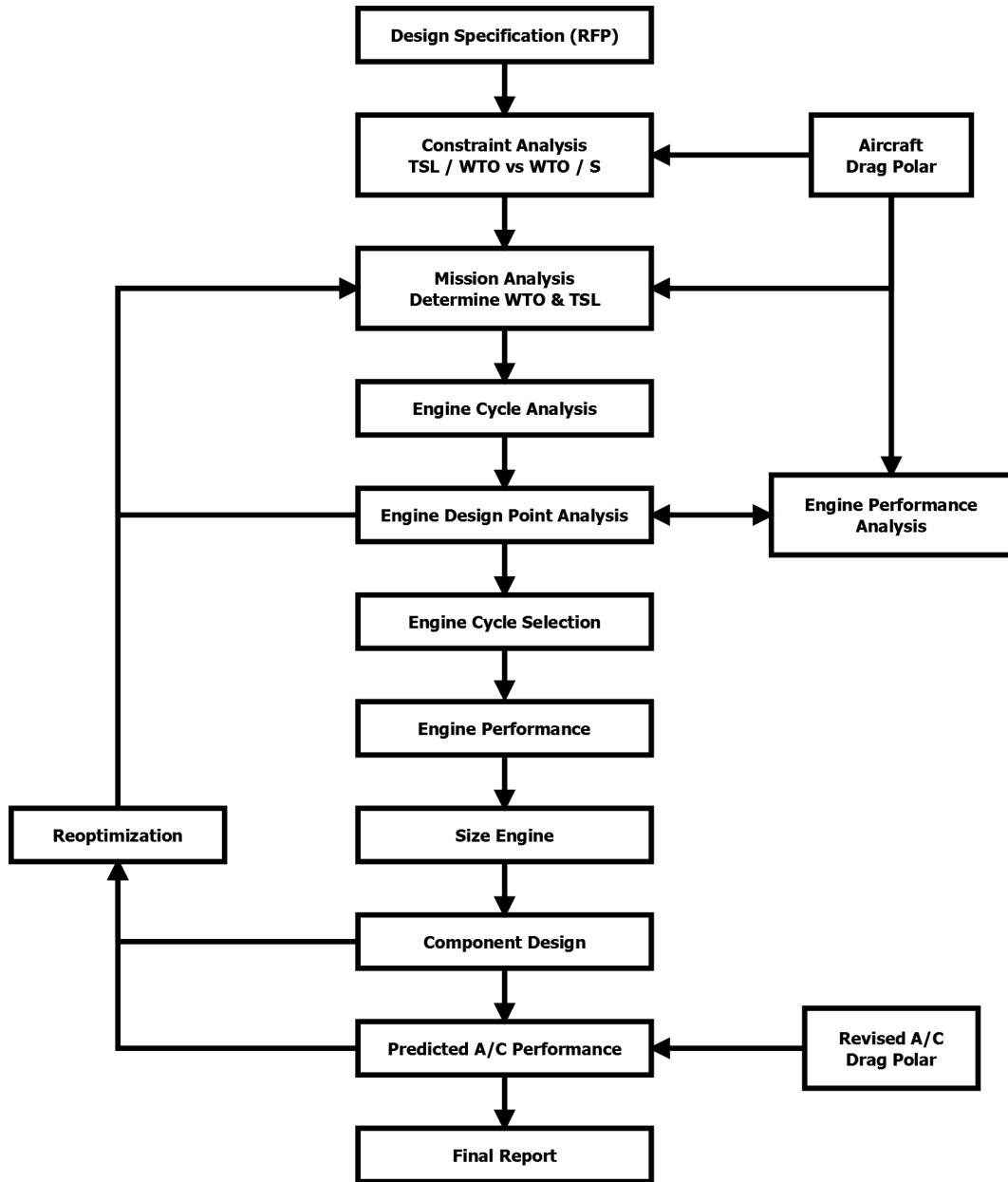


Figure 7: Mattingly's Preliminary Engine Design Process[69]

Table 4: Typical Commercial Transport Engine Design Requirements and Constraints

Commercial Transport Engine Design	
Requirements	Constraints
- Range	- Turbine Inlet Temperature Limit
- Payload	- Cooling Flow Limits
- Balanced Field Length	- Rotor Speed Limits
- Top of Climb Thrust	- Shaft Torque Limits
- Takeoff Thrust	- Mach Limits
- Engine Horsepower Extraction	- Component Loading Limits
- Engine Out Climb Gradient	- Maximum Component Pressure Ratios
- Noise and Emissions	- Nozzle Area Limits
- Growth Capability	- Fan Diameter Limits

component efficiency levels. Table 4 provides lists of requirements and constraints that are typical when considering a commercial transport engine design.

2.2.1.3 Engine Cycle Analysis

The objective of turbine engine cycle analysis is to determine the performance of an engine cycle[13]. This includes the process of obtaining estimates of the performance parameters in terms of design limitations, flight conditions, and design choices[86]. The performance parameters Oates[86] refers to are primarily thrust and specific fuel consumption. Design limitations can be numerous, including maximum allowable temperatures due to the current level of technology, or weight and geometry limits imposed by the customer. The design choices consist of the engine architecture(s) under consideration, the size of the engine, and the candidate engine cycle selection.

A candidate aerothermodynamic engine cycle needs to be both feasible and offer the competitive performance for the vehicle application at hand. The measure of performance can be the efficiency of an engine, whether it be the thermal, propulsive, or overall efficiency. The thrust specific fuel consumption (TSFC) at a design operating point is a popular measure of performance of the engine. However, in order to find the engine cycle that allows for the vehicle to perform best, the estimated design mission fuel burn should be minimized.

At a given technology level, the turbofan engine designer typically has four thermodynamic parameters of which he or she has the freedom to vary in order to determine the best engine cycle for a particular application. The four engine design parameters are the overall pressure ratio ($\pi_{Overall}$), the turbine inlet temperature (TIT), the fan pressure ratio (π_{Fan}), and the bypass ratio (BPR). The selection of values for these cycle parameters is dependent on the aircraft mission and design[21]. For example, in the case of a long range commercial airliner, the specific fuel consumption of the engine at cruise conditions is much more important to minimize than that of an engine designed for a military interceptor. This variation in the importance of performance metrics is one primary reason why many different engine cycles are being utilized in today's turbine engines.

Typically, engine performance is determined based on the achieved output (thrust or shaft horsepower), efficiency (TSFC), and weight. The net thrust of a separate flow turbofan is estimated using the following equation. For each flow stream of the turbofan, there are two sources of thrust. The primary source is the momentum flux of the flow, while there is also a secondary source due to pressure when the nozzle flow is not fully expanded to ambient pressure.

$$F_n = \dot{m}_{19}(v_{19} - v_0) + \dot{m}_9(v_9 - v_0) + A_{19}(p_{19} - p_0) + A_9(p_9 - p_0) \quad (1)$$

The TSFC of a gas turbine engine is defined as

$$TSFC = \frac{\dot{m}_f}{F_n} \quad (2)$$

where \dot{m}_f is the fuel flow rate of the engine. The TSFC is an efficiency measure of an engine, measuring how much fuel is required to achieve a given thrust, and permits the comparison of engines[35]. Another useful parameter of an engine is the specific thrust, the ratio of thrust achieved to the total mass flow entering the engine.

The bypass ratio (BPR) of an engine is the ratio between the mass flow bypassing the engine core to the core mass flow. It is defined as

$$BPR = \frac{\dot{m}_{byp}}{\dot{m}_{core}} = \frac{\dot{m}_{13}}{\dot{m}_{23}} \quad (3)$$

Knowing that the total engine air flow is the summation of the two flows, each stream mass flow term can be expressed in terms of the BPR and the total engine air flow.

$$\dot{m}_0 = \dot{m}_{13} + \dot{m}_{23} \quad (4)$$

$$\dot{m}_{13} = \frac{BPR \cdot \dot{m}_0}{BPR + 1} \quad (5)$$

$$\dot{m}_{23} = \frac{\dot{m}_0}{BPR + 1} \quad (6)$$

Neglecting the mass flow contribution to the engine's achieved thrust as well as any mass flow lost, such as customer bleed requirements, the thrust equation can now be expressed in terms of the *BPR* and total engine air flow.

$$F_n \approx \frac{BPR \cdot \dot{m}_0}{BPR + 1}(v_{19} - v_0) + \frac{\dot{m}_0}{BPR + 1}(v_9 - v_0) + A_{19}(p_{19} - p_0) + A_9(p_9 - p_0) \quad (7)$$

In order to show how the bypass ratio impacts the specific thrust of the engine, Figure 8 shows how the specific thrust of an engine changes with the bypass ratio. As the bypass ratio increases, the specific thrust decreases. Instead of looking at the ratio of thrust to the total engine air flow, Equation 6 can be rearranged to define an expression for the total mass flow as a function of core flow and bypass ratio, and this expression can be placed in Equation 7. An expression can then be roughly established, showing the impact the bypass ratio has on the resultant thrust per unit of core air flow. Figure 9 shows this relationship. The thrust per unit core flow increases until losses incurred by a large fan diameter start to overwhelm the available thrust at higher bypass ratios. When considering selection of an engine core design, this offers an interesting way to look at the relationship between the net thrust achieved by an engine and the core air flow. It offers an incentive to search for the engine cycle that maximizes the thrust per unit of core air flow. Particularly, if a core definition is established, the designer should ask how to maximize the thrust for the established core's design airflow levels. Conversely, *what range of thrust levels can be achieved for a particular core airflow, and what kinds of performance compromises are made for each of those thrust level applications compared to an engine core flow sized just for the application under consideration?* For a particular set of engine requirements, constraints, and assumed technology capabilities, there exists an optimal cycle, and compromises will be

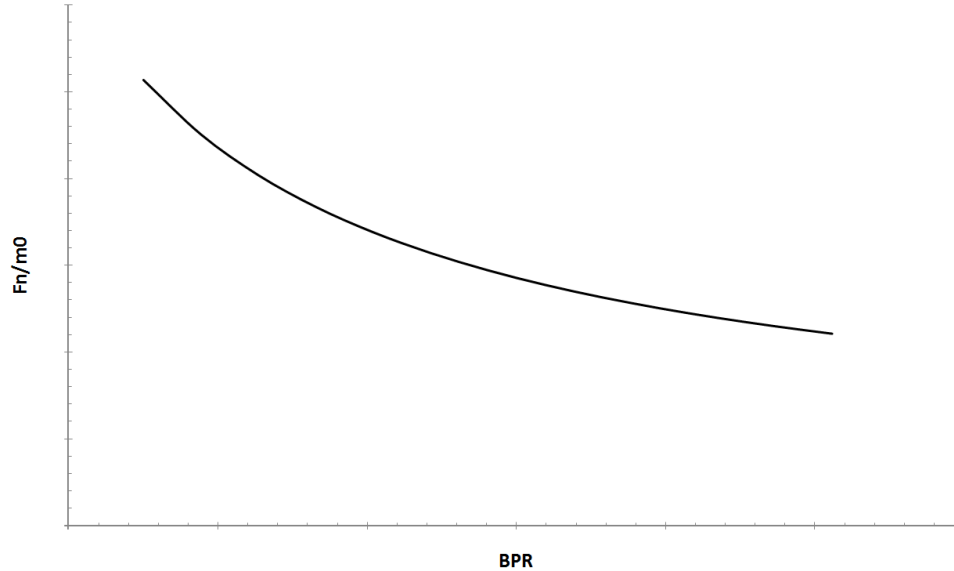


Figure 8: Impact of BPR on Specific Thrust

made when utilizing a common core definition for multiple sets of requirements, constraints, and assumed technology capabilities. A common core application’s low pressure system should be designed to minimize the performance compromises made when employing a common core definition, while remaining within all limits imposed on the design. For a given amount of air flow, Fowler indicates that a key driver in the achievable thrust levels of an engine cycle is the turbine inlet temperature (TIT) rather than the cycle pressure ratio. On the other hand, the TSFC is primarily driven by the BPR while being moderately sensitive to the cycle pressure ratio and weakly dependent upon the TIT[36].

A turbofan engine can be defined in terms of its size, pressure ratio, bypass ratio and specific thrust[24]. The size of the engine is typically thought of as the amount of airflow capacity required, given the selected engine cycle, to produce the required thrust levels at various design points. The pressure ratio, specifically the $\pi_{Overall}$ is determined based on the technology level of the engine and the associated material temperature limits of the compression system. The engine’s bypass ratio is selected that maximizes vehicle performance by increasing the propulsive efficiency, while limiting the losses incurred by large fan diameters. Ideally, the higher the bypass ratio, the higher the propulsive efficiency which

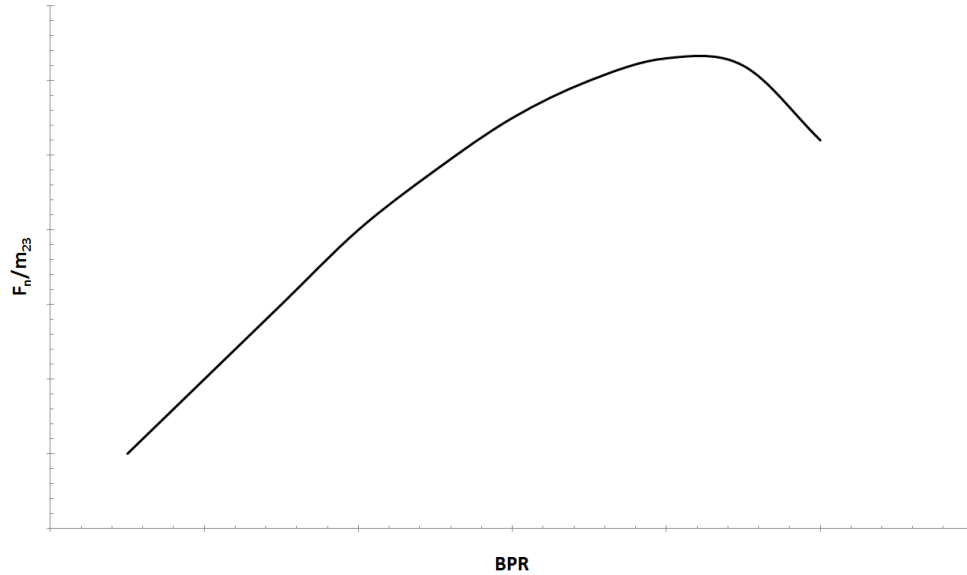


Figure 9: Impact of BPR on the Thrust per unit Core Air Flow

in turn reduces the TSFC. However, an increase in bypass ratio while utilizing a fixed core size causes an increase in fan diameter, which can cause increases in tip losses, ram drag, and nacelle drag. The fan diameter is also usually limited by the placement of the engine under the wing of an aircraft.

There are a wide range of pressure ratios utilized in engines that are in service today. This is due to the many factors considered by designers when selecting their design pressure ratios. These factors include the specific power output required by the engine, achievable loading levels which determine the number of stages required to reach certain pressure ratios and the corresponding number of turbine stages required to extract thermal energy from the expansion flow, driving the compression system, propulsor(s), and/or power shaft. Other factors to consider are temperature levels allowed or desired at various locations in the engine, reliability, initial cost, and operating costs[8]. Another key determiner in the maximum pressure ratios attainable for compressors of various types and sizes are the losses incurred due to size effects and/or speed effects.

2.2.1.4 Design Point Analysis

From an academic perspective, the traditional way to size and evaluate an engine cycle is to perform on-design cycle analysis at a single design point. This design point is usually at a cruise or top of climb condition for commercial engine designs. After performing on-design cycle analysis at the single design point, off-design cycle analyses are performed at various, possibly constraining operational points of interest. These off-design points usually consist of takeoff points, top of climb, various points during cruise, noise certification points, and so on. The off-design analysis ensures that requirements and constraints are satisfied at operational points other than the design point. If all the requirements and constraints are satisfied, then the engine cycle can be considered a feasible candidate design. If all of the requirements and constraints are not met, then a re-design at the design point must be done.

Schutte[102] shows that an engine designed using a single design point requires many assumptions to be made for other operational points, which rarely allows for the selection of the best feasible engine design. Examples of typical assumptions made are engine cooling flow sizes and component temperature levels at various operational points. Using a single design point approach, these assumptions must be made due to the relationship between how an engine designed at one point operates at another point in the operational space. By making these assumptions, Schutte shows that a shift in the feasible cycle design surface, which can be thought of as an unnecessary design margin, is required in order to achieve a feasible design[102]. This shift and the corresponding changes in the shape of the cycle design space can cause the designer to select a candidate engine cycle that may actually perform worse than another candidate solution.

By utilizing a multiple design point (MDP) approach, the designer is able to exactly meet requirements at various operational points, instead of requiring margins to be built into the design resulting in an oversized engine that exceeds various requirements. The MDP approach treats all operational conditions where requirements or constraints are imposed as design points, allowing for the analysis of the engine cycle design to be performed with significantly less assumptions required. This prevents over-sizing of the engine design,

which can result in compromised performance and the selection of an inferior cycle for the application at hand.

The premise of Schutte’s MDP approach will be utilized in the present work, applying it to the design of a common engine core. Instead of designing an engine for a single application and then testing whether requirements are met for other engine applications, multiple common core engine applications will be designed and evaluated simultaneously when exploring and selecting the candidate core design. Consideration of possible requirements and constraints of various applications, analogous to the consideration of multiple design points in Schutte’s work, allows the designer to arrive at a common engine core design requiring less assumptions to be made in accounting for future engine applications.

2.2.1.5 Engine Cycle Sizing, Simulation, and Selection

The engine designer must visualize the cycle design space (CDS) in order to arrive at a design engine cycle that will offer the best performance for the end user. This visualization is referred to as cycle design space exploration. The predicted cycle performance is plotted against various cycle design parameters. Typically, a performance metric such as the predicted TSFC at a certain design condition is plotted against the design values of the overall pressure ratio ($\pi_{Overall}$) and fan pressure ratio (π_{Fan}) cycle parameters, allowing for a three-dimensional representation of the CDS. The shape of the surface generated is based on the performance requirements while the feasible boundaries are based on the technical constraints applied to the otherwise unconstrained design space[103].

Construction of the CDS requires many evaluations of candidate engine cycle designs over a range of cycle parameters. Each cycle evaluation is an iterative matching procedure[107] that, when performing numerous cycle evaluations, can become very computationally expensive. NASA created an object-oriented program with a built-in Newton-Raphson solver that assists in modeling the performance prediction of gas turbine engine cycles, called the Numerical Propulsion System Simulation (NPSS)[25, 65, 66, 63, 64, 82, 81, 83, 49]. The Newton-Raphson solver is a nonlinear root finding algorithm, the goal being to converge on a specific set of conditions[98][120][5]. Input variables are intelligently varied using derivative

data until convergence upon the root of the function being evaluated is achieved.

Parametric vehicle studies can be beneficial when performing engine cycle selection. The size of a parametric vehicle is determined based on the mission analysis of the vehicle with the selected engine cycle being “flown” on board. By allowing a vehicle to resize based primarily on the predicted fuel burn required for the design mission, the required thrust levels of the vehicle can be updated as well. In the design of an entire vehicle, this can allow for additional fuel burn savings, due to the possibility of having a smaller airframe and in turn a smaller engine than initially selected[37]. If this sort of design freedom is granted to the engine designer, then parametric vehicle studies should be performed. As previously mentioned, the engine requirements greatly determine the shape of the CDS. Therefore, if engine resizing is allowed, then parametric studies must be performed in order to arrive at the engine cycle definition that will perform best at the size required[52].

Proper visualization of a design space offers the designer the ability to relate design parameter settings directly to corresponding performance levels. This task can be trivial when exploring a design on very few input variable dimensions and when a single metric is used in design evaluation. However, as the number of input dimensions are explored as well as the number of metrics used in the evaluation of candidate designs, visualization of a design space can be quite difficult. Three visualization techniques are used in the present work to offer the designer the ability to quickly evaluate favorable regions of the design space.

The first two techniques are sample-based visualization techniques, where plots are constructed that show where discrete design samples lie in the input- and metric-space. Multivariate scatterplots display pairwise scatterplots of variables on a single view in a matrix format[112][48]. Each row and column represents one dimension of interest. The sampled data points are placed in each pairwise view. The multivariate scatterplot is very powerful, particularly when viewed within a software package that allows for the dynamic selection of candidate design points allowing the designer to see where the design lies in all input/output dimensions considered.

Another sample-based visualization technique displays the same information as the multivariate scatterplot, but in an alternative fashion. Parallel coordinate charts consist of various axes of interest, and line segments representing sampled designs are drawn across these axes at its input or estimated metric values[44][45][79]. These plots are extremely powerful when analyzing designs based on many input and output dimensions. Trends can be easily identified across many dimensions, showing the designer favorable design regions within a high-dimensional space.

Whether constructing the engine cycle design space for a fixed vehicle or performing parametric vehicle studies, the designer gains understanding of the effects design requirements and constraints have on the engine cycle design and corresponding engine size required. Numerical optimization methods allow the designer to arrive a feasible engine cycle optima within the highly nonlinear multidimensional design space that performs best for of a given metric[58]. A multi-objective optimization can also be performed, allowing the designer to identify pareto optimal designs for a given engine design problem[95, 76]. Optimization schemes can also be utilized in collaboration with an organization's technology development program in order for the engine designer to arrive at an engine cycle and corresponding technology package which will allow for the best performance improvement while minimizing risk[54, 94, 55].

For a common engine core design, the selection of the core definition must be made while simultaneously considering a wide range of capabilities an engine program aims to provide competitive solutions for. For each individual engine application, many requirements and constraints will likely exist. These must each be accounted for in the design of a common core. Otherwise, limiting requirements or constraints will not be identified, causing possible required design mitigation after initial downselections have been made, costing the designer both in the development capital required, and also in the performance levels achievable when utilizing the common core across various engine applications. Additional consideration of possible technology improvements over the life of a program, in addition to various application requirements while exploring and evaluating candidate core designs will allow for the selection of a more versatile engine core design that can take advantage of technology

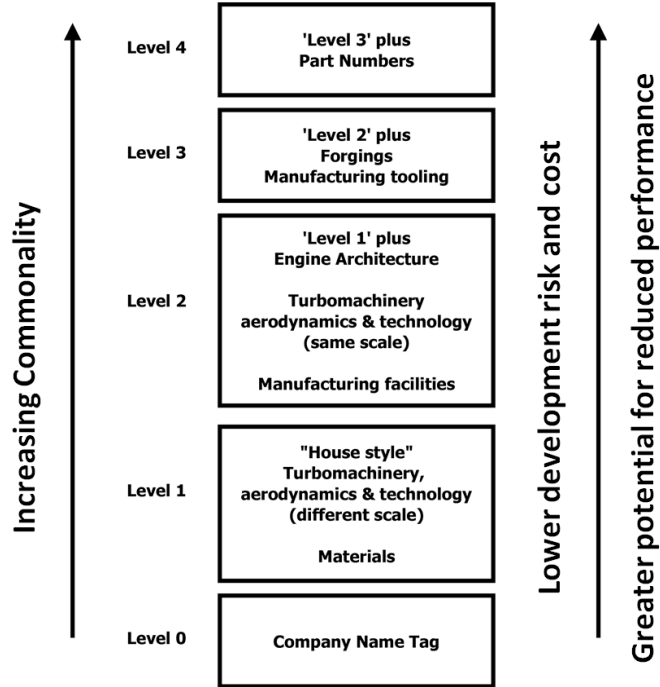


Figure 10: Engine Program Commonality Levels[15]

developments over time and provide more competitive engine applications over the life of the engine program than otherwise would be able to be attained.

2.2.2 Development of Engine Variants

A successful commercial gas turbine engine program is one that offers a wide variety of competitive products. By distributing their development costs among a range of engine applications, the company can offer their products with competitive pricing. However, customers will not buy a product that does not perform competitively. Therefore, in common core engine family design, there is a tradeoff between the distribution of the development cost among a variety of engines and the performance compromises made by employing a common engine core across the family. Bradbrook[15] explains the advantages and disadvantages of the different levels of commonality in an engine program, which is depicted in Figure 10

Similar lessons can be learned from other industries where families of products share common parts or components in order to share capital risk, but at the expense of attainable

performance levels. Simpson[104] established a product variety tradeoff chart, where designs are placed on the chart based on two indices: the performance deviation index (PDI) and the non-commonality index (NCI). The NCI assesses parametric variation within a product family using a normalized measure of the variability of design variable settings across members of a product family. The NCI is essentially a weighted sum of design commonality across a family of products, as shown in Equation 8. The PDI provides a measure of how well products making up a family meet their individual performance targets, consisting of a weighted sum of all products' performance with respect to more ideal designs that are designed for each individual application, as shown in Equation 9. Minimal PDI values are always desired for a product family, indicating their performance levels lie close to ideal, individually optimized designs. Low NCI product family designs are also desired, indicating a large amount of commonality exists between the products that make up the family. The product variety tradeoff chart that Simpson presents has been recreated in Figure 11.

$$NCI = \sum_{i=1}^{N_{apps}} \left| \frac{w_i(x_{i*} - x_{common})}{x_{i*}} \right| \quad (8)$$

$$PDI = \sum_{i=1}^{N_{apps}} w_i(y_{i*} - y_i) \quad (9)$$

Efforts have been made to optimize a product family design. Fellini et al.[34] presents an approach to making product family platform selections based on the performance deviations of each product from its corresponding ideal independently optimal design. By minimizing the overall weight sum of the performance deviations from each corresponding optimal performance level, a product platform design selection can be optimized within a region a feasible commonality in order to improve the overall product family performance. Figure 12 is a recreation of a chart the authors present defining the performance deviations of each product family member from their corresponding ideal designs if no commonality were present.

Combining Simpson's efforts to measure commonality and the performance deviations across a product family with Fellini's efforts to measure performance deviations, Figure 13

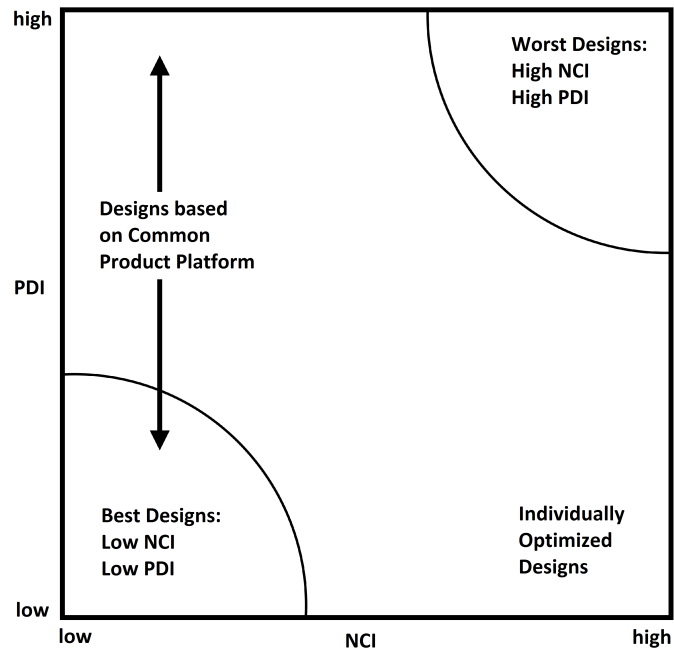


Figure 11: Product variety tradeoff chart[104]

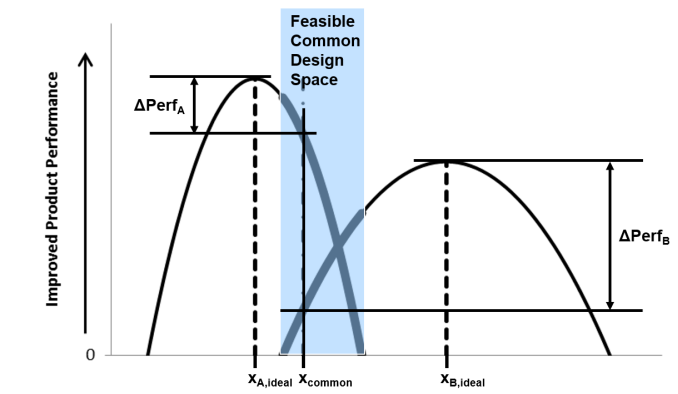


Figure 12: Performance deviations due to commonality[34]

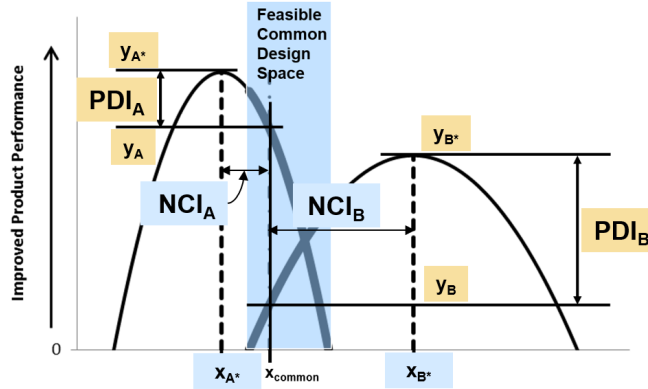


Figure 13: Simpson’s NCI and PDI measurements visualized on Fellini’s chart of performance variation across design settings for various design applications.[104][34]

displays Simpson’s measurements on Fellini’s plot of performance levels across various design variable settings across products.

The various ways of measuring commonality as well as the measurement of performance compromise made when accepting a certain level of commonality offer a way of pushing common core designs away from unique engines for each unique set of requirements and towards a more common engine between applications. The various approaches also show a need for a design parameter whose setting should be selected to maximize the performance of a product line while also maintaining a common parameter value across product applications. For a common gas turbine engine core, the question is: *What engine parameter should serve as the parameter that maintains its value across all common core engine variant applications in order to ensure commonality across applications?* Lehmann[61] provides the answer to this question. By maintaining a common value of the corrected compressor exit flow (W_{3R}) at some design operating condition, commonality can be ensured across common core engine applications. Lehmann actually considers possible common core applications across different aerospace vehicle types where there is overlap in the compressor exit corrected flow values that offer feasible solutions for different engine application types, as shown in Figure 14.

It is well established that for a unique set of requirements, a completely unique engine design would offer the best performance, under an assumed set of technology limits and

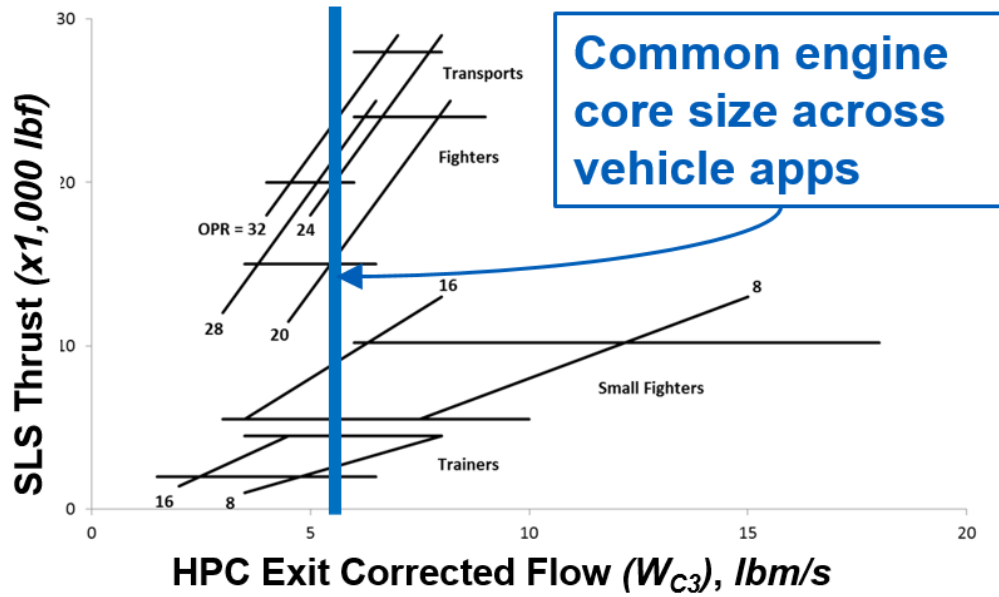


Figure 14: Core size requirements for multiple types of engine applications[61]

capabilities. The design freedom of a new centerline engine offers the best ability to tailor every component of the engine design to the specific set of requirements that the resultant engine design must satisfy. However, completely unique engine designs are rarely utilized due to the massive development cost required. The three factors that impact whether a new engine is designed or a derivative engine is made for a new set of customer requirements are timing, cost, and performance[29]. If time is of the essence, a simple derivative that can meet the requirements with a rating change of a current engine may be the best solution, although significant performance compromises would probably be made. If the customer wishes to have the best performance possible and is willing to pay substantially for it, then a completely unique design may be warranted. Although this is rarely the case in today's increasingly strained economy.

Derivative, or variant engines of an existing engine program often offer a convenient compromise between the performance achievable and the associated cost of development. It has been shown that with a minimal amount of modifications, an existing design can offer

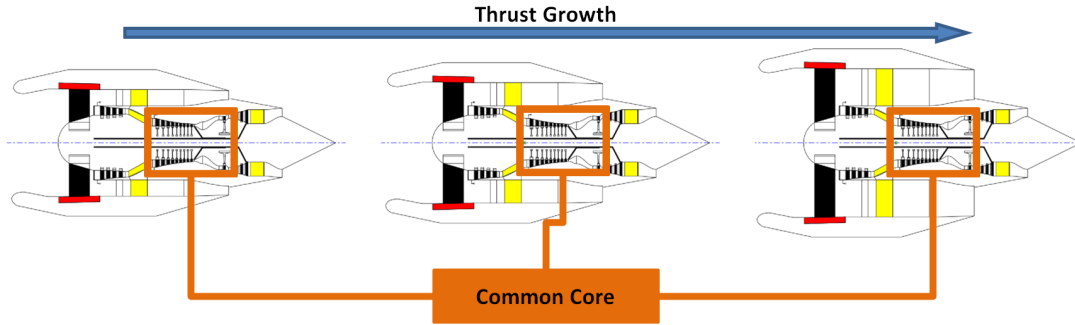


Figure 15: Common Core Flowpath Variety

increased capabilities in performance, such as was seen with the F100 program[56]. Performance improvements can be applied to existing systems, taking advantage of the latest aerodynamic and materials technology in an existing engine system[57]. In reality, engine programs continuously improve upon their existing engine fleet. In most cases, an engine program's foundation is in the common engine core architecture and size. The engine core flow areas and stage counts typically remain fixed throughout an engine program. Performance improvement packages, otherwise referred to as technology infusions are applied to the common core modules. These packages take advantage of material and aerodynamic improvements that the latest technology allows for. The packages are restricted to the same engine core size and architecture in most cases.

The low pressure system of the gas turbine engine is where significant design modifications can be made if necessary. Completely new fan designs can be utilized, in addition to LPC and LPT design updates, while care is taken to ensure that the design updates will match well with the common engine core. It must be verified that the core gas generator will be able to supply ample power to any low pressure system being used with the common engine core. Figure 15 shows notionally a variety of flowpaths achievable with a common engine core. With the design freedom of the low pressure system, a range of thrust levels can be achieved with a common core engine program. The question that arises is how should the core size and architecture be selected in order to perform well for a wide variety of engine program applications?

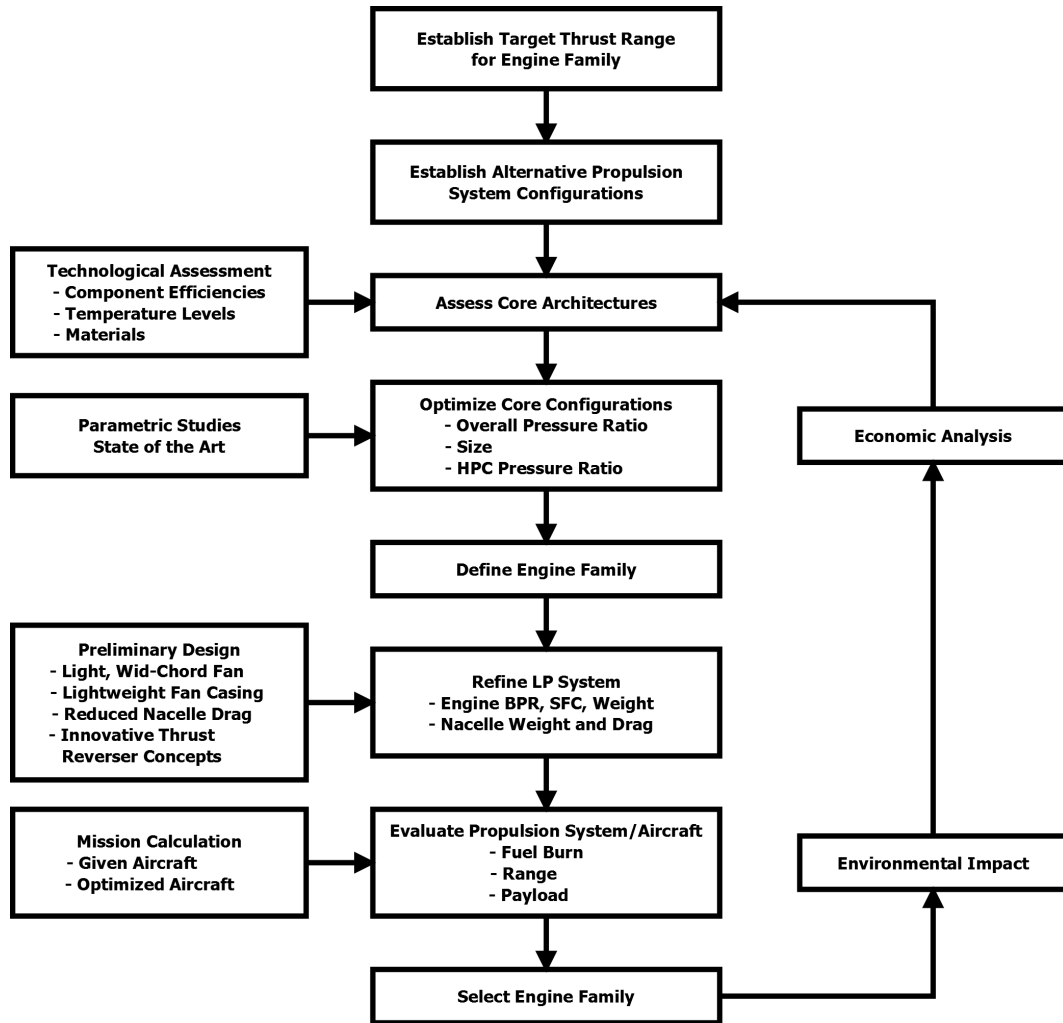


Figure 16: Engine Family Selection Process[46]

Jacquet and Seiwert[46] present a process for analyzing and selecting an engine family, which is shown in Figure 16. This method will serve as a basis for the proposed methodology because it evaluates the entire engine family when considering the selection of a common core architecture.

2.2.3 Development Cost Considerations

While one of the primary goals of an engine program is to minimize the overall development cost, it is one of the most difficult measurements of the engine program to predict, especially without published figures on the actual costs of existing engine programs. However, there

are obvious relationships between the cost of producing an engine and the engine's associated size and cycle. For example, if a gas turbine engine has an extremely high OPR, then it is able to operate at higher temperature than engines with lower achievable OPRs, indicating that significant investment was made on the core materials technologies in order to operating at high sustained pressures and corresponding temperatures. At the very least, changes in the design cycle parameters and size of an engine may be a good indicator of associated changes in the development cost of an engine design. The goal is not to estimate the cost of an engine to a dollar amount, but rather to place candidate engines on a cost scale, allowing for a rough comparison to determine that one engine would be more expensive to develop than another.

Collopy[22] developed a cost metric named *surplus value*, which aimed at predicting the total profit of the airline, airframer, and engine manufacturer. He used this metric in a competition model to characterize interactions between competitors in different market types. Younghans et al.[119] describe various techniques for propulsion system cost modeling. Parametric techniques use statistical relationships from historical data, and have been shown to be generally valid within a small range of technologies. Bottom-up techniques build up the estimated cost of a system based on each development operation's estimated cost. Comparative techniques adjust cost based on component size, material, configuration, and feature changes. Younghans et al. also applied a variation of Collopy's surplus value method to select an engine cycle for performance and economic factors[119]. Another way to relate the performance and cost of an engine is to consider a capability-to-cost index, which the Versatile Affordable Advanced Turbine Engines (VAATE) program aimed to improve upon[108].

From an academic perspective, the drawback to most cost estimation techniques is just that there does not exist enough public data in order to accurately estimate the cost of even a current system, let alone a future one. However, simple comparisons between engine size, capabilities, and operating temperatures and pressures will offer a way to place candidate designs on a very simple cost scale. The primary goal of this current work though can be achieved independently from this limitation. A common core engine program that offers a

wide variety of current and future competitive capabilities will allow for the development cost of the program to be distributed across a wide number of products. This on its own will presumably reduce the cost of a single engine unit produced by a common core engine program.

2.2.4 Probabilistic Design Methods

There are many sources of uncertainty in the initial development of any product. Probabilistic design methods aim to account for different sources of uncertainty that affect the final design. This aids the designer when faced with making a design decision. By identifying sources of uncertainty, it not only helps the designer make sound decisions that will avoid future problems, but it also allows him to select a robust design: one that achieves the requirements while remaining insensitive to the various sources of uncertainty. Robust and probabilistic design aims to improve overall performance by determining: required design margins, impacts of design parameters on performance uncertainty, and means of reducing the impacts of uncertainty[72].

Within the context of commercial engine design, Roth and Mavris[96] identify five sources of uncertainty when developing a comprehensive engine uncertainty analysis environment. These sources and their associated impacts include:

- Impact of Mission Requirement Uncertainty on Engine Size
- Impact of Cycle Uncertainty on Vehicle Performance
- Impact of Aircraft Design Uncertainty on Engine Size
- Environmental and Regulatory Uncertainties
- Technology Uncertainty/Risk and Analysis Method

Traditionally, design margins would be built into systems in order to ensure that the system could provide technically feasible solutions. Sands et al.[100] shows that the feasible engine design space can disappear if optimistic assumptions on component performance are made, while realized performance levels are lower than originally expected. A size margin applied to an engine design allows for feasible options in the event that performance levels

are not as high as originally estimated. However, without knowing how large of a margin to build into an engine system, the designer may oversize a system and pay an unnecessary performance and weight penalty. On the other hand, not allowing for a large enough size margin could completely eliminate the ability to provide a feasible solution at any performance level. There is a crucial need for enumerating the major sources of uncertainty and accounting for these sources in the initial design phase of an engine program.

In order to account for the major sources of uncertainty, probability density functions (PDF) must be constructed representing the variability in an impact, and the distribution of impact values should be applied to each source's associated input variables. The PDF, shown notionally with a normal distribution in Figure 17, indicates the relative likelihood for a variable to take on a given value. The required thrust an engine may need to achieve can be treated as a probability distribution of thrust values that may be required. Instead of designing and evaluating an engine at a specific thrust requirement, the engine design should be evaluated based on a distribution of possible thrust requirements, and a corresponding cumulative distribution function (CDF) should be constructed[97]. The CDF then allows for the designer to determine the probability of achieving a requirement, in this case the required thrust of the engine design. A CDF of the notional PDF previously constructed is shown in Figure 18. The plot shows that there is roughly a 15% chance that the value of x will take on a value less than or equal to 9.

A more robust design can be achieved by accounting for the various sources of uncertainty and making design decisions with the use of probabilistic design methods. In order to account for all the sources of uncertainty simultaneously in the design process, a Joint Probabilistic Decision-Making Method (JPDM) can be employed[7][96]. JPDM techniques evaluate designs based on the probability of satisfying all criteria at the same time.

Probabilistic design methods can assist the designer in selecting a robust design that performs well and is insensitive to the uncertainties inherent in a new product line. By evaluating a common engine core design over a wide range of possible requirements and under the major assumed sources of uncertainty present in the multiple application design problem, the designer can be confident that the design selections made will result in a

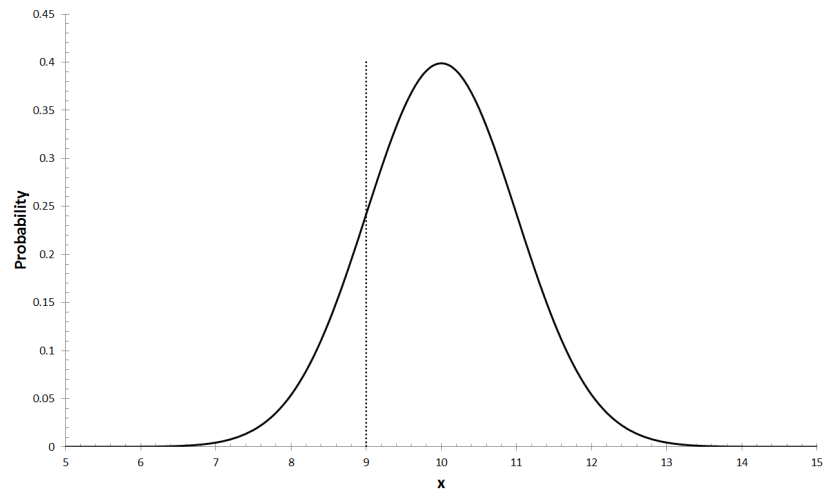


Figure 17: Probability Density Function

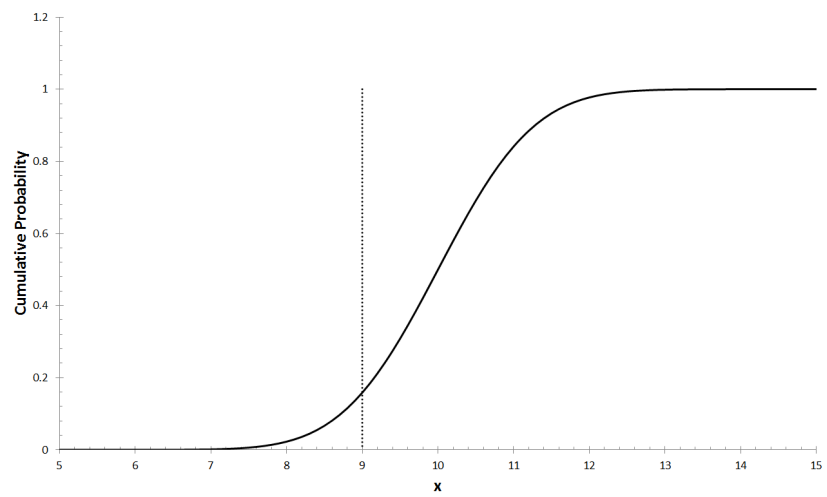


Figure 18: Cumulative Distribution Function

common core that performs well over the range of requirements. By evaluating a common core design over a range of technology limits and corresponding distributions of their likely impacts, the designer is also able to consider engine applications over time as technology matures. Considering possible requirements and technology capabilities for near- and late-term applications will enable the designer to select a common core design that best meets all requirements imposed, resulting in competitive solutions throughout a wide range of capabilities, both initially and well into the future.

2.2.5 Enhanced Robust Design Simulation for Gas Turbine Engine Design

The goal of any design is to incorporate performance and features that are desired by the customer, while also ensuring a quality, cost-competitive product[28]. The robust design process aims to maximize the likelihood of selecting a quality candidate design that satisfies all performance requirements while minimizing risk due to uncertainty. In other words, design parameter settings are selected that achieve mean values of estimated performance levels within an acceptable region, while simultaneously minimizing the negative impacts due to sources of uncertainty affecting the design's performance. Sources of uncertainty could include, but are not limited to, technology impacts, component performance, cost levels, design requirements and risk[70],[110],[96].

The traditional robust design process is outlined by Mavris et al.[71]. The authors' work provides an outline for the determination of robust design solutions and the rationale for quantifying and mitigating the effects of uncertainty early in the design process. Monte Carlo analysis is used to evaluate candidate designs, estimating the probability of meeting requirements. In this process design performance estimation is repeated many thousands of times, allowing for the designer to observe how variation in noise parameter settings affect the estimated performance of a candidate design. Likewise, variation in the design parameters for a given set of noise parameters can be used to examine what design is robust to design noise such as technology uncertainty[51]. Impacts of noise variable distributions have also been represented through Monte Carlo sampling and extended to economic and business metric variance estimation[62]. It should be noted that a robust design is different

from a deterministic, optimal design. An *optimal design* performs the best for a given set of metrics assuming a fixed level of technology benefit. A *robust design* may not have the outright performance of the true optimum; however, the robust design should be more insensitive to changes in noise variable settings. A robust design may be more desirable from a cost perspective since it reduces the need for costly redesign and delays should the design fail to meet required metric values.

In order for the designer to adequately explore the candidate design space and perform the required Monte Carlo simulation of each candidate design to achieve meaningful results, a substantial number of computer simulations perturbing design and noise parameters are required. The amount of computational time required to perform the simulations using even a conceptual level physics based model limits the amount of candidate designs that can be explored and/or the number of Monte Carlo repetitions representing the uncertainty present in the design[62]. A significant amount of work has been performed, aiming at overcoming the computational challenge present in probabilistic design, or design under uncertainty.

Du and Chen[30] attempt to increase the efficiency of probabilistic design with the use of a sequential optimization and reliability assessment method. With the goal of implementing probabilistic design while overcoming the computational burden of uncertainty analysis, Du and Chen break up the design process into two main steps. The first step is to perform a deterministic optimization, making design selections which are then further explored. For a deterministic design selection, they then perform reliability analysis, where impacts of uncertainty on the reliability are estimated. The authors indicate that a majority of the computations required are attributed to the reliability analysis. This essentially reduces the number of design alternatives that can be explored, potentially causing the designer to overlook the global optimal reliable solution.

Another method for overcoming the computational cost of accounting for uncertainty through Monte Carlo sampling is the elimination of sampling altogether. Chen et al.[19] present generalized analytical formulations that can assist with global sensitivity analysis of a variety of popular metamodel types. By performing global sensitivity analysis, the designer is able to screen out variables that provide negligible amounts of variability in

the metamodel response. Their work considers the main effects and first order interaction effects of the control and noise variables considered, eliminating variables from future analyses that do not contribute significantly to the uncertainty in the metamodel response. Only considering the main effects and first order interactions between variables may cause the designer to overlook certain variables and corresponding regions of a design space when aiming to make robust design selections, specifically within the regime of the highly nonlinear behavior of the aerothermodynamic engine models employed in the present work. The Enhanced Robust Design Simulation (ERDS) method, a key enabler of the larger method presented in the next chapter aims at reducing the computational burden of design under uncertainty without the need to reduce the dimensionality of the design problem considered. While Monte Carlo sampling of the surrogate model employed is still required of the ERDS method, the probabilistic information gained from the Monte Carlo sampling is then utilized to allow the designer to perform significantly more probabilistic studies without the need for further Monte Carlo sampling.

It has been shown that surrogate model representations of the physics based model can be constructed, greatly reducing the amount of time required for a single simulation from seconds to milliseconds, without a significant reduction in the accuracy of the model[54]. Surrogate models taking the form of Artificial Neural Networks (ANN) have been shown to accurately represent highly non-linear physics based models of gas turbine engines. This model form was first explored in the 1940s aiming to represent biological processes, achieve artificial intelligence, and allow for machine learning[73][39][33][90][93].

ANNs consist of *nodes* and *arrows*, resembling the interconnected network of neurons found in the brain. These mathematical representations are very flexible in their overall form and allow for a significant number of input variables to be explored. Training of an ANN results in a closed-form equation to predict some output metric value based on the input variable settings. No functional form is assumed, allowing various ANN architectures to be explored for the given regression. These models can handle both discrete and continuous input variables and can be used in the estimation of a continuous metric value or in the classification of a set of input variable values. A generic graphical representation of an ANN

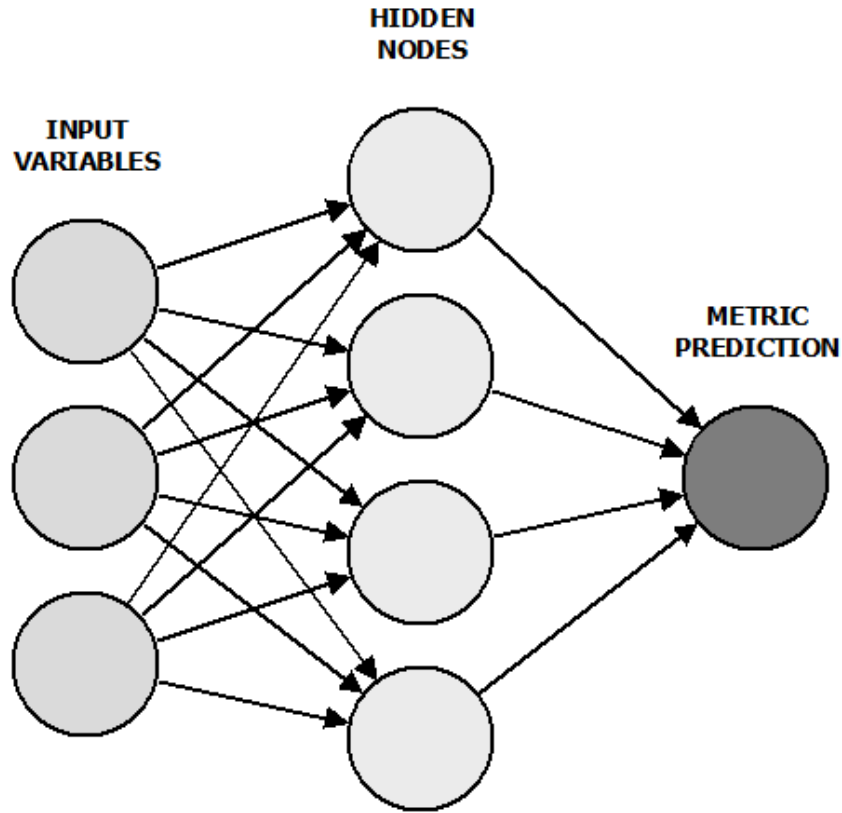


Figure 19: Generic form of an Artificial Neural Network (ANN) surrogate model.

is shown in Figure 19. Each node has a corresponding activation function taking the form of a step, linear, logistic, or hyperbolic tangent function. Each activation function contains weighted sums of the connections present as well as an intercept term. When training an ANN, the values of the weight factors and intercept terms present in the network are converged upon that offer the best model fit quality. Due to the high nonlinear form of the ANN model, when evaluating candidate ANN models great care must be taken to ensure that the model achieves minimal levels of model fit and model representation error, assuring accurate model prediction within the entire input space considered.

For problems where classification of input sets is desired, Support Vector Machine (SVM) models can distinguish between various types of data[23][4]. Given a set of training data each identified as belonging to one of two categories, a SVM model can be trained to classify future data. The more recent kernel-based SVMs have the ability to perform nonlinear

classification with great accuracy[89][14] The goal of SVM training is to identify training points that are closest to the boundary between the two categories of data. These points are referred to as support vector machines. An adaptive sampling approach can be used to improve upon the quality of classification of an SVM model. Identification of training data points that are close together but are contained in separate categories can be performed. New training samples are generated that lie between these data pairs. With the increased set of sample data, another training of the SVM model can be performed, usually resulting in an improvement in the quality of the SVM model.

The training of surrogate models requires a batch of simulations to be performed, whose input and output data is representative of the design space within the input variable regions sampled. Typically, a limited number of simulations can be performed and evaluated with the physics based model. Therefore, intelligent sampling of a multivariate design space can be done with the use of a Design of Experiments (DoE). These strategies enable the most information to be gained about the design space with a limited budget of design samples allowed[116][77]. DoE strategies began in the agricultural community, aiding farmers in determining which combinations of soil, seed types, fertilizers, etc. would offer the highest yield. It would not be viable to explore every possible combination of variable settings. For a given set of variables, DoEs offer an efficient sampling of any variable space. With the limited set of input/output data can then be used to train and test the resultant fit quality of a surrogate model[88].

Latin Hypercube Sampling (LHS) is a specific DoE sampling strategy, known as a space filling design, aiming to gain as much information as possible from a limited set of model training samples[74][31]. This DoE sampling technique lies within a group of techniques referred to as *space filling designs*. For a given budget of sample points, the technique locates each sample on its own axis-aligned hyperplane. LHS ensures that the input space is sampled evenly and that the sample data is representative of the real variability present in the system being sampled[43][42].

The use of surrogate models enables a designer to perform significantly more design simulations within a given computational budget. Probabilistic assessments can now be performed throughout the design space, where repetitions of design simulations are performed under the assumed uncertainty present. The designer is now faced with an overwhelming amount of probabilistic data for each candidate design considered. This traditionally requires a significant amount of post processing to draw any useful information from the large set of probabilistic data. The designer is then able to arrive at probabilities of each candidate design being able to meet its requirements under the assumed uncertainty distributions of the noise variables. In the traditional Robust Design Simulation (RDS) approach, as described in by Mavris et al.[71], the user must repeat the Monte Carlo analysis any time the specific scenario of uncertainty distributions are even slightly changed. Once the Monte Carlo analysis is repeated, the user must then create joint probability distributions of each metric relative to the inputs in order to identify robust solutions. While this process can be scripted and sped up using various computing methods, the problem remains that the designer cannot easily see the probabilistic impacts of noise variables due to changes in uncertainty assumptions. This method restricts probabilistic assessments to be performed for a single fixed set of uncertainty assumptions.

Enhancements were made to the traditional RDS method, aiming to enhance the traditional filtered Monte Carlo robust design approach by regressing surrogate responses of joint confidence intervals for metric responses of interest. The regression model associates a performance estimate with a corresponding desired level of confidence by implicitly accounting for the effects of the uncertainty distributions. This allows for efficient resampling of the design space without the need for repetitions of the Monte Carlo simulation enabling more informative exploration of the design space. Because the development of the ERDS method was led by the present author and because the method plays a central role in allowing the COMMENCE method to be carried out in a computationally efficient manner, details of the ERDS method will be further discussed in the following chapter. The process will be described in detail and will then be applied to a case study considering the robust design of a two-spool axi-centrifugal turboshaft engine cycle[99].

2.2.6 Bayesian Belief Network for Engine Upgrade Option Selection

The present author also took effort in a design problem where consideration of various turboshaft engine upgrade options was made in response to various sets of shaft power growth and ESFC reduction requirements[50]. The ERDS method was applied in order to estimate the achievable performance levels of discrete upgrade options and corresponding continuous cycle explorations within each option's allotted design freedom. Five upgrade options were considered, from the option of a fixed engine that was allowed to be throttled being the least costly option all the way up to the option of designing a new engine with technology infusion being the most costly option. An approximate inference Bayesian Belief Network (BBN) model representation of the design problem was constructed[18][17]. One strength of the BBN is its ability to easily update and integrate probability distributions as evidence is observed in a design problem. An updated distribution is referred to as posterior, or marginal distribution. These distributions allow for input variable settings to be assigned a value, or preference. These are particularly helpful in making design selections based on less concrete preferences, such as cost, instead of evaluating candidate designs purely based on their performance estimates. Within the gas turbine field, some of the first applications of Bayesian networks were used for gas path and fault diagnostics[67][92][60].

For the engine upgrade option study performed by the present author with the use of a BBN, prior distributions were utilized to place preference on lower cost upgrade options as well as on lower design turbine inlet temperature (T_{41}) level design selections. A fixed technology package was considered for possible technology infusion. For each upgrade option considered, the design T_{41} and design overall pressure ratio ($\pi_{Overall}$) could be explored. Sampling of the design variable settings were based on the prior belief distributions, meaning that the less expensive upgrade options and lower design T_{41} designs were sample more often than more expensive design options. After sampling the BBN model representation of the design problem, three sets of surrogate models were used to predict the likely performance of the five upgrade options:

- Prediction of fixed engine performance

- Prediction of a fixed core / new LP system design performance (with and without technology infusion)
- Prediction of new engine performance (with and without technology infusion)

The functional nodes of the BBN used conditional logic to execute the pertinent surrogate model for each upgrade sample, and a large requirements space was considered. A two-dimensional grid of requirements was constructed. One axis of the requirements grid set the amount of ESFC reduction required of the engine upgrade with respect to the baseline engine definition. The other axis of the requirements grid set the amount of shaft horsepower growth required of the turboshaft engine upgrade. For each set of ESFC and power growth requirements, the various design options were sampled using the prior belief distributions, and the upgrade option was displayed that offered the highest likelihood of meeting the requirements, given the prior distributions. In other words, the option that had the greatest number of samples that met the imposed requirements was determined to be the most viable option for that particular set of requirements.

This engine upgrade study demonstrated the advantages and capabilities of the ERDS method that enabled this probabilistic study to be performed in an efficient manner with minimal computational burden. It also showed how a Bayesian Belief Network model can be used for robust design problems, offering similar sampling capabilities of Monte Carlo analyses. The BBN model also allowed for the implementation of prior distributions to be used in the sampling technique, integrating subjective cost-related factors into the otherwise quantitative evaluations of various design option responses.

2.2.7 Multi-Objective Optimization of Engine Designs

For most gas turbine engine applications, the designer is interested in arriving at an engine design that performs well for a variety of figures of merit. NASA's Environmentally Responsible Aviation (ERA) project, for example, aims at the simultaneous reduction in vehicle fuel burn, noise, and NO_x emissions in the N+2 time frame. Table 5 shows the NASA subsonic transport system level goals for various time frames.

A method for measuring how well a candidate design performs with respect to multiple

Table 5: NASA Subsonic Transport Vehicle Level Metric Goals[109]

Technology Benefits	Technology Generations (Technology Readiness Level = 4 - 6)		
	N+1 (2015)	N+2 (2020)	N+3 (2025)
Noise (cum below Stage 4)	- 32 dB	- 42 dB	- 71 dB
LTO NOx Emissions (below CAEP 6)	-60%	-75%	-80%
Cruise NOx Emissions (rel. to 2005 best in class)	-55%	-70%	-80%
Aircraft Fuel / Energy Consumption (rel. to 2005 best in class)	-33%	-50%	-60%

figures of merit is the Technique for Order Preference by Similarity to Ideal Solution (TOPSIS), first developed by Hwang and Yoon[41][118][40]. Evaluation of candidate designs are performed by first establishing a positive ideal solution and negative ideal solution on all metric levels considered when evaluating a design. The euclidean distance in the metric space from each of the ideal solutions is then determined for each candidate design. The non-dominated candidates that lie on the pareto front can all be considered to be optimum designs, depending on the preference of one metric over another. Figure 20 shows a notional pareto front of candidate designs and their measurements from the positive ideal and negative ideal solutions. Note that the figure shows a pareto front where improvement in both metrics is achieved by increasing the metric value. For example, two metrics typically considered in aircraft design are fuel burn reduction from a baseline and noise margin below a cumulative noise limit. Equal preference can be placed for each figure of merit, or weightings can be applied to each figure of merit based on the relative importance of maximizing performance of certain metrics. The weighted separation of candidate design i from the positive and negative ideal solution considering M metrics can be calculated using the Equation 10 and Equation 11, respectively.

$$S_i^+ = \sqrt{\sum_{m=1}^M \left[\frac{1}{w_m} (y_{m,i} - I_m^+) \right]^2} \quad (10)$$

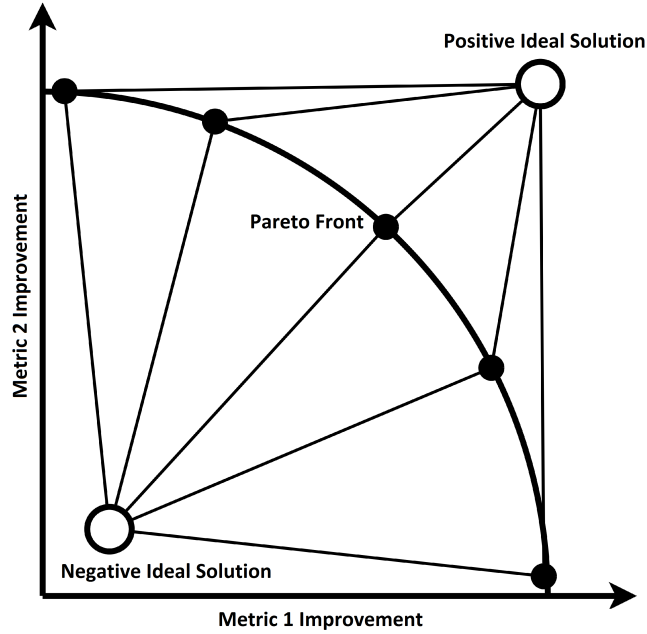


Figure 20: Pareto Front and Corresponding TOPSIS Measurements

$$S_i^- = \sqrt{\sum_{m=1}^M [w_m (y_{m,i} - I_m^-)]^2} \quad (11)$$

The relative closeness to the ideal solution, where 1 is the best and 0 is the worst, can then be determined based on the separation from the positive and negative ideal solutions, as shown in Equation 12.

$$C_i = \frac{S_i^-}{(S_i^+ + S_i^-)} \quad (12)$$

There are many methods of numerical optimization that exist to improve upon designs. All aim to provide a reduction in design time. The designer can explore many design variables and constraints without bias, while virtually always yielding at least some design improvement[113]. However, there are limitations to numerical optimization. Computational time increases with the number of design variables considered. Also, the results of an optimization can be misleading and there is rarely a guarantee of a resultant global optimum design[113].

Optimization methods have been applied to gas turbine designs in the past. Optimization of an axial compressor gas path was done to maximize efficiency[87]. A gradient method of optimization has been applied to the cycle selection of a derivative turbofan engine with a given core[58]. Turbofan engine designs have been optimized based on multiple objectives for a specific aircraft[11]. Engine core designs have even been optimized for a predefined engine family aiming to provide solutions optimized on a technological and economic basis[46]. Multidisciplinary considerations have also been made when considering engine design decisions, estimating the impacts of disciplinary uncertainty on design robustness in terms of the changes in performance metric variance and a design's probability of success[26].

Surrogate modeling is a helpful way of capturing the relationships between the metric responses designers are interested in and the input variables the designer has the freedom to change, allowing for the computational burden to be greatly reduced when performing assessments that required many candidate designs to be evaluated. By performing a Design of Experiments (DoE), a population of unique design simulations can be used to adequately represent the design space the designer is interested in exploring. Intelligently selecting the values of all required input variable settings for each member of the representative population, limited simulations can be used to produce surrogate models in order to explore and optimize a particular design for multiple objectives without the need to perform costly simulations for every unique design considered. Artificial Neural Networks (ANN) have been shown to be a promising type of surrogate model that can capture the highly nonlinear nature of gas turbine engine performance[54] while also allowing for a significant number of input variables to be explored. An optimization method can be utilized with surrogate models of the figures of merit the designer is interested in, and the method can arrive at a unique design that, at the very least, improves upon the overall system performance.

CHAPTER III

TECHNICAL APPROACH

This chapter aims to establish the overall method and enabling capabilities developed for integration into the Common Engine Core Evaluation (COMMENCE) design framework. The method is intended to help the engine designer to make more informed initial core design decision early in the design process. The method estimates performance implications of core design selections initial and future engine family applications, while accounting for any assumed sources of uncertainty present. The process, shown in Figure 21, contains blocks which require exploration into specific research areas in order to establish techniques required to enable such a design problem to be computationally feasible. The research objective and corresponding research questions guiding the method development are listed in Table 6.

The COMMENCE method displayed in Figure 21 begins with blocks that establish the overall engine architecture to be employed across the product family, as well as a discrete set of requirements for applications the common core family intends to provide competitive applications. In order to successfully place the engine family in a desirable market sector, significant market analyses must be performed. The common core engine family must provide very successful initial applications, establishing the common core program as an attractive product line. Additionally, accurate projections of future customers' requirements must be established. By establishing a representative set of discrete engine applications, candidate common core designs can be evaluated based on the likely performance levels of variant engine applications of the common core definition.

With the established sets of engine requirements for the engine applications considered, candidate benchmark engine designs must be selected. Similar to how the core design and variant applications will be selected, single-application robust design selections are made using the ERDS method for each set of application requirements. These more idealized

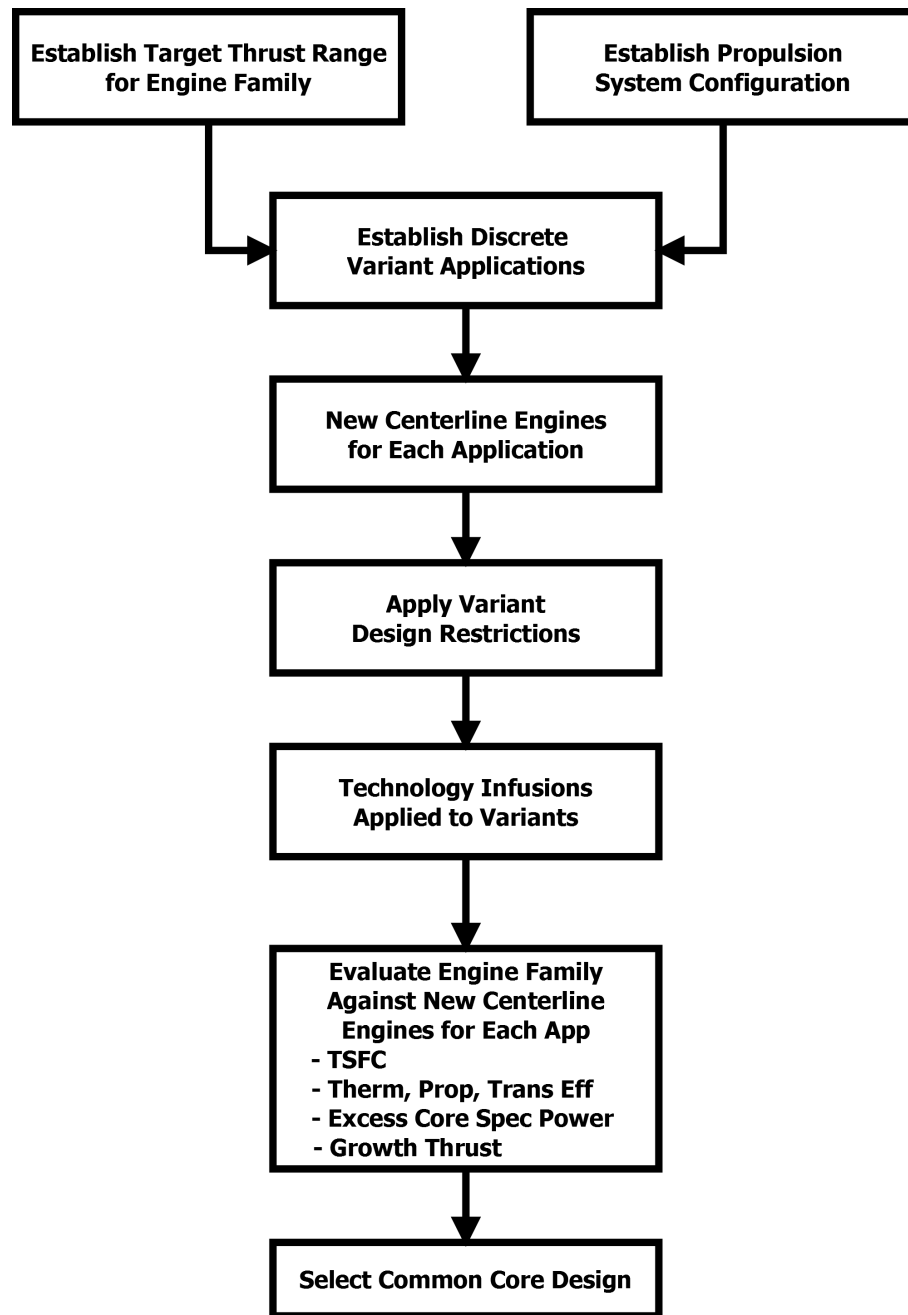


Figure 21: Common Engine Core Evaluation (COMMENCE) Method.

Table 6: Overall Objective and Corresponding Research Questions to be Addressed

RESEARCH OBJECTIVE	Develop a gas turbine engine design and decision making process that aims to increase the useful competitive life and overall versatility of a common core engine family. The process should consider current and future competitive engine family performance, utilizing current and eventual technology improvements without the need for a core re-design.
Primary Research Question	How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?
Research Question 1	How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?
Research Question 2	For a given gas turbine engine core, how should a common core engine variant design be simulated? What parameter(s) must be held to consistent values in order to maintain geometric and aerothermodynamic commonality between engine applications?
Research Question 3	What design options should be considered for common core engine variant applications in order to distribute development capital across the engine program by taking advantage of commonality, while also offering more design freedom when needed for more demanding applications?
Research Question 4	What range of capabilities can various common core design options achieve without significant compromises made in application performance?
Research Question 5	How should a common core engine program consisting of multiple variant design applications be evaluated?

clean sheet designs allow for the determination of the amount of performance compromise made by employing a common core across the applications. Measurement of performance deviation with respect to each benchmark design will be performed just as Simpson and Fellini et al. suggest when evaluating product families[104][34].

After benchmark designs for each engine application are established, the common core design space can be explored. The physics based modeling and simulation environment used should enforce geometric core commonality while also allowing for variant design freedom where permitted. In addition to the variant design rules, the modeling and simulation environment must be able to apply and evaluate impacts of unique technology infusion options for the core defining and variant engines. A probabilistic surrogate modeling technique will be developed and employed to provide efficient means of exploring the multiple application common core design space. Rich sampling, evaluation and selection of candidate common core variant designs and the overall engine family as a whole can then be carried out. Figure 22 shows where key advancements must be made in order for the COMMENCE method to be a viable option for engine family design.

In order to show where the COMMENCE method lies within the larger Integrated Product and Process Development process, Figure 23 displays the IPPD process. The COMMENCE method lies primarily within the shaded regions of the IPPD environment, spanning across Quality Engineering, Systems Engineering, and the Design Decision Support regions of the IPPD process.

This chapter establishes and describes the overall structure and process of the Common Engine Core Evaluation (COMMENCE) method. Detailed descriptions and demonstrations of a key enabling processes will first be provided. The enabling capabilities of the Enhanced Robust Design Simulation (ERDS) method will be provided, identifying it is an integral part of the overall COMMENCE method. It enables efficient exploration and evaluation of designs under various uncertainty scenarios. A case study will be provided to demonstrate how the ERDS method was used for a single engine design.

The common core variant design rules used to enforce geometric commonality also be established and discussed. As is discussed by Lehmann[61], consistent design HPC exit

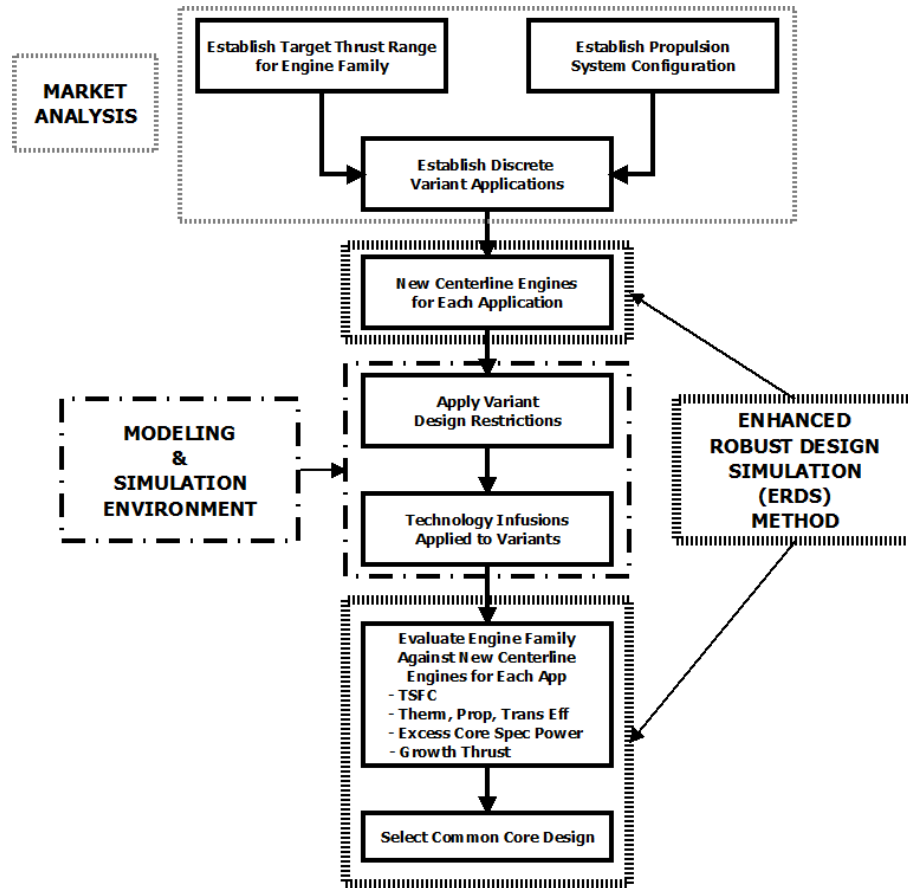


Figure 22: Identification of required advancements to be made and integrated into the overall COMMENCE process.

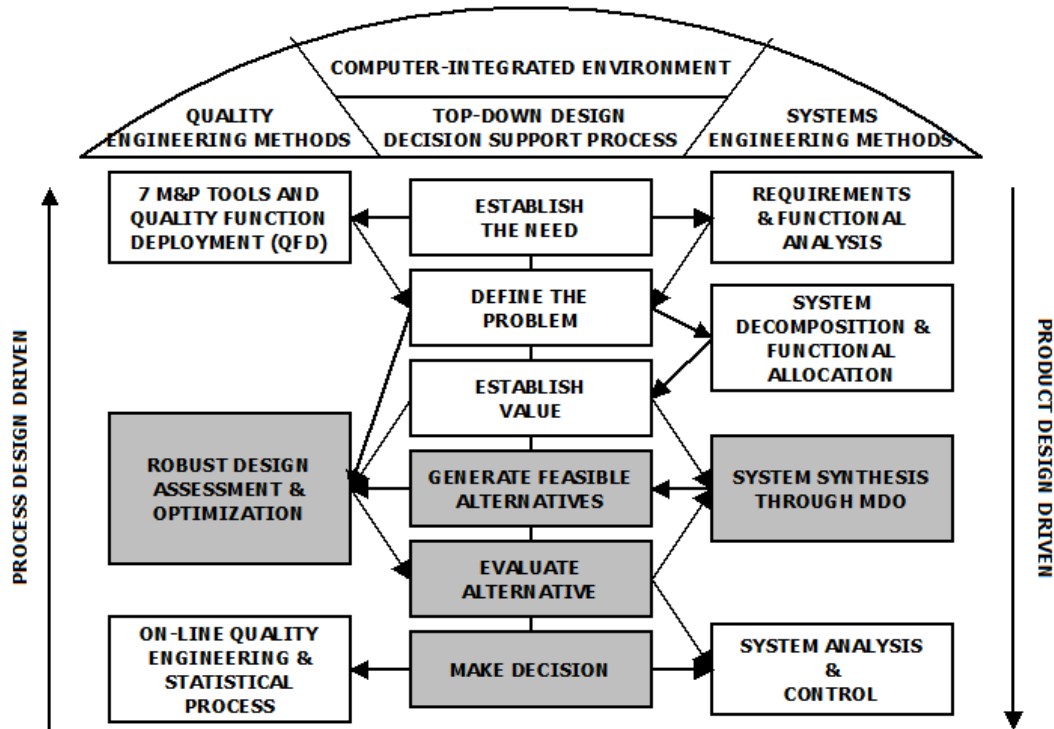


Figure 23: Integrated Product and Process Development decision-making process[101]

corrected flow across applications ensures uniform HPC blade heights and geometrically common flow areas of the engine core components. This maps directly to commonality in the core casing and flowpath geometry. In order to accurately enforce geometric commonality, additional supporting rules are used to enforce a consistent design HPC map operating point where HPC exit corrected flow is made consistent across all common core applications.

The resultant surrogate models trained with data samples of the physics based common core modeling and simulation environment have geometric commonality implicitly enforced. This prevents the need for the designer to filter exploration data or match a core defining engine to a common core variant engine. This greatly improves the ability of the designer to richly sample the common core design space. It will also be shown that by implicitly enforcing commonality also allows for many variant designs to be explored, while all being tied to the same core defining design engine. This key contribution enables a large number of engine family designs to be explored simultaneously, with geometric core commonality

ensured.

After the description of the ERDS method and the common core variant design rules, the fully integrated COMMENCE method will be discussed. The specific steps of the process will be provided, establishing the COMMENCE method as a repeatable, transparent, and methodical approach to providing the designer valuable information when making common core design decisions.

3.1 Enhanced Robust Design Simulation (ERDS)

- *Research Question 1: How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?*

Hypothesis 1: Probabilistic performance levels of candidate cycle designs should be estimated with the use of surrogate models that predict likely performance under various inputted uncertainty scenarios for any desired confidence interval.

The goal of robust design is to select candidate designs that maximize the likelihood of achieving the design requirements while also satisfying any constraints imposed, all in the presence of uncertainty. For the present work, the sources of uncertainty considered are the technology impact uncertainty as well as the installation bleed and power extraction requirements uncertainty. While the following process description primarily discusses simulation of technology uncertainty, the process can be easily extended to other noise sources including operating conditions, cost factors, or anything else as appropriate that can be simulated within a computational tool. In this context, examples of technology impacts include component level weight reduction or component efficiency improvements. A full list of assumed uncertainty distributions will be presented in the following chapter for each of the experiments performed, but the generic method for generating the actual distributions will be presented in this section.

Before estimating the impacts of technologies the designer may infuse into a new design, a baseline model definition must be established. The baseline acts as the datum that all new

candidate designs can be evaluated against, and usually represents a design at the current technology level present in the market. The baseline engine model also acts as a technology collector onto which technologies can be applied and the cycle advanced to represent a new evolutionary design.

Once the baseline definition is established, technologies can be identified that aim to provide specific benefits. For each technology considered, distributions must be defined that represent the likelihood of the technology achieving its assumed impact(s). Beta distributions are easily constructed to represent the variation in technologies' likely impacts. Beta distributions are bounded and can be shaped based on four factors: the minimum and maximum possible impact, the location of the most likely impact within the range considered, and the amount of confidence in achieving the most likely impact. Based on the domain of the most likely value within the bounds and the level of confidence in attaining that most likely value the beta shape parameters, α and β , can be determined using a table lookup provided by Batson[9], which has been recreated in Table 7, and the distribution can be defined.

Assumed technology impact distributions can be fixed as is the case with the traditional RDS process and robust design selections can be made. However, in order to have the ability to consider various uncertainty scenarios, as is necessary for the multi-application common core design problem, the shape factors defining the distributions must be varied. Modification of a prior assumed uncertainty distribution can be used to represent a *Tech Push*, an initiative taken to improve the mean technology impact and/or reduce the variance in the impact by maturing a particular technology, as depicted in Figure 24. The x-axis would represent the level of technology benefit, such as component efficiency improvement, and the y-axis represents the likelihood of achieving the specific impact within the range.

In order to simplify the demonstration of the process the technology uncertainty distributions will be held constant for the sample problem demonstrated later. However, this is not a limitation of the ERDS method. Various uncertainty distributions will be simulated in the following chapter when considering initial and future common core applications with different assumed technology levels. Robust design and technology selections can then be

Table 7: Beta distribution shape parameter lookup table based on domain of normalized modal value and confidence level of modal value[9].

Confidence Level	Domain of Normalized Modal Value								
	0.00 to 0.15	0.15 to 0.25	0.25 to 0.35	0.35 to 0.45	0.45 to 0.55	0.55 to 0.65	0.65 to 0.75	0.75 to 0.85	0.85 to 1.00
Highest	$\alpha = 2$ $\beta = 8$	$\alpha = 3$ $\beta = 8$	$\alpha = 4$ $\beta = 8$	$\alpha = 6.5$ $\beta = 8$	$\alpha = 8$ $\beta = 8$	$\alpha = 8$ $\beta = 6.5$	$\alpha = 8$ $\beta = 4$	$\alpha = 8$ $\beta = 3$	$\alpha = 8$ $\beta = 2$
Higher	$\alpha = 1.5$ $\beta = 6$	$\alpha = 2.25$ $\beta = 6$	$\alpha = 3$ $\beta = 6$	$\alpha = 4.25$ $\beta = 6$	$\alpha = 6$ $\beta = 6$	$\alpha = 6$ $\beta = 4.25$	$\alpha = 6$ $\beta = 3$	$\alpha = 6$ $\beta = 2.25$	$\alpha = 6$ $\beta = 1.5$
High	$\alpha = 1.25$ $\beta = 4$	$\alpha = 1.75$ $\beta = 4$	$\alpha = 2.25$ $\beta = 4$	$\alpha = 3$ $\beta = 4$	$\alpha = 4$ $\beta = 4$	$\alpha = 4$ $\beta = 3$	$\alpha = 4$ $\beta = 2.25$	$\alpha = 4$ $\beta = 1.75$	$\alpha = 4$ $\beta = 1.25$
Lower	N/A	$\alpha = 1.5$ $\beta = 3$	$\alpha = 1.75$ $\beta = 3$	$\alpha = 2.25$ $\beta = 3$	$\alpha = 3$ $\beta = 3$	$\alpha = 3$ $\beta = 2.25$	$\alpha = 3$ $\beta = 1.75$	$\alpha = 3$ $\beta = 1.5$	N/A
Lowest	N/A	$\alpha = 1.25$ $\beta = 2$	$\alpha = 1.5$ $\beta = 2$	$\alpha = 1.75$ $\beta = 2$	$\alpha = 2$ $\beta = 2$	$\alpha = 2$ $\beta = 1.75$	$\alpha = 2$ $\beta = 1.5$	$\alpha = 2$ $\beta = 1.25$	N/A

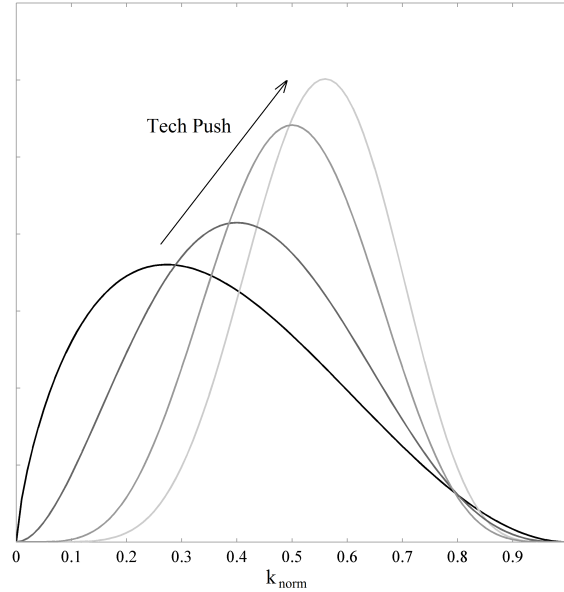


Figure 24: Progression of probability density function for impact of technology being matured.

made, selecting designs that aim to maximize the likelihood to achieving high performance while being less sensitive to assumed noise in the level of technology uncertainty impacting the final performance metrics.

Once the baseline model definition is established, technology impacts and their corresponding distributions have been identified, and the design variable ranges are established, the RDS process can begin. The generic RDS methodology developed by Mavris et al.[71] has been adapted for the present engine design studies and is depicted in Figure 25. The enhancements of the traditional process are contained within the shaded nodes of the process flow chart.

The goal of the first portion of the RDS method is to generate a deterministic set of surrogate models that accurately represent the physics based engine design and technology space considered for the robust design study. This enables assumed input values or distributions to be rapidly changed during the analysis process. For the variables and associated ranges considered, a design of experiments (DoE) is constructed for the cycle design variables and the technology noise variables[116]. Each DoE case, a vector of design and noise

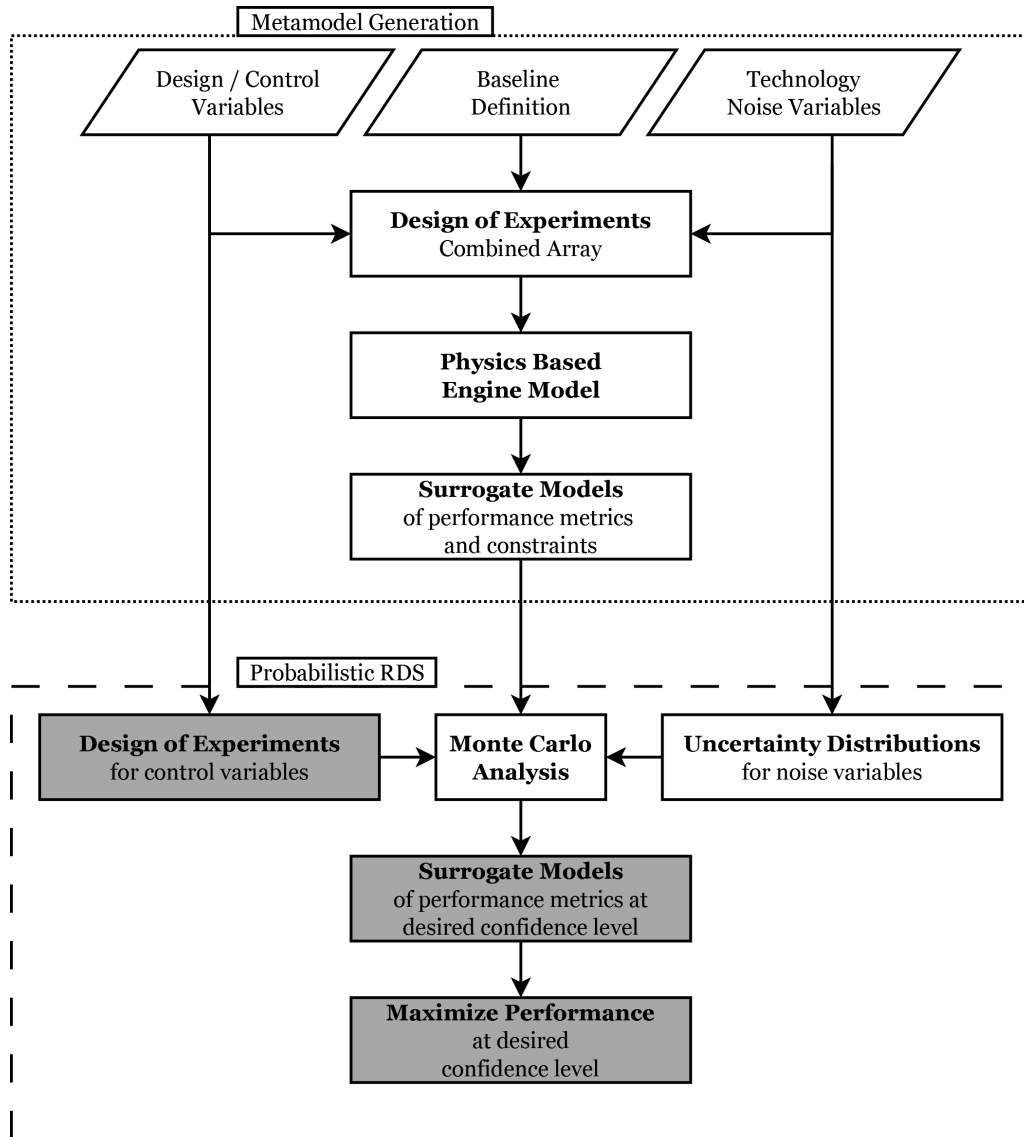


Figure 25: Robust Design Simulation method modified for robust engine and technology design.

variable settings, represents a unique engine design and the corresponding performance is estimated by the physics based engine model. The DoE is executed through the physics based model and the results are used to train deterministic surrogate models of performance metrics to evaluate a design’s ability to satisfy all requirements and constraints. The goal is to accurately estimate an output metric value y as a function of a set of input variable values consisting of design variables, \mathbf{X} , and technology impact variables, \mathbf{K} , as shown in Equation 13. As was discussed in the previous chapter, Artificial Neural Networks (ANN) have shown to accurately represent the nonlinear behavior of engine aerothermodynamic cycle performance models with minimal losses in the accuracy of the model[54]. However, it must be ensured that when employing the surrogate models to predict engine performance that the input vector must contain variable values that are within the variable ranges originally considered in the DoE used to train the model.

$$y = f(\mathbf{X}, \mathbf{K}) \tag{13}$$

At this point, if one were applying the traditional RDS process, the user selects distributions for the technology parameters, as described previously. Then Monte Carlo simulation is performed by selecting a chosen set of fixed inputs for the design parameters and Beta distributions, as defined in Table 7, for the technology variables. Using the traditional process the user would then have to use the results of the Monte Carlo to generate cumulative distribution functions (CDFs), as shown in Figure 26. The CDF shown indicates the likelihood of a design meeting or exceeding (in this case exceeding refers to reducing the metric value in the direction of improvement) a specific metric value. Often times joint probability plots are also constructed to show the effects of multiple random variables on a response.

This demonstrates the fundamental limitation of the traditional RDS process. The user must select assumed (fixed) technology parameter distributions and is then shown the probability of meeting certain metrics for a single set of design parameter settings. If the designer wants to estimate the confidence level for a different set of uncertainty distributions he must rerun the Monte Carlo analysis and recreate the single or joint CDFs. This process does not allow the designer to effectively visualize and understand the entire design space.

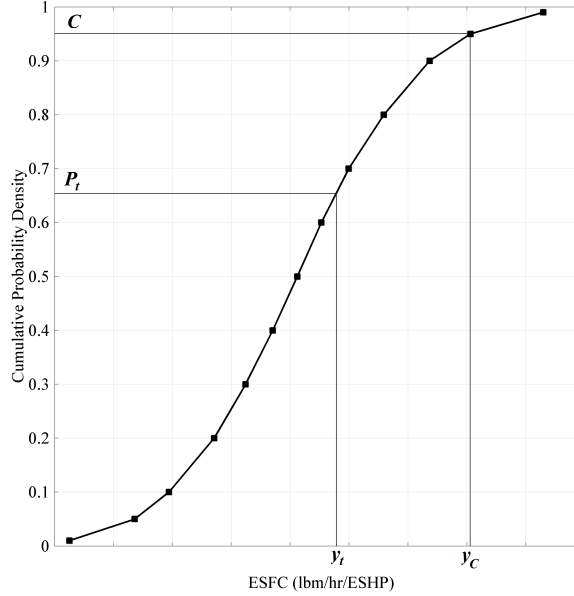


Figure 26: Cumulative distribution function for likely performance of a candidate design.

The ERDS method augments the traditional RDS process to address the aforementioned shortcoming. Once all of the surrogate models representing the physics based engine model have been trained, the probabilistic assessment is performed. Again, a DoE is constructed, but this time only of the control variables. The control variables traditionally contain only the design variables. However, the uncertainty distribution shape parameters for technology j (α_j and β_j), as well as the minimum and maximum technology impact values ($k_{j,min}$ and $k_{j,max}$) can also be included as control variables. A DoE case constructed for the probabilistic assessment now consists of a unique vector of design variable settings and a unique set of uncertainty distribution shape parameter settings. For each given set of design variable settings and inputted uncertainty scenario, Monte Carlo simulations are performed, accounting for the assumed uncertainty distributions present, resulting in confidence interval performance levels for the candidate design considered under the inputted uncertainty scenario. Figure 27 depicts the process for performing the Monte Carlo probabilistic assessment that results in estimates of likely performance levels of candidate designs under uncertainty.

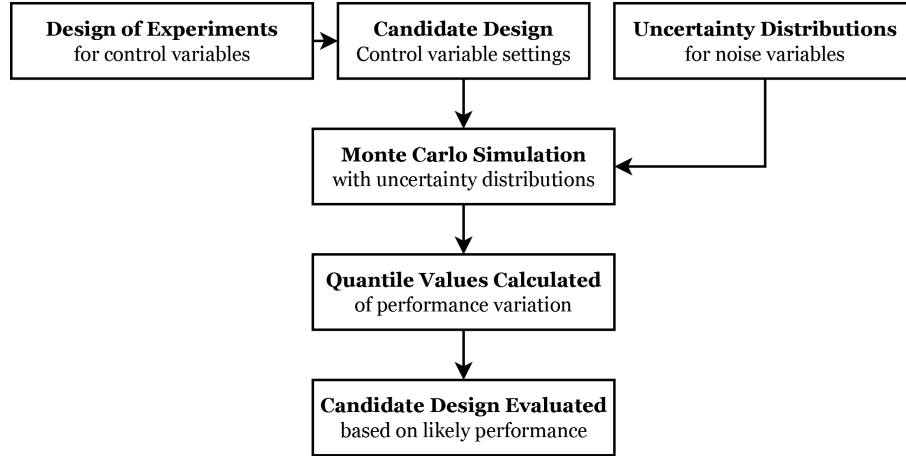


Figure 27: Monte Carlo probabilistic analysis process.

With the probabilistic data produced by the Monte Carlo probabilistic assessment, another layer of surrogate models can be trained. The resultant probabilistic surrogate models are able to instantaneously estimate variability in the likely behavior of candidate designs consisting of a unique set of control variable settings. Originally, when developing this new process, surrogate models were constructed to estimate the probability of meeting or surpassing a particular performance target (Probability P_t of meeting performance target y_t , shown in Figure 26) based on the control variable settings, formalized in Equation 14. However, if a substantial amount of Monte Carlo simulations used to train the surrogate model either do not satisfy the target performance level and/or a large number of samples easily meet the target with 100% likelihood, it becomes extremely difficult to train a surrogate model that provides accurate prediction of this multi-modal behavior. Therefore, instead of training models to estimate the probability of achieving a discrete target value, surrogate models are trained to predict the metric performance level at desired (inputted) confidence levels (Performance level y_C at desired confidence level C , shown in Figure 26).

No matter where the likely performance of a design lies, there will always be a cumulative distribution function (CDF) of likely performance levels. This eliminates the possibility of having multi-modal data, having sparse data with which to train a surrogate model as is often the case with the traditional approach. Also, this way of capturing probabilistic data

now allows for candidate design comparison of performance levels to be done at consistent confidence levels. For example, it allows for comparison of two designs' 95% likely performance levels, or any other confidence level of interest. Another added benefit of performing probabilistic design evaluation in this form is that the data trends take the same shape as when performing deterministic assessments, instead of data trends taking a new form that resemble CDFs. This allows designers to continue with analysis of trends they are comfortable with instead of forcing them to learn new probabilistic trends predicting the probability of meeting some arbitrary performance target.

With the addition of control variables for the uncertainty distribution factors as well as the added freedom in the desired confidence C the designer is interested in having corresponding performance estimates for, the probabilistic surrogate models take the form of Equation 15. Vector \mathbf{X} contains all design parameters. Vectors \mathbf{K}_{\min} and \mathbf{K}_{\max} contain the minimum and maximum technology impact levels of each technology distribution considered. Vectors \mathbf{A} and \mathbf{B} contain the alpha and beta shape parameters for each technology uncertainty distribution. Adding the significant number of control parameters required for full control of the technology uncertainty distributions and training surrogate models as functions of the additional variables requires additional scrutiny when accepting whether a surrogate models accuracy is acceptable. The ANNs used to represent the probabilistic behavior of the engine cycle and technology design space have shown to accurately represent the nonlinear behavior of gas turbine engine models, while also allowing the surrogate models to be functions of a large number of independent variables, on the order of hundreds of variables[54].

$$P(y \leq y_t) = f(\mathbf{X}) \quad (14)$$

$$y_C = f(\mathbf{X}, \mathbf{K}_{\min}, \mathbf{K}_{\max}, \mathbf{A}, \mathbf{B}, C) \quad (15)$$

After training the probabilistic surrogate models that estimate performance at any desired confidence levels, the designer is able to richly sample the candidate design and technology space and predict probabilistic behavior of the unique designs considered. Candidate design cycle settings can be explored and robust selections can be made that offer the best

performance at a desired confidence interval level. Additionally, identification of technology subsets can be identified that offer the best cost-to-benefit ratios. Effects of reducing certain technology impact variation due to uncertainty can offer insight into which technologies should be matured in order to allow for the highest likely performance gains, while also showing how the reduction(s) in technology uncertainty affects the resultant robust cycle selections. The models can be combined with economic models and integrated into business decision making environments, such as Bayesian Belief Networks (BBNs), allowing decisions makers to quickly and efficiently perform cost-benefit analyses and through forwards and backwards inference, predict how their decisions will impact the likely design performance[53],[50].

The above description of the ERDS process offered the rationale of using the method. Alternatively, the following section offers a step-by-step procedure explicitly describing the steps required to carry out the ERDS method.

3.1.1 ERDS: Step-by-Step Procedure

This section describes the specific steps to follow when carrying out the ERDS method for a robust design problem. Description of each of the steps is intended to be generic enough to apply to the robust design of any product of interest. More specific procedures are provided for each of the experimental applications of the ERDS method found in the following chapter. The primary steps of the ERDS method are as follows.

1. Physics based model preparation
2. Deterministic assessment and regression
3. Probabilistic assessment and regression
4. Robust design exploration and selection

3.1.1.1 Step 1: Physics based model preparation

The goal of this initial step is to establish a baseline definition within the physics based model and to determine the variables for which explorations will be performed. The baseline

should be tuned to represent either an existing design that currently exists or an agreed-upon technology level design against which advanced designs will later be evaluated.

Once the baseline definition is established, a set of input variables should be identified for exploration. These should first consist of design variables which settings will later be selected to define an advanced design that offers the best performance under uncertainty. Ranges of these design variables should be established so that ample exploration of the design space can be performed. It is important to ensure that the input variables ranges are large enough to encompass all design regions that may be of interest.

In addition to the set of design variables, a set of noise variables must be selected which uncertainty distributions will later be applied during the probabilistic assessment. Similar to the design variables, the noise variable ranges must be wide enough to envelop any possible uncertainty distributions that the designer may want to account for when performing later probabilistic assessments.

Once the superset of input variables to be explored has been established, the designer must consider the computational budget allowed for the initial deterministic assessment. Based on the computational cost of each unique simulation using the physics based model and the overall computational budget, the designer may be limited in the number of simulations allowed. If necessary, screening can be performed on the input variables, eliminating input variables that have very little impact on the metric values of interest.

With the final set of input variables and corresponding ranges, a Design of Experiments (DoE) is created, establishing a combined array of design and noise variable settings that will be used to sample the physics based model for later regression. As mentioned in the previous chapter, latin hypercube designs are space filling designs that aim to maximize the amount of information about the sampled space with a limited number of cases allowed. A space filling DoE containing a number of cases that is between 20 and 40 times the number of input variables typically offers enough training data to produce accurate surrogate model representations of the sampled model. In addition to these training cases, additional test samples are necessary to ensure that resultant surrogate models not only accurately estimate the data used to train the model, but also provide accurate estimates for cases that were

not used to train the regression models. These testing samples can be randomly sampled within the input variable ranges considered, and should contain between 15 and 20 percent of the total number of samples.

3.1.1.2 Step 2: Deterministic assessment and regression

Before launching the DoE with the physics based model, a set of output metrics of interest should be established. The model should be able to produce a file that compiles the input and output data of interest into a single file for all cases to be simulated. Once the DoE is created and the model prepared, execution of the DoE is performed.

After all simulations have been performed, the resultant input/output data can be then be used in the regression and testing of deterministic surrogate models. A model will be required for every single output metric of interest to the designer when making design decisions. It is recommended that the samples making up the entire latin hypercube are used in the training, and the remaining samples reserved for testing of the resultant surrogate models for fit quality. Many software packages offer training algorithms for surrogate model regression. When fitting Artificial Neural Network (ANN) surrogate models, it is recommended to explore the model architecture, allowing for the number of hidden nodes of the ANN to change. The ANN architecture and corresponding model coefficient values that produces the best fit quality is then selected.

In order to ensure adequate model fit quality, the following surrogate model fit requirements should be met:

- Model Fit Error (MFE), the distribution of surrogate model error that is determined through the comparison of predicted performance to the actual performance levels used in the training of the surrogate model
 - Shape of MFE distribution should resemble a normal distribution
 - Mean value of MFE distribution should be close to zero
 - MFE standard deviation (STD) should be less than one
- Model Representation Error (MRE), the distribution of surrogate model error that

is determined through the comparison of predicted performance to the actual performance levels of the reserve data not used in the training of the surrogate model. MRE quality determined based on the same metrics used in the evaluation of the MFE distribution.

- The quotient of MRE and MFE standard deviations should have a value less than two.
- A high coefficient of determination (R^2) must be achieved, having a value greater than 0.99. A value close to one indicates the model is able to explain the vast majority of variation in the metric response of interest. The model's R^2 can be determined using Equation 16.
- When visualizing the actual-by-predicted plot, the data should not contain tails where actual-by-predicted values depart from the perfect fit line at low and high values of response data.
- The residual error should be at least two orders of magnitude less than the actual response levels.

A model's coefficient of determination (R^2) is determined by:

$$R^2 = 1 - \frac{SS_{error}}{SS_{total}} \quad (16a)$$

where

$$SS = \sum (Y - \bar{Y})^2 \quad (16b)$$

and

$$\bar{Y} = \frac{\sum Y}{N} \quad (16c)$$

If the above requirements are not met by the various surrogate models, then attempts to improve surrogate model fit can be done with the existing DoE data, or additional training samples may need to achieve adequate surrogate fit quality. Once the level of

model fit quality is achieved, the resultant set of surrogate models can be used to provide deterministic performance estimates for design and noise variable values within the ranges explored for training.

3.1.1.3 Step 3: Probabilistic assessment and regression

Now that accurate mathematical representations of the physics based model has been achieved, the design and uncertainty space contained within the ranges previously samples can now be more richly sampled. The range of uncertainty scenarios of interest are used to determine the ranges of the uncertainty distributions shape parameter settings to be sampled in the following assessment. Again, it must be insured that the distributions to be simulated have noise variable impacts that fall within the ranges initially sampled to produce the deterministic surrogate models.

The same set of design variables and corresponding ranges previously sampled are then used to construct another DoE. This DoE should also contain the uncertainty distribution shape parameters and corresponding ranges that envelop any scenarios of interest. Another latin hypercube training set and random surrogate testing set of ample size is then constructed.

The probabilistic assessment can then be performed. For each DoE case containing unique values for design variables and uncertainty distribution shape parameter settings, Monte Carlo simulations are performed. When a significant number of uncertainty distributions is to be accounted for, many tens-of-thousands of Monte Carlo simulation are necessary to adequately evaluate the design under uncertainty. For example, if 20 to 30 distributions are to be simulated, repetitions on the order of 50,000 simulations were shown to provide the resolutions necessary to accurately represent the design under the uncertainty scenarios considered. If 3,000 DoE cases are necessary, each requiring 50,000 Monte Carlo simulations, 150 million function calls would be required for each output metric of interest. Although this sounds computationally expensive, the deterministic surrogate models being evaluated are closed form equations that can be evaluated almost instantaneously. For example, a probabilistic assessment of this exact size was performed using a Matlab script on

a single quad-core machine in less than 3 hours.

For each DoE case, the Monte Carlo simulation data should be sorted for each metric of interest, allowing for the calculation of quantile metric values. For example, if it is desired to have the ability to estimate any quantile value for the metrics of interest, as is the case for the studies of this work, quantile metric values should be calculated across the entire cumulative distribution function for each DoE case. A resultant file should be constructed, where each row contains the design variable settings, the shape parameter settings, a quantile value, and the corresponding quantile metric values for the design/scenario considered.

Training of probabilistic surrogate models can then be regressed, where the same model fit quality requirements must be met as with the deterministic surrogate models. The resultant set of probabilistic surrogate models now predict the likely performance at the inputted confidence interval for an inputted set of design variable and uncertainty distributions shape parameter settings. This again highlighted the powerful capabilities of the ERDS method. Any confidence interval performance level can be instantaneously predicted with a single function call for a unique design and uncertainty scenario within the range of designs and scenarios sampled for surrogate training.

3.1.1.4 Step 4: Robust design exploration and selection

Now that probabilistic surrogate models are available, exploration of the robust design space can be performed. There are various techniques of exploration. Visualization of the design space is often desired, allowing the designer to identify performance trends with design variables.

As was described in Chapter 2, multivariate scatterplots and parallel coordinate charts are powerful sample-based visualization techniques, particularly when explorations are performed on many input variable dimensions. These techniques require samples of a design space and they display each sample's placement in the input space and the resultant metric space. The probabilistic surrogate models enable any millions of samples to be evaluated, allowing for very rich sampling of the design space under consideration.

Another visualization technique is enabled through the generation of probabilistic surrogates. Contour profilers can be constructed, where a given design plane can be shown, and contours of constant metric values can be displayed on the chart. The placement of the contours in the design plane is set based on the settings of the two axes of the chart as well as the remaining set of fixed input variable settings. These charts allow the designer visualize the continuous design space, showing how variable changes impact the metric values of interest.

In addition to visualization, filtering can be performed, eliminating either samples or continuous design regions that violate constraints or do not meet imposed requirements of the design. Optimization can also be performed within the design space of interest, resulting in candidate designs that provide competitive performance and lie in a desirable design region.

Regardless of the technique of design selection, more informed candidate design selections can be made with the ERDS process. Various uncertainty scenarios can be accounted for, and candidate designs can be selected that have competitive performance estimates at high confidence interval levels. Exploration of the effects of technology infusion or the like can also be performed, showing how likely performance changes with the uncertainty scenario and how the resultant design selection would change under different scenarios.

If ample considerations are made at the onset of the ERDS method when initial variables and ranges are established, the resultant set of probabilistic surrogate models can be valid for a wide variety of design problems and uncertainty scenarios. The increased upfront computational requirements result in significant decreases in the later computational burden when carrying out robust design probabilistic assessments. As will be shown in the next chapter, this process was carried out only once, providing a series of deterministic and probabilistic surrogate models that were utilized for all of the experiments contained in the following chapter.

The next section presents the first publicized application of the ERDS method. The current author carried out the above steps, applying the method to the robust design of a two spool turboshaft engine for a heavy-lift helicopter application under significant technology

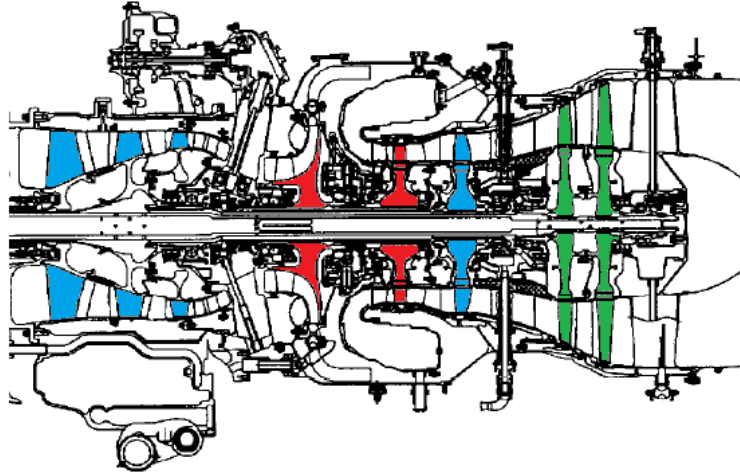


Figure 28: Two spool axi-centrifugal turboshaft engine architecture.

uncertainty[99]. Comparison to a deterministic design method is included, showing both the changes in the design selection and the performance estimates when employing the ERDS method.

3.1.2 ERDS Case Study: Robust Design of Two-Spool Axi-Centrifugal Turboshaft Engine

In order to demonstrate the enhanced robust design simulation process, it will be applied to the design of a two-spool axi-centrifugal turboshaft architecture design. The notional power turbine shaft horsepower requirements at various flight conditions are provided in Table 8. The power requirements are based around those typical for a heavy lift rotorcraft application. An example of the turboshaft architecture considered is shown in Figure 28. The axial LPC and turbine stage powering the LPC are colored blue. The centrifugal HPC and accompanying turbine stage are colored red. The power turbine stages are colored green.

Table 8: ERDS Case Study - Shaft horsepower requirements at various flight conditions.

Design Point	Top of Climb	Sea-Level Static Takeoff	High Hot Takeoff	Cruise
Designation	TOC	SLSTKO	HTKO	CRZ
Altitude (<i>ft</i>)	6,000	0	5,400	6,000
Mach	0.11	0.00	0.00	0.17
dTs ($^{\circ}F$)	57.4	51.0	55.3	0.0
Power (<i>shp</i>)	5,500	7,300	6,500	3,200

3.1.2.1 Modeling and Simulation Environment

An aerothermodynamic cycle performance model was constructed using the Numerical Propulsion System Simulation (NPSS) software developed by NASA[49]. A corresponding weight prediction model was constructed using NASA’s Weight Analysis of Turbine Engines (WATE) software[111]. NPSS and WATE models together estimated the cycle performance of the turboshaft engine, which in turn was used to calculate the required weights of the engine components. Artificial neural network surrogate models were trained using Matlab’s Neural Network Toolbox[10]. Matlab software was also used to construct a latin hypercube space-filling DoE, the noise variable beta distributions, and subsequent Monte Carlo probabilistic analyses of candidate engine designs.

3.1.2.2 Baseline Model Definition

In order to ensure that all horsepower requirements are met by the constant speed free power turbine, a multi-design point (MDP) approach was used[103]. The MDP approach allows for the designer to ensure that all requirements are exactly met by the aerothermodynamic engine cycle while also satisfying all imposed constraints using the NPSS solver. The design maximum turbine inlet temperature ($T_{41,max}$) was treated as a constraint and effectively sizes the engine at the high hot takeoff (HTKO) design point. It was assumed that the free power turbine must provide a constant physical RPM at each of the four design points. A constraint was also imposed that limited the engine pressure ratio (EPR) from being lower than 1.05 at all design points.

The two engine design points that determined a majority of the engine component performance and size were the top of climb (TOC) design point and HTKO design point. The TOC point was treated as the aerodynamic design point for the turbomachinery components, where the assumed baseline polytropic efficiency was set and the turbomachinery performance maps were scaled accordingly. The HTKO design point was the sizing point, where the engine was sized to provide enough air flow to meet the power requirement while also satisfying the input design $T_{41,max}$. The cruise (CRZ) design point is merely an evaluation point, used for the purpose of comparing candidate design performance levels.

3.1.2.3 Axi-Centrifugal Design Considerations

Since the results of this study are dependent on the design consideration of axi-centrifugal machines, some of the relevant modeling details are explained here. Axial compressors have the potential to be more efficient than centrifugal compressors. They can be easily staged to gradually add pressure to the flow. However, as the engine size, typically represented by the HPC exit corrected flow (W_{3R}), becomes small, an axial compressor's achievable polytropic efficiency becomes greatly degraded due to secondary flow and clearance losses[84]. The manufacturability of the small axial compressor stages required for more compact engines becomes challenging as well. Therefore, as the engine size becomes small enough, centrifugal compressors become more feasible. While centrifugal compressors are relatively insensitive to *size effects* compared to the axial compressors, they are sensitive to *speed effects*[84]. Polytropic efficiency losses due to speed effects can be estimated as a function of the compressor's specific speed $N_{s,HPC}$, a geometric similarity parameter defined in Equation 17. The specific speed is a function of the centrifugal compressor physical speed N , the volumetric flow rate Q , and the total enthalpy rise ΔH across the compressor. Specific speeds on the order of 0.75 are desirable. Lower specific speeds incur frictional losses due to longer, lower aspect ratio passages while higher specific speed compressors induce aerodynamic losses due to increased relative exit velocity levels[16].

$$N_s = \frac{N\sqrt{Q}}{\Delta H^{0.75}} \quad (17)$$

Centrifugal compressor material stress limitations also restrict the compressor efficiency

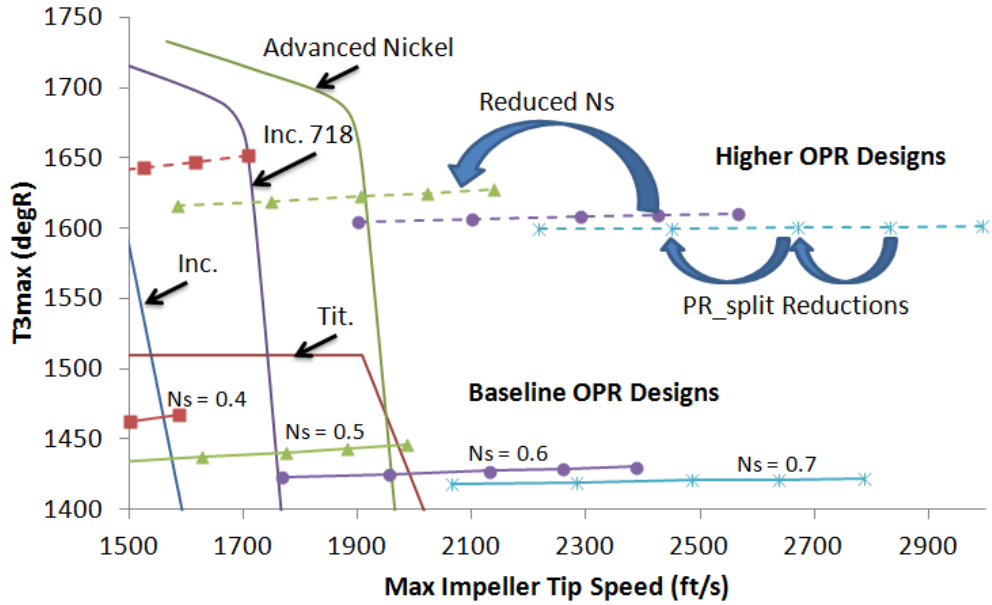


Figure 29: Centrifugal compressor tip speed constraints due to material stress limits.

achievable. At high impeller exit temperatures (and corresponding higher pressure ratios), the allowable exit tip speed is greatly reduced. Therefore, in order to achieve specific speeds that minimize efficiency losses while also achieving high OPR values, the centrifugal compressor and overall engine must remain small, limiting the amount of power an engine with a centrifugal compressor can generate. This leads to the need for an axial-centrifugal design. The axial compressor is used to efficiently compress the flow until a centrifugal compressor is more efficient from a size effects perspective. Splitting the pressure rise between the two components also allows the centrifugal compressor pressure ratio to remain low, limiting design tip speed and diameter.

Figure 29 contains tip speed limits for various centrifugal compressor materials[84],[114]. Even with the most advanced centrifugal compressor materials found in the public domain, the feasible specific speed levels are limited to values lower than desirable levels due to tip speed restrictions imposed on the large diameter centrifugal compressors at the baseline technology level. For higher OPR designs, the tip speed limits are even more stringent due to elevated compressor exit temperatures.

To demonstrate the design trades Figure 30 shows the variation in engine ESFC and weight for a constant OPR baseline design with varying design specific speed and pressure ratio split between the axial and centrifugal compressors. Each curve represents an axi-centrif design with a fixed design specific speed centrifugal compressor. As the pressure ratio split, or the portion of the overall pressure ratio assigned to the centrifugal compressor (defined in Equation 18) decreases, the design becomes more competitive in both ESFC and weight. This shows that the centrifugal compressor should provide a minimal pressure ratio to replace the small, aft axial compressor stages that might induce efficiency losses due to size effects. The baseline definition is representative of 1980s level technology, and acts as a technology collector for applying advanced cycle and material technologies. At the baseline technology OPR of 16.6 and without the addition of advanced centrif materials, the baseline engine, depicted by the red circle in Figure 30, is able to achieve cruise ESFC levels on the order of $0.59 \text{ lbm}/(\text{hr} \cdot \text{eshp})$ with a corresponding bare weight of approximately $2,180 \text{ lbm}$. As advanced cycle and material technologies are applied to the baseline technology collector, ESFC and weight estimates will be able to be reduced, with the ultimate goal of the present study to minimize cruise ESFC and weight estimates of candidate engine designs.

$$HPCPR = OPR^{P_{R_{split}}} \quad (18)$$

3.1.2.4 Technology Identification

Advanced technologies were identified that aim to enable a compact, highly efficient turboshaft engine. Advanced cycles with high OPR and T_{41} levels were projected by work done with the Large Civil Tilt-Rotor (LCTR) program[106]. Additional technologies identified in the public domain and whose impacts are accounted for in this study include:

1. Inlet recovery and turbomachinery efficiency improvements through improved design and manufacturing techniques[38]
2. Improved turbine cooling leakage and durability[38]
3. Titanium-Aluminide (TiAl) HPC and FPT[105][91]

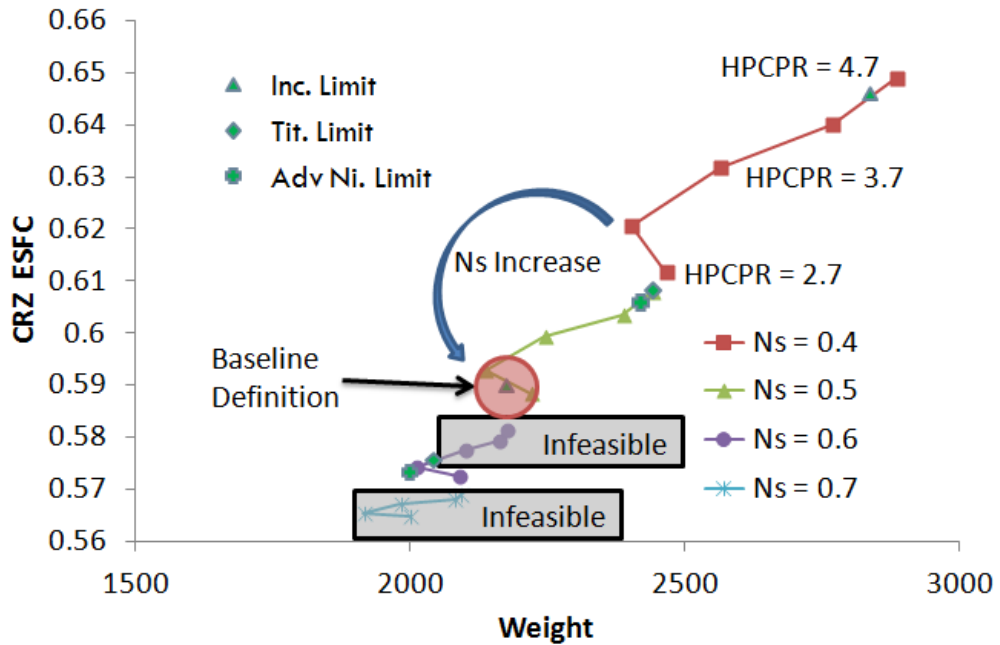


Figure 30: Axi-centrif engine performance limited by material stress limits.

4. Highly loaded axial compressor[27]
5. Advanced HPT material (Rene-N6)[115]
6. Highly loaded free power turbine[20]
7. Advanced Nickel centrifugal compressor[114]

Beta distributions were constructed for the identified technologies' associated impacts. Additional beta distributions were constructed to represent increases in assumed airframe horsepower extraction and bleed requirements due to likely increases in airframe system requirements[6][75]. The ranges of each impact variable and their associated beta distribution shape factors were estimated. The various uncertainty distributions under which robust designs were explored are contained in Table 9, Table 10, Table 11, and Table 12.

3.1.2.5 Surrogate Model Creation and Validation

The goal of the initial DoE is to provide adequate sampling of the physics based engine model to allow for quality surrogate model fits that provide good representation of the pertinent

Table 9: ERDS Case Study - Installation Uncertainty Distributions

Description	k_{min}	k_{max}	Alpha	Beta
Customer HP Extraction (shp)	35	70	1.5	3.0
Customer Bleed (lbm/s)	0.02	0.04	1.5	3.0
Accessory Weight (lbm)	50	68	3.0	2.25
Inlet Pressure Recovery	0.95	0.975	1.25	4.0

Table 10: ERDS Case Study - Component Efficiency Uncertainty Distributions

Description	k_{min}	k_{max}	Alpha	Beta
Base Poly Eff, Axial LPC	0.8875	0.9005	2.25	6.0
Base Poly Eff, Centrif	0.87	0.923	2.25	6.0
Base Poly Eff, HPT	0.83	0.92	4.25	6.0
Base Poly Eff, LPT	0.83	0.92	4.25	6.0
Base Poly Eff, FPT	0.85	0.91	2.25	6.0
Charge Cooling Factor, HPT and LPT	0.95	1.01	6.0	3.0
NonCharge Cooling Factor, HPT and LPT	1.9	2.0	6.0	2.25

Table 11: ERDS Case Study - Component Weight Uncertainty Distributions

Description	k_{min}	k_{max}	Alpha	Beta
Forward Stage Wt Factor, LPC	0.9	1.0	1.75	3.0
Aft Stage Wt Factor, LPC	0.5	1.0	1.5	3.0
Wt Factor, HPC	0.9	1.0	1.75	3.0
Wt Factor, HPT	1.0	1.1	4.0	4.0
Wt Factor, LPT	1.0	1.1	4.0	4.0
Wt Factor, FPT	0.5	1.0	1.5	3.0

Table 12: ERDS Case Study - Miscellaneous Uncertainty Distributions

Description	k_{min}	k_{max}	Alpha	Beta
GE Loading, FPT	0.6	0.8	3.0	2.25
GE Loading, HPT	0.8	1.06	3.0	2.25
GE Loading, LPT	0.8	1.06	3.0	2.25
Intercomp Duct Pt Loss	0.0	0.025	4.0	3.0
Delta Centrif Tip Speed Limit	-200	200	4.0	4.0

metrics and constraints as functions of the design (control) and technology (noise) variables. The initial DoE consists of a combined array of all control and noise variables, using a latin hypercube space filling design sampling throughout the design and technology space. Along with the ranges established for the technology impacts, ranges must be constructed for all of the design variables considered. The design variables for which robust selections will be made for this study are listed in Table 13 along with their corresponding ranges. The ranges of the cycle design variables were selected based on current capabilities of similar engine architectures as well as considerations of possible capabilities of future advanced technology engines. A space filling DoE design of all control and noise variables was used to ensure efficient sampling of the design and technology space considered. Due to the number of model variables considered, a total of 4,000 DoE cases were generated, with 80% of those being allocated for surrogate model training and the remaining 20% for verifying a good model fit quality. After executing the initial DoE, surrogate models of the performance and constraint metrics found in Table 14 were generated for use in the Monte Carlo probabilistic analysis to follow. An example of the ANN fit quality is shown in Figure 31 for the ESFC surrogate model regression, showing acceptable surrogate model error with respect to the output data from the physics based engine model DoE cases. Mean values of the model fit error (MFE) and model representation error (MRE) should lie close to zero, indicating that the surrogate model error is not skewed in one direction or another when comparing model estimates to both training and test data. In addition, the standard deviation of the model fit and representation error should be minimized, with values of one being the threshold of accepting a particular model. The actual-by-predicted and residual-by-predicted plots in Figure 31 serve to visualize the model error throughout the range of estimated values. The closer the training and test data points are to the perfect fit line in the actual-by-predicted plot, the better the regression can be assumed to be. Additionally, the random scattering of the data points around zero in the residual-by-predicted plot indicates the assumed functional form of the response model sufficiently captures the nonlinearity of the physics based model it represents.

Table 13: Design variables and corresponding ranges considered for ERDS case study.

Parameter	Description	Min Value	Max Value
OPR	Overall pressure ratio at TOC	16.6	30.0
$HPCPR$	High pressure compressor pressure ratio at TOC	2.5	5.5
$T_{41,max}$	Maximum turbine inlet temperature, $^{\circ}R$	2,650	3,450
s_{HP}	Design horsepower growth margin	1.0	1.5
$N_{s,HPC}$	Centrifugal compressor specific speed at TOC	0.4	0.8
N_{FPT}	Design speed of constant speed FPT	9,600	16,000

Table 14: Engine model output metrics for which surrogate models were trained as functions of control and noise variables.

Metric	Units	Type
ESFC at CRZ	$lbm/hr/eshp$	Evaluation
Maximum Nacelle Diameter	lbm	Evaluation
Engine Length	$inches$	Evaluation
Corrected HPC exit flow at SLS	lbm/s	Evaluation
Maximum HPC exit temperature	$^{\circ}R$	Constraint
Maximum HPC tip speed	ft/s	Constraint
Maximum AN^2 of HP, LP, and FP Turbines	$Billions\ of\ in^2 \cdot rpm^2$	Constraint

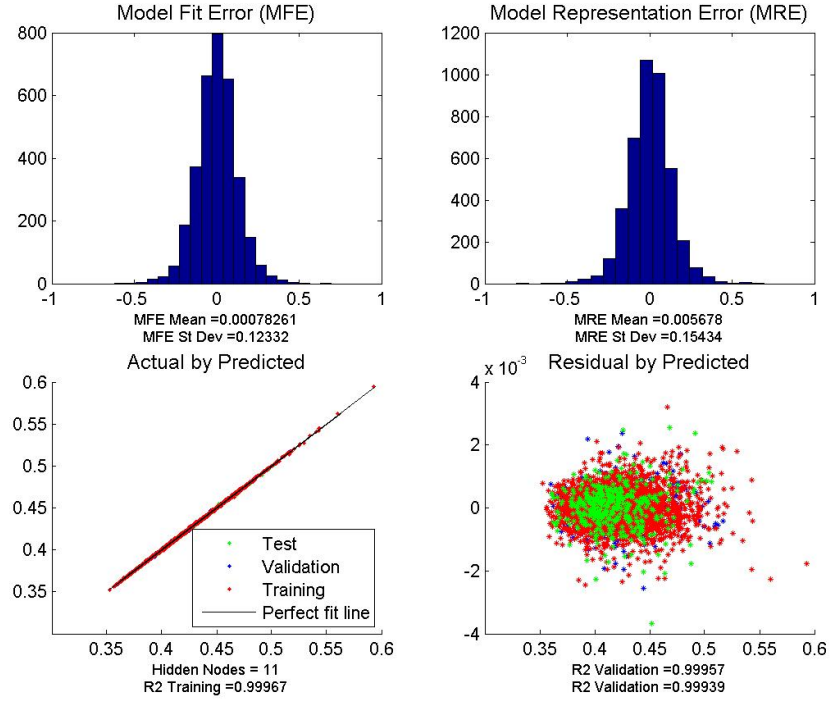


Figure 31: Artificial Neural Network surrogate model fit quality - cruise ESFC.

3.1.2.6 Monte Carlo Probabilistic Analysis

Using the surrogate model representations of the physics based engine model, Monte Carlo analysis of candidate engine designs in the presence of technology uncertainty can be performed. A second Design of Experiments is constructed solely for the design variable settings and technology uncertainty parameterization variables. Again, a space filling DoE is generated in order to explore the design space within the ranges of the six design variables considered for the present study. For the exploration within the six dimensional design space, 1,500 DoE cases were generated. As mentioned previously, if the designer is interested in pursuing a possible technology push and identify which technologies should be matured, the beta distribution shape factors should be included as control variables. Otherwise, the assumed prior distributions are fixed and implicit in the subsequent surrogate model training from the Monte Carlo analysis results.

Each case of the DoE represents a unique candidate engine design. In order to evaluate

how each candidate engine performs in the presence of technology impact uncertainty, many thousands of Monte Carlo simulations are performed and the pertinent surrogate models are evaluated for each simulation, following the process shown in Figure 27. For the present study, 10,000 Monte Carlo repetition simulations were performed for each of the 1,500 DoE cases. The noise variable settings for the simulations are sampled to represent the prior assumed uncertainty distributions, and the variability in the candidate engine's performance is then tracked. Quantile values of likely metric performance levels are calculated, allowing the designer to construct the cumulative distribution function of performance likelihood of each metric for every candidate design explored.

3.1.2.7 Probabilistic Data Regression

In order to allow the designer to richly sample the robust design space without the need to replicate each candidate design simulation, subsequent surrogate model regressions are generated. These probabilistic models are functions of the control and technology parameterization variables defining the uncertainty distributions under which robust setting selections are to be made. In order to account for the uncertainty implicit in the data collected from the Monte Carlo probabilistic analysis, the surrogate models are formulated to predict the likely metric values for a given vector of control variable settings and a corresponding desired level of confidence in the probabilistic metric prediction. For the present case study, no control is given to the shapes of the technology impact distributions, only design variable settings are explored and evaluated in order to simplify the explanation of the process.

Just as with the training of the first set of surrogate models representing the physics based engine model, a latin hypercube space filling design is used to sample the probabilistic design and technology space. Each sampled set of control parameter values represents a unique design, a set of technology impact distribution assumptions, and an inputted confidence interval for which the metric evaluations are estimated. Instead of having to evaluate each sample with the physics based engine model, the first set of surrogate models are used, allowing for many more input parameter settings to be explored within a given computational budget. This enables a richer sampling of the probabilistic design space,

allowing for more accurate training of the probabilistic surrogate models. The present study employed a latin hypercube design consisting of 19,500 samples throughout the control variable ranges.

An example of the probabilistic ANN surrogate model fit quality is shown in Figure 32. Again, the mean values of the MFE and MRE lie close to zero, indicating that the surrogate model error is not skewed in one direction or another. The standard deviation of the model fit and representation error is minimal. The actual-by-predicted plot indicates that the training and test data points lie close to the perfect fit line, and the random scattering of the data points around zero in the residual-by-predicted plot indicates the assumed functional form of the response model represents the probabilistic response throughout the design and technology space sampled. The surrogate models predicting engine length as well as the engine weight had the worst fit quality among the models trained. This is due to the models having to predict metric values that are affected by discrete engine changes, such as compressor or turbine stage counts. However, these models had fit quality levels that were acceptable as well. All probabilistic surrogate models generated were able to achieve acceptable fit quality, with error levels well below the threshold of rejecting the models.

3.1.2.8 Discussion of Case Study Results

The total computational clock time necessary to carry out the steps of the study up until this point was less than 5 total hours on a single quad-core machine. Simulations performed within the physics-based engine model contributed most to this, requiring about 3 total hours to carry out the necessary simulations for deterministic surrogate model training and testing. The remainder of assessments and surrogate regressions were performed in less than two hours. The ERDS process utilized for this study, while requiring additional steps upfront compared to traditional methods, greatly reduces the total amount of time required to produce the amount of probabilistic data as is shown, compared to competing methods.

The present case study sampled the candidate design space again using a space filling design consisting of 50,000 samples in combination with the probabilistic surrogate models. Constraints were imposed and only the remaining samples were considered feasible

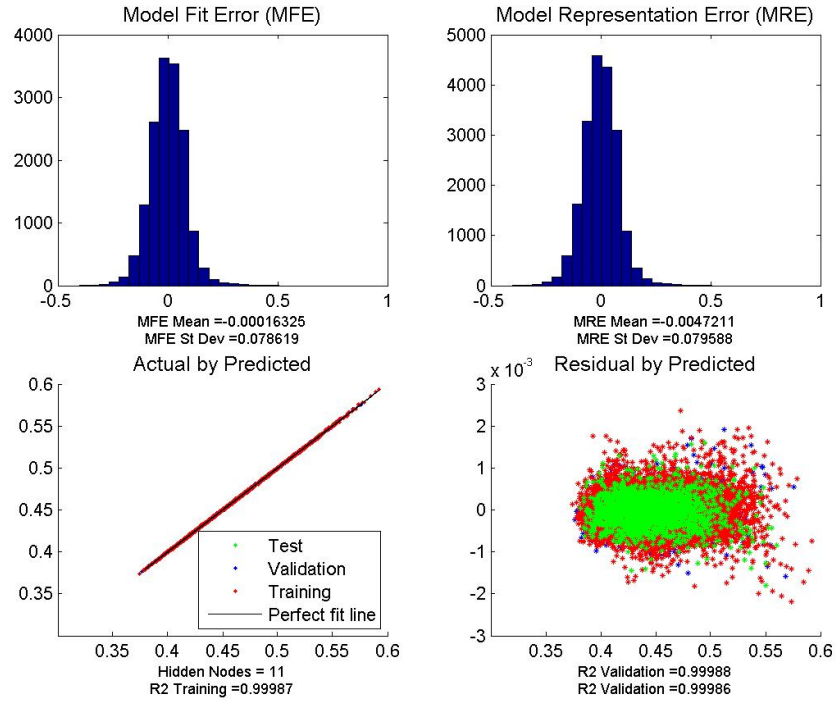


Figure 32: Probabilistic surrogate model fit quality - likely cruise ESFC.

candidate designs. Any design that had turbine AN^2 values above 50 (billion $in^2 \cdot rpm^2$) were filtered in order to maintain realistic turbine stress levels. This required regressions of likely maximum AN^2 levels of the high pressure, low pressure, and free power turbines. In addition to the turbine constraints, centrifugal compressor exit tip speed constraints were imposed to ensure that candidate designs satisfied the advanced Nickel material stress limitations, shown in Figure 29. This enabled the highest centrifugal compressor tip speeds for the high OPR designs. An uncertainty distribution was constructed around the assumed Nickel centrifugal compressor tip speed limits. This allows for robust design selections to be made accounting for the uncertainty in the actual tip speed limitations of the advanced Nickel material.

The candidate two-spool axi-centrifugal turboshaft engine design samples that had a 95% probability of satisfying the imposed constraints were examined. The feasible design space was identified within the full range of design variable settings. In Figures 33 - 36 design inputs are plotted against each other. For example, the intersection of $N_{s,HPC}$

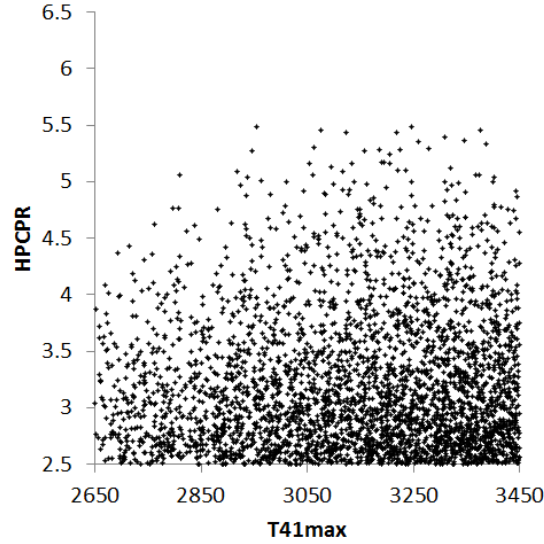


Figure 33: Feasible design centrifugal compressor pressure ratio settings for various max turbine inlet temperatures.

(centrifugal specific speed) and HPCPR it indicates that high HPCPR is inversely related to $N_{s,HPC}$ specific speed. As HPCPR increases, the diameter of the disk must also increase. As a result the rotational speed must be reduced in order to satisfy material constraints.

Apparent in Figures 33 - 36 are several obvious trends. First, higher values of HPCPR require higher values of $T_{41,max}$, as shown in Figure 33. As OPR increases, higher T41 is needed to maintain a smaller core to keep impeller stress within limits. Also high values of T_{41} enable higher speed, and therefore more compact and lighter, free power turbines. As the design turbine inlet temperature increases, the hot gas path flow area will decrease. Since the free power turbine speed is constrained by an AN^2 limit, smaller areas allow for higher speeds. Also worth noting is the correlation between HPCPR and s_{HP} (scale factor on baseline design horsepower). Larger power engines require larger mass flows. Examining Equation 17, this trend means that centrifugal compressor speed, and therefore pressure ratio, must be reduced to maintain a specified specific speed. As a result larger, more powerful engines shift more of the compression work to the axial stages, resulting in more efficient, lighter engines. The relationship between the amount of design power and feasible HPCPR levels is shown in Figure 34, identifying the tendency of lower HPCPR designs

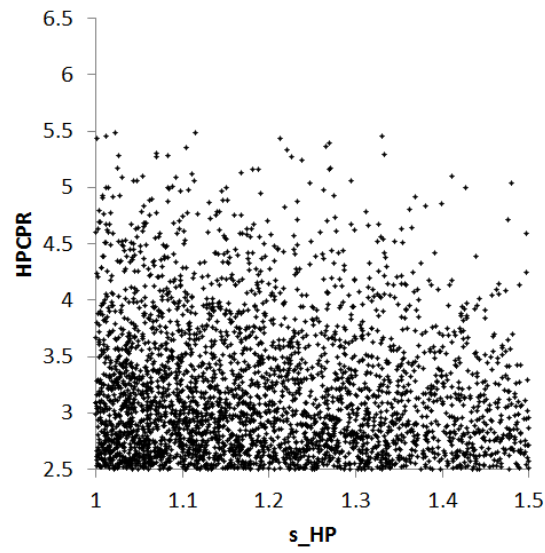


Figure 34: Feasible centrifugal compressor design pressure ratio settings for various design power levels.

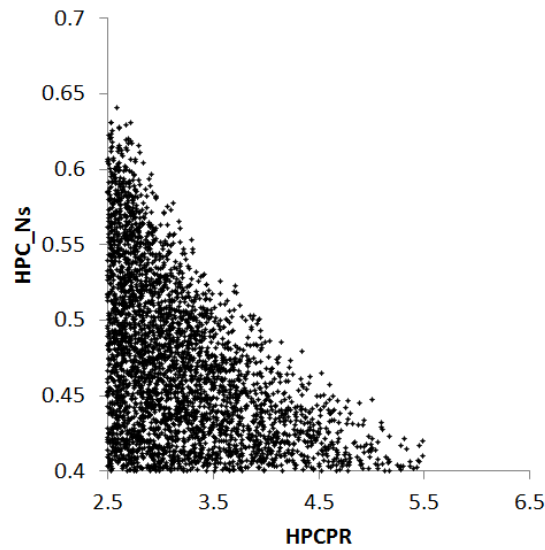


Figure 35: Feasible centrifugal compressor specific speed settings for various design compressor pressure ratios.

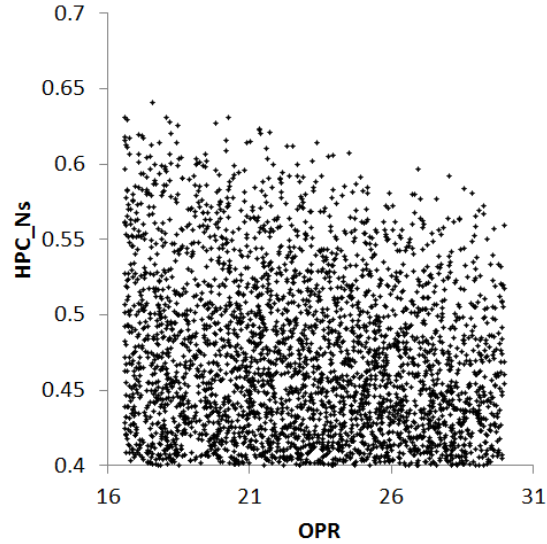


Figure 36: Feasible centrifugal compressor specific speed settings for various design overall pressure ratios.

offering more design feasibility and corresponding flexibility.

The major limitation in the design feasibility is in the attainable centrifugal compressor specific speed ($N_{s,HPC}$), which depends primarily on the design centrifugal compressor pressure ratio (HPCPR) and engine OPR, shown in Figure 35 and Figure 36. There is also a strong interaction between HPCPR and OPR, indicated in Figure 37. High HPCPR designs are only feasible with intermediate OPR designs. Selecting a high pressure ratio centrifugal compressor would greatly limit the design flexibility of the overall engine design due to the material limitations and corresponding achievable performance of the centrifugal compressor. These trends can be further explained by examining the impacts of the design centrifugal compressor pressure ratio on system performance, as shown in Figure 38 and Figure 39.

Figure 38 and Figure 39 show the variation in the likely performance due to the design choices made for the engine compression system. The plots show the 95% likely ESFC and weight estimates for each candidate design. In other words, for the settings of the design parameters indicated by HPCPR, OPR, and $N_{s,HPC}$, the plotted trends for ESFC and weight have an associated 95% confidence level given the assumed technology uncertainty

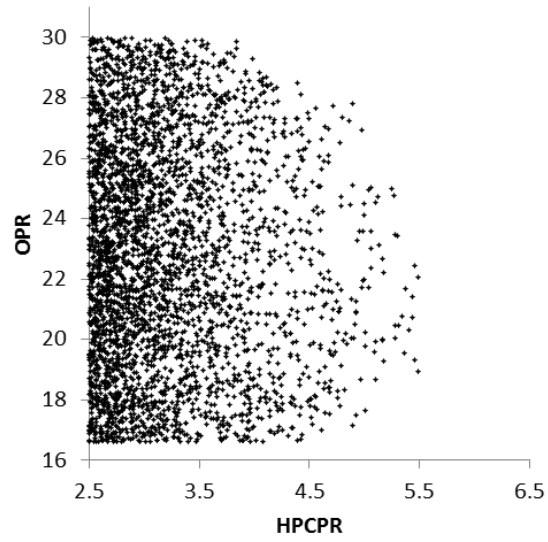


Figure 37: Feasible design overall pressure ratio settings for various design centrifugal compressor pressure ratios.

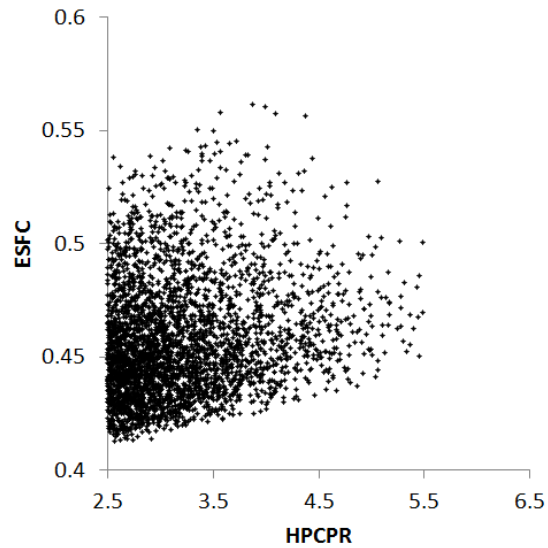


Figure 38: 95% likely ESFC levels for various centrifugal compressor pressure ratio designs.

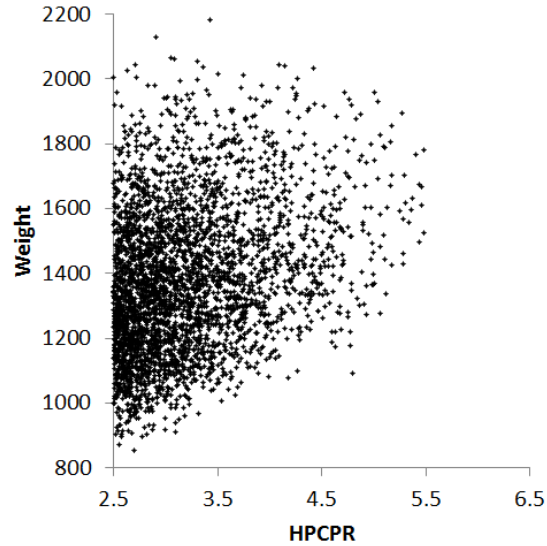


Figure 39: 95% likely engine weight levels for various centrifugal compressor pressure ratio designs.

distributions. There is strong correlation present between HPCPR and OPR, highlighted again by Figure 37. High HPCPR and low OPR combinations indicate that more of the compression is being done by the centrifugal system, leading to large diameter, heavier engines. This can be seen by the correlation between HPCPR and weight, shown in Figure 39. On the other end, high HPCPR, high OPR combinations drive up the exit temperature of the impeller. As a result these designs violate the material constraints imposed on the centrifugal compressor. The intersection of HPCPR and specific speed, depicted in Figure 35, shows that reducing specific speed can alleviate the high OPR, high HPCPR issues resulting from the high tip speeds; however, this leads to reduced compressor efficiency and engine cycle ESFC, shown in Figure 38. As specific speed is reduced, the centrifugal compressor moves away from its optimal size and efficiency suffers. Essentially there is a trade between centrifugal compressor speed, performance and weight that must be evaluated.

The turbine loading (AN^2) constraints also reduced feasible engine design space, although to a lesser extent. The maximum physical speed of the constant speed free power turbine is directly related to the FPT stress limit imposed. The HPT loading limit imposed normally limit the specific speed levels and centrifugal pressure ratios attainable. However,

Table 15: Case Study - Comparison of robust and deterministic candidate cycle ranges ($T_{41}=3,450^{\circ}R$).

Cycle Parameter	Min Weight		Compromise		Min ESFC	
	Det.	Prob.	Det.	Prob.	Det.	Prob.
$HPCPR$	2.5-3.0	2.5-2.6	2.5-3.0	2.5-2.7	2.5-3.0	2.5-2.7
s_{HP}	1.0-1.1	1.0-1.1	1.1-1.3	1.1-1.3	1.3-1.5	1.3-1.5
$N_{s,HPC}$	0.48-0.56	0.58-0.60	0.48-0.56	0.58-0.60	0.50-0.56	0.56-0.60
RPM_{FPT}	12,500-14,500	12,500-14,500	9,600-13,000	10,800-12,500	9,600-12,000	9,600-12,000
$\pi_{Overall}$	28-34	26-29	28-34	26-29	28-34	26-29

the centrifugal tip speed limits caused a greater limitation in the feasible $HPCPR / N_{s,HPC}$ combinations than the turbine loading constraint.

3.1.2.9 Robust Cycle Selection

Further analysis can be conducted using the data presented in Figures 33 - 39 to assess differences in using the RDS process and a more conventional, deterministic assessment. Since most of the points in these figures are non-optimal, further analysis can be performed to consider only the engine cycles that are near the pareto front in terms of weight and ESFC while meeting the constraints listed in Table 14. The downselection of candidate designs considered near the pareto front are assessed with deterministic evaluations of the candidate robust designs assuming the most likely impacts are realized for every single uncertainty distribution considered. It is important to view the optimum cycles along this front because the optimum solution may lie anywhere on the optimum weight vs. ESFC space depending on specific vehicle requirements.

The remaining candidate designs under consideration lie in three regions of the ESFC and weight metric space: ESFC optimal solutions, a balance between weight and ESFC, and weight optimal solutions. In general, for both the probabilistic and deterministic analyses, one-half percent of ESFC trades with approximately 400 *lbm* of engine weight. In order to better understand the data, ranges of cycle parameters for these three regions are summarized in Table 15.

Several interesting trends are noticeable when examining the cycle parameter ranges

between the deterministic and probabilistic (ERDS) analyses. First, most of the ranges for the cycle parameters on the pareto front are the same between the deterministic and probabilistic runs, except for HPCPR and OPR. The robust selections tend to favor lower OPRs and lower HPC pressure ratios. This trend is driven by uncertainty in the HPC material constraints, presented in Figure 29. Should the designer choose to pick a higher, albeit more efficient, OPR using the deterministic analysis, and the impeller material fails to realize the stresses placed on it by the higher temperatures and speeds, the deterministic case will end up worse off due to the mitigation actions necessary. Mitigation often manifests itself as weight or performance detriments. The robust cycles also have higher minimum HPC specific speeds. This also has the effect of minimizing impeller size and reducing sensitivity to not meeting HPC technology impacts. It should also be noted that the ranges on technology uncertainty will greatly affect how robust cycle ranges vary relative to deterministic ones.

Note that several pareto cycles selected with the deterministic analysis exceeded the maximum OPR of 30 originally sampled to produce surrogate representations of the physics based engine model. As mentioned earlier, extrapolation should be performed with caution when evaluating surrogate models. Therefore, it was ensured that the error imposed by extrapolating outside the original range of OPRs considered did not exceed an acceptable level. While ESFC prediction error was well within 1% of levels estimated by the physics based model, the weight estimation model under-predicted the weight on the order of 5% for engines with design OPR levels of 34, compared to what is estimated by the physics based engine model.

While the impact of uncertainty on cycle selection can be quantified, the real proof of the usefulness of the ERDS process must still be proven. If the RDS process does not select an engine that performs better when technology fails to mature then it is not worth the extra effort required to apply the process. In order to evaluate the robustness of cycles chosen using the RDS, the cycle settings corresponding to the pareto fronts for the deterministic (i.e., most likely) technology and probabilistic (i.e., 95% probability given technology distributions) impacts were then reassessed using the most likely technology and the worst

case technology assumptions. Note that depending on the technology parameter the worst case will either be the minimum or maximum value. For example, worst case efficiency would be the minimum and worst case weight impacts would be the maximum. Figure 40 shows the results of this study. There are two groups of pareto fronts. The blue and red cluster shows the deterministic and probabilistic cycles, respectively, when the worst case technology assumptions are applied. The yellow and green points show the deterministic and probabilistic pareto fronts when the most likely technology impact assumptions are used. In the worst case scenario, where all technology fails to mature, the probabilistic cycle space pareto front offers improved performance compared to the deterministic cycle pareto front. Conversely, should all technology programs achieve their goals and hit their likely targets, the deterministic cycle space will provide more performance and weight benefit than if the designer had chosen to go with the robust cycle selections. This highlights a fundamental choice between choosing a robust versus a deterministic optimum cycle. The robust cycle is less sensitive to technology impact variation, but will not outperform the deterministic cycles should all most likely technology impacts be realized in the resultant design. Looking more closely at the most likely technology impact data in Figure 40, the robust cycle offering the best ESFC performs an average of 0.3% worse, in terms of cruise ESFC, compared to the corresponding deterministic design for a given weight. For the worst case technology scenario, the robust cycle outperforms its corresponding deterministic counterpart by almost 0.8% for a fixed weight. Overall there is a 16% difference in ESFC and 300 *lbm* weight difference between the two technology scenario pareto fronts. Looking at it from this perspective, choosing the robust cycle allows the designer to gain back 5% of the potential losses from failing to meet technology goals while only sacrificing 2% of potential performance if all goals are met.

In reality not all technologies would fail to meet their goals and the exact differences in ESFC and weight depend on the vehicle application; however, following the ERDS can provide mitigation of the levels of risk associated with developing multiple technologies for a new engine platform.

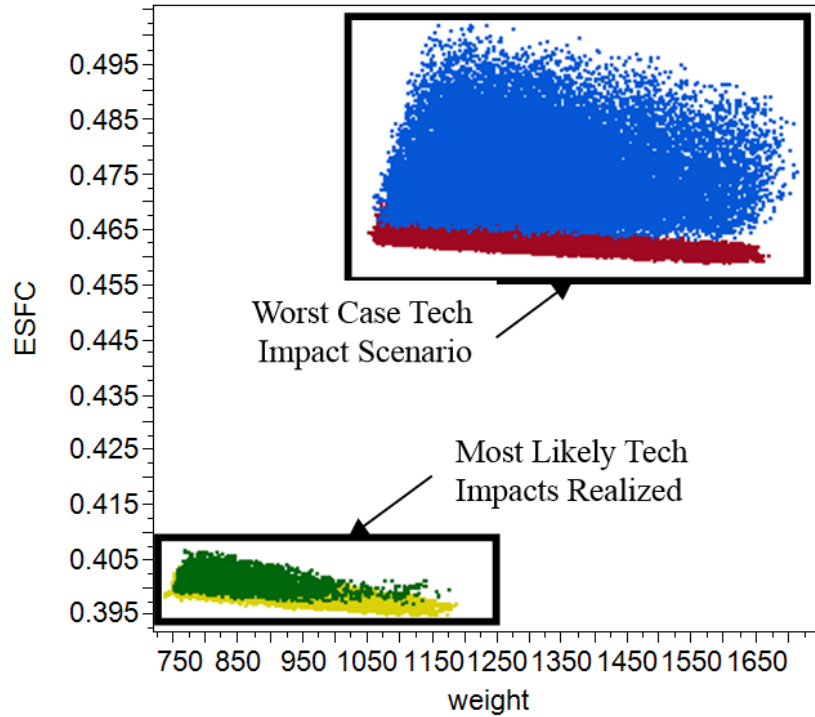


Figure 40: Evaluation of robust and deterministic candidate cycles against technology realization.

3.1.2.10 Conclusions

This case study presents a demonstration of the Enhanced Robust Design Simulation (ERDS) process. Two primary enhancements of the traditional method are presented. The first improves the ability of the designer to create accurate regressions representing probabilistic analysis data. Rather than train surrogate models to estimate the likelihood of achieving or surpassing discrete performance targets, models were trained to estimate performance levels as various desired levels of confidence. This offered a substantial improvement in the fit quality of models representing the probabilistic analysis data. The second enhancement to the traditional RDS method provides design freedom in the factors defining the otherwise fixed uncertainty distributions present in the RDS. Treating distribution shape factors as control variables in the RDS allows the designer to explore possible benefits of technology maturation to an engine program as well as the changes in resultant robust design selections due to maturation.

As applied to the turboshaft design problem, choosing a robust cycle using this process allowed a 5% recovery in performance assuming technologies failed to meet their stated goals and performance targets. Choosing a robust cycle means that it will tend to be more conservative. As a result there will be performance and weight penalties if all technologies meet their targets; however, this penalty is less than the performance recovered in the worst case. In the end it comes down to the level of risk that the designer is willing to accept.

3.1.3 ERDS Summary

The previous sections established the Enhanced Robust Design Simulation method as a way to efficiently explore and evaluate candidate designs under various sources of uncertainty. The ERDS method enables the more complex, multi-application common core design studies to be performed in the next chapter. The next section discusses the technique for enforcing geometric core commonality across engine applications. This development is another key contribution that also serves to enable the larger, fully integrated COMMENCE method.

3.2 Modeling of Geometrically Common Core Variant Designs

The COMMENCE method necessitates the enforcement of geometric commonality across N variant applications. A crude technique for enforcing commonality may be achieved through data filtering of simulation data. Such a technique would only be applicable when considering a small family of common core variant applications, as the probability of independently sampled data having consistent design HPC exit corrected flow would decay exponentially with the number of unique variant applications considered. This identifies the need for a modeling and simulation environment that enforces core commonality and allows for independent design exploration where permitted.

3.2.1 Modeling and Simulation of Common Core Application

In response to the identified needs, a modeling and simulation environment was created as a two-engine NPSS model, consisting of a core defining design engine and a common core variant engine. A representation of the environment is shown in Figure 41. For a given simulation within this environment, a design engine cycle is first converged upon just

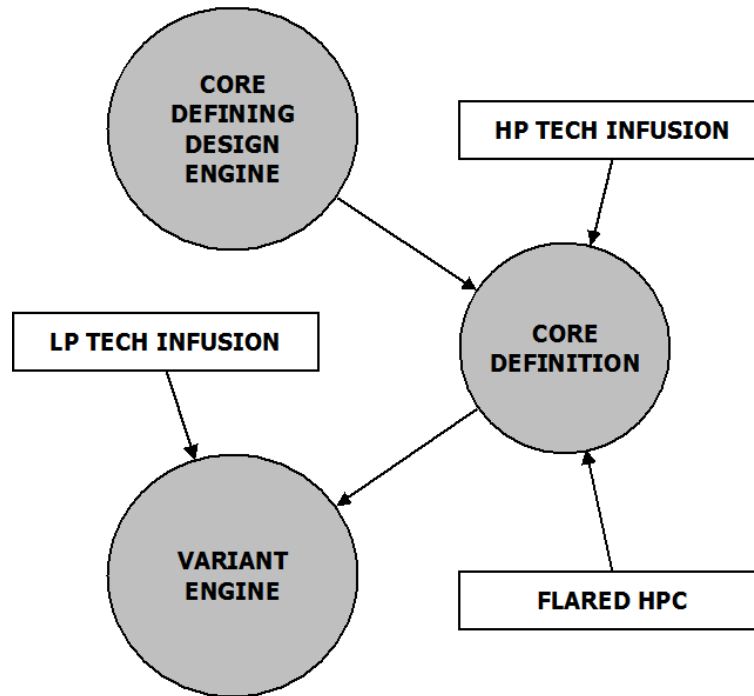


Figure 41: Common Core Variant Engine - Modeling and Simulation Environment Representation

as with any other gas turbine engine design. A set of desired input variable settings are provided, consisting of cycle parameter settings, design thrust levels, technology impacts, etc. With the set of input variable settings, the engine cycle solver attempts to converge upon a cycle representative of the inputted variable settings. The entire engine is allowed to scale in order to achieve the inputted thrust levels with the inputted cycle characteristics. This initial engine cycle simulation is used purely to establish an engine core definition that is to be applied to a variant engine design.

Once the core defining design engine is established, the core definition is frozen, no longer allowing for scaling of the core components. The core definition, made up of the HPC, the combustor, the HPT, and the ducting between these components, is now applied to a subsequent engine simulation within the same modeling and simulation environment. This variant engine again allows the components found outside of the core to be scaled. To prevent the common core definition from operating at a different characteristic conditions at the design point, several common core variant design rules are enforced. These allow not

only for geometric commonality to be enforced, but they also enable the baseline core definition to be upgraded for the common core variant design, just as performance improvement packages have been applied to common core designs in existence today.

3.2.2 Geometrically Common Core Applications: Variant Design Rules

By itself, the design-variant two-engine model would only allow for geometrically fixed common core applications to be simulated. Consideration of core upgrade options require a set of rules that allow for modification of the baseline core definition while concurrently enforcing commonality between the design and variant engine simulations. Recall that maintaining design HPC exit corrected flow across engine applications is an accepted way to simulate geometric commonality, resulting in applications having consistent compressor exit blade heights and performance characteristics. In addition, maintaining fixed design HPC exit corrected flow enables similar performance characteristics and flow capacity levels of the downstream core components: the combustor and the high pressure turbine.

If core upgrade options are of interest, whether they are advanced technology or upgrades in core flow capacity, accompanying rules are necessary to accurately enforce geometric commonality between the baseline core definition and upgraded common core variant. The additional rules ensure advanced core variant applications are simulated at the same HPC design map operating point as the baseline core definition.

A typical HPC map operating point is displayed in Figure 42. The design operating point on the map is defined by the corrected speed line and the R-line, also referred to as the Beta-line. At the design map operating point, the capabilities of the HPC are determined by the corrected flow and pressure ratio at that point. When designing a clean sheet engine, turbomachinery performance maps are scaled to represent the assumed technology level and capabilities of the new engine being designed. When simulating fixed engines, map scale factors are fixed, ensuring variation in map performance is due only to changes in the operation of the turbomachinery.

Selecting a baseline core definition and applying it to a variant application with or without upgrades requires a total of four variant design rules. Within the NPSS modeling

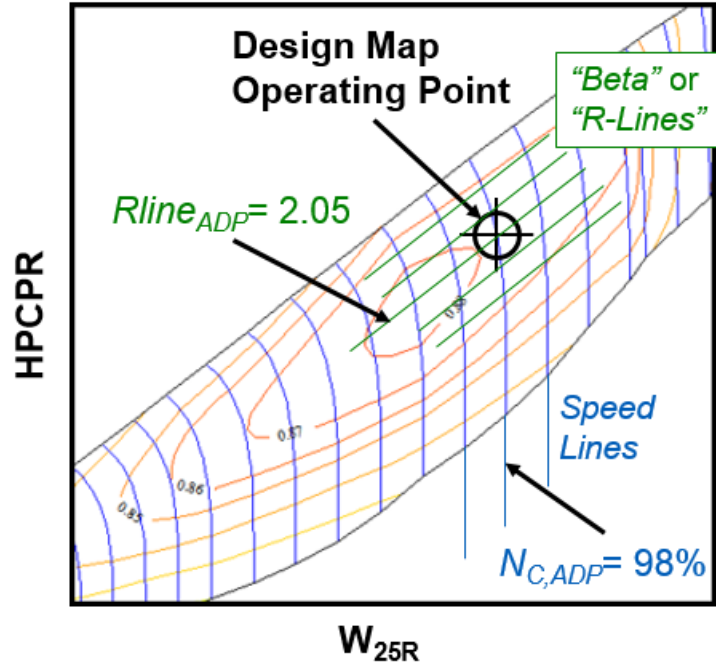


Figure 42: Common Core Variant Engine - Design Map Operating Point and Defining Parameters

and simulation environment used in the present work, NPSS solver pairs are used to enforce these rules. Equation 19 is used to set the corrected core inflow, whether at the baseline values or some increased core flow capacity when needed. When the HPC flow capacity is allowed increase, the map flow scalar is increased such that the corrected flow at the design operating point is achieved. Equation 20 enforces HPC exit design corrected flow across applications, accounting for changes in design inflow when allowed. As shown in the rule logic, maintaining consistent HPC exit corrected flow and a fixed HPC exit blade height when corrected core inflow is increased requires a higher HPC design pressure ratio.

$$\text{Vary : } \overbrace{sW_{C,HPCmap,Var}}^{\text{HPC Map Flow Scalar}} \quad (19a)$$

$$\text{To Satisfy : } \underbrace{W_{25R,Var,ADP}}_{\text{New Core Inflow}} = \underbrace{sW_{25R}}_{\text{DoE Input}} \times \underbrace{W_{25R,Des,ADP}}_{\text{Base Core Inflow}} \quad (19b)$$

$$\text{Vary} : \overbrace{sPR_{HPCmap,Var}}^{\text{HPC Map PR Scalar}} \quad (20a)$$

$$\text{To Satisfy} : \underbrace{W_{3R,Var,ADP}}_{\text{New HPC Exit Flow}} = \underbrace{W_{3R,Des,ADP}}_{\text{Base HPC Exit Flow}} \quad (20b)$$

The next rule ensures that commonality is enforced correctly by setting the variant design map operating point to that of the baseline core design. On a fixed HPC performance map, the map operating point can be fully defined by two parameter settings: the design corrected speed and the design R- or beta-line. The design corrected speed can be directly set, but in order to maintain the HPC design R-line, the HPT capacity is varied. Operating on the HPT map flow scalar is representative of changing the HPT flow capacity with either a vane-reset or a re-blading of the HPT. This rule to enforce the HPC design map operating point is found in Equation 21.

$$\text{Vary} : \overbrace{sW_{C,HPTmap,Var}}^{\text{HPT Map Flow Scalar}} \quad (21a)$$

$$\text{To Satisfy} : \underbrace{Rline_{HPCmap,Var,ADP}}_{\text{New Core Map Point}} = \underbrace{Rline_{HPCmap,Des,ADP}}_{\text{Base Core Map Point}} \quad (21b)$$

Now that the HPC design map operating point and flow capacity have been addressed, the issue of maintaining or allowing for an upgrade in the HPC efficiency can be addressed. Equation 22 is used to maintain the base HPC polytropic efficiency while also allowing for infusion of core efficiency technology into the variant engine simulation. Since HPC design pressure ratios are allowed to vary between common core variants, efficiency consistency is enforced on the polytropic efficiency instead of the adiabatic efficiency. This ties all common core applications to a consistent or upgraded technology level regardless of the applications' pressure ratios.

$$\text{Vary} : \overbrace{sEFF_{HPCmap,Var}}^{\text{HPC Map Efficiency Scalar}} \quad (22a)$$

$$\text{To Satisfy} : \underbrace{\eta_{p,HPC,Var,ADP}}_{\text{New Polytropic Efficiency}} = \underbrace{\eta_{p,HPC,Des,ADP}}_{\text{Base Polytropic Efficiency}} + \underbrace{\Delta\eta_{HPC}}_{\text{DoE Input Distribution}} \quad (22b)$$

By implementing these four solver pairs into the hybrid DESIGN/OFF-DESIGN variant engine simulation, geometric commonality is enforced between the two engine simulations

of the NPSS modeling and simulation environment. The NPSS code created to implement these rules and simulate the common core variant design engine can be found in Appendix A.

The effort involved in establishing a two-engine NPSS model where geometric commonality is enforced is a key enabler of the overall COMMENCE method. When performing design space explorations of both the core defining design engine and the common core variant design, commonality is now implicitly enforced. The relationships and common characteristics between the design and variant engine cannot be violated when sampling the two-engine NPSS model. Therefore, the resultant surrogate models produced from design samples of the physics based NPSS model will always have geometric commonality implicitly enforced. This prevents the need of post processing or filtering of assessment data in order to ensure commonality. Even for a single application design, if filtering was necessary, much of the engine simulations would likely be performed for naught. Needless to say, if commonality was not implicitly enforced and a very large number of common core applications were being considered, the likelihood of having geometric commonality across all applications would lie very close to zero.

The geometrically common core variant design modeling and simulation environment developed provides the COMMENCE method with a great advantage. Integration of this logic allows the integrated method to consider any number of common core applications, which would produce great amounts of probabilistic variant engine data, all tied implicitly to a single core defining design engine as the resultant common core definition.

3.3 Integrated Common Engine Core Evaluation (COMMENCE) Method

Now that the enabling capabilities have been established, they can be integrated into the COMMENCE method. The ERDS method that enables the efficient explorations of designs under uncertainty is used both in the selection of common core designs and is also used in the exploration and establishment of the benchmark designs against which the common core applications are evaluated. This ensures a consistent technique for evaluating the benchmarks and the common core applications, isolating the enforcement of geometric

commonality as the sole contributor of performance deviation from the benchmark design.

The common core variant design modeling and simulation environment is obviously used to explore the multi-application design space. Having relationships established between the design and variant engine offers the COMMENCE method the ability to apply the relationship to many common core applications that are all tied to a single core defining design engine.

The following sections offer descriptions and discussion of the steps of the fully integrated COMMENCE method. Explicit step-by-step procedures will be provided with the experiments that utilize the COMMENCE method in the following chapter.

3.3.1 Establishing the Target Market

- *Research Question 4: What range of capabilities can various common core design options achieve without significant compromises made in application performance?*

Hypothesis 4: A common core variant engine is able to provide a specific range of capabilities while maintaining acceptable performance levels, based on the technology level of the variant design, the core size, the overall engine architecture, and the amount of design freedom permitted for the particular common core application considered.

For an undertaking as substantial as a new centerline engine program, significant research must go into which segments of the gas turbine engine market products will be offered. Whether the organization is trying to enter into a new market or maintain their competitive edge in an existing segment, it must be ensured that the capital invested into the new engine program will pay off both during the initial release of products as well as throughout the life of the program. The first steps of the COMMENCE method, highlighted in Figure 43 address where within the gas turbine engine market the engine program will produce applications.

Establishing a strategy for the placement of an organization's product line that results in a successful and lasting program is crucial to the survival of an engine program and the organization as a whole. In the context of a new gas turbine engine program, determining the range of thrust and power levels for which engine applications will be developed contains this

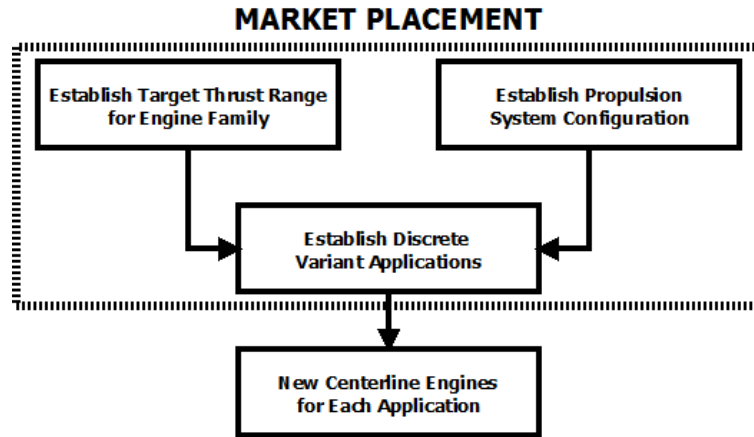


Figure 43: COMMENCE Method - Establishing Market Placement

crucial effort. Determination of these strategies is outside of the scope of the present work, but it is acknowledged that this portion making up this initial step of the COMMENCE method is of primary concern to any organization planning such an undertaking.

Another key factor in determining what the resultant family of engines will look like is the selection of the propulsion system configuration, or the overall engine architecture employed. The various engine companies typically adhere to their area of expertise as far as engine architectures are concerned, due to their vast experience on which they can lean on in the initial development of a new program. However, an advanced architecture may be pursued if the benefits outweigh the development cost of designing, testing, and utilizing a new overall engine architecture. Exploration of multiple engine architectures is also outside the scope of the present work. All engine applications considered presently will be two spool direct drive, separate flow turbofan architectures, and the method of evaluating and selecting candidate designs within a single engine architecture will be employed. It is worth noting that more complex engine architectures, containing additional engine shafts and/or gearing of the propulsor would likely offer the designer more flexibility when employing a common core across a variety of applications. The added complexity of the design, however, comes with additional uncertainty, so the tradeoff between added design flexibility and the accompanying uncertainty must be made.

The last task in establishing a new common core engine program is the consideration of unique sets of requirements for which applications will be designed. Consideration of new and existing customers with interest in new engines would be made in addition to consideration of the aging fleet of commercial and military engines. Entire departments within organizations work to gain customers for which engine applications are designed and employed. For the present work, arbitrary scenarios of unique sets of requirements will be used in order to explore the proposed method of evaluating and selecting common core designs.

The three initial steps of the COMMENCE method provide the method with sets of requirements for which a common core engine program will be designed and selected. These steps are necessary for the establishment of a successful strategy for a new common core engine program, which is why they are present in the proposed framework. But as mentioned above, options within these steps will not be presently explored.

3.3.2 Benchmark Engines for Applications Considered

After setting up the common core design problem and determining the unique sets of requirements for which engine applications are to be designed, benchmark engine designs are generated for each application considered, as shown in Figure 44. In order to determine the amount of performance compromise being made by employing a common core engine variant instead of a new centerline engine for a given application, a clean sheet design is used as a reference. This allows the designer to answer, given the projected technology level of the engine application time frame, *What is the maximum performance attainable if a new engine were designed specifically for this set of customer requirements?*

The selections made for each application can be performed in any traditional manner. However, the new centerline engine design explorations and selections in the present are selected by following the Enhanced Robust Design Simulation (ERDS) method procedure. This allows for consistent comparison between the common core applications and the benchmark engines. For each engine application considered, a candidate benchmark engine will be selected that has the highest TOPSIS score of the feasible candidate designs considered

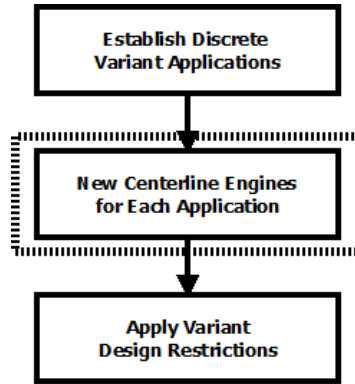


Figure 44: COMMENCE Method - Benchmark Engine Generation for Each Application Considered

for that application’s set of requirements. When common core engine variant designs are considered for a given application, the performance levels attainable will be compared to their corresponding benchmark engine under the same set of uncertainty assumptions.

Even though the benchmark designs are representative of clean sheet designs for each set of customer requirements, the common core modeling and simulation environment and its surrogate representations can still be used. The benchmark designs would just be simulated using the core defining design engine simulation. This highlights the flexibility and range of capabilities of the probabilistic surrogate models produced for this work. As long as the engine architecture, any assumptions made, and the input variable ranges of interest for various designs are enveloped within the surrogate training data, the resultant deterministic and surrogate models representing the core defining design engine can be utilized for the single application benchmark design explorations.

3.3.3 Engine Variant Design Options and Restrictions

- *Research Question 2: For a given gas turbine engine core, how should a common core engine variant design be simulated? What parameter(s) must be held to consistent values in order to maintain geometric and aerothermodynamic commonality between engine applications?*

Hypothesis 2: In order to simulate a common core engine variant, design rules must

be established and enforced that maintain the design level of corrected flow exiting the high pressure compressor. Maintaining HPC exit corrected flow at design conditions will ensure geometric similarity between common core applications. In order to provide significance to maintaining design HPC exit corrected flow, the design rules must also ensure that the compressor map design operating point is also fixed between common core applications.

- *Research Question 3: What design options should be considered for common core engine variant applications in order to distribute development capital across the engine program by taking advantage of commonality, while also offering more design freedom when needed for more demanding applications?*

Hypothesis 3: In order to provide a wide range of capabilities with common core applications, a range of design options should be available with differing levels of upgrade cost and design freedom. Geometrically fixed core and modified common core options should exist to allow for significant core power growth if needed. The common core applications should have design freedom in the LP system in order to be sized for a new set of customer requirements, and technology infusion should be considered for the core and/or the LP engine components in order to provide feasible, competitive common core solutions while placing preference on less expensive upgrade options when at all possible.

The primary advantage of utilizing a common core engine program is the ability to distribute development costs of the most expensive region of the engine: the high pressure, high temperature core. Maximization of the number of units sold would minimize the development recovery liability of each unit. Increases in the number of units likely to be sold can be achieved through the production of engine variant designs that aim to provide a variety of capabilities to various customers. The accompanying challenge of a common core program is being able to ensure that each unit's performance is not compromised, which would prevent the program from selling units for which development capital is invested to design and certify. This portion of the COMMENCE method is highlighted in Figure 45.

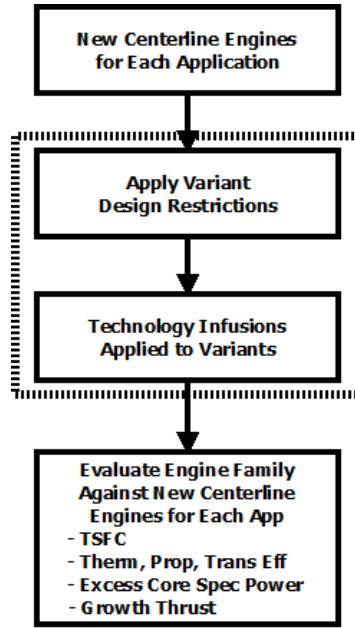


Figure 45: COMMENCE Method - Engine Variant Design Options

Table 16: Engine Variant Design Options - Matrix of Alternatives.

Category	Option 1	Option 2
Core Design	Geometrically Fixed Core	Modified Common Core
LP System Design	Fixed LP System	New LP System
LP Tech Level	Parent Tech Level	Technology Infusion
Core Tech Level	Parent Tech Level	Technology Infusion

The present work allows for certain design modifications to be made in order to provide a solution for a particular set of customer requirements and constraints. These design options have been compiled in the matrix of alternatives found in Table 16. In reality, variant engines will share similar sets of requirements and constraints while also having similar technology level low pressure systems. Therefore, in industry, when at all possible, off the shelf fan designs for example will be utilized if they meet the limitations imposed on their geometry and also are able to produce the required thrust with the airflow available to them. For all common core engine applications considered in this work, a new LP system will be designed to meet a unique set of customer requirements.

3.3.3.1 Common Core Variant Simulation

As was discussed in the section where the common core variant design rules were established, the modeling and simulation environment simulates two engines. The *core defining design engine* and the *common core variant engine*, utilize the same NPSS components making up their architectures. The core components, the ones shaded in Figure 46, are designed during the core defining design engine simulation, their characteristics are saved and then applied to the variant engine simulation. This enforces geometric core commonality between the design and variant engines. Table 17 contains the major NPSS components and their DESIGN or OFF-DESIGN modes during the simulations of the design and common core variant engine cycles. When in *DESIGN* mode, the components are allowed to be sized to exactly meet requirements imposed at the design points considered. Turbomachinery maps are scaled accordingly to provide the necessary flow and thrust levels at the various design points. When in *OFF-DESIGN* mode, the components are not allowed to scale, and they operate at speeds and corresponding efficiency levels that offer a steady state flow solution at the inputted engine power setting. The turbomachinery components use the same performance maps that were previously scaled during the core defining design engine run. Its operation is allowed to vary during off-design simulations, but the map is not allowed to scale. For the common core variant simulation, this is how the core components operate, even at design point conditions.

3.3.3.2 Geometrically Fixed Core with New LP System

When additional core working fluid is not needed to meet the requirements and/or constraints imposed on the variant application under consideration, the core definition will remain geometrically fixed, representing an off-the-shelf core application. Also, by remaining as common as possible, the knowledge of the existing core design can be carried over to the new engine application. Therefore, when simulating a geometrically fixed common core in the modeling and simulation environment with the imposed variant design rules, the term sW_{25R} will remain at a value of one, maintaining the design core inlet flow at ADP.

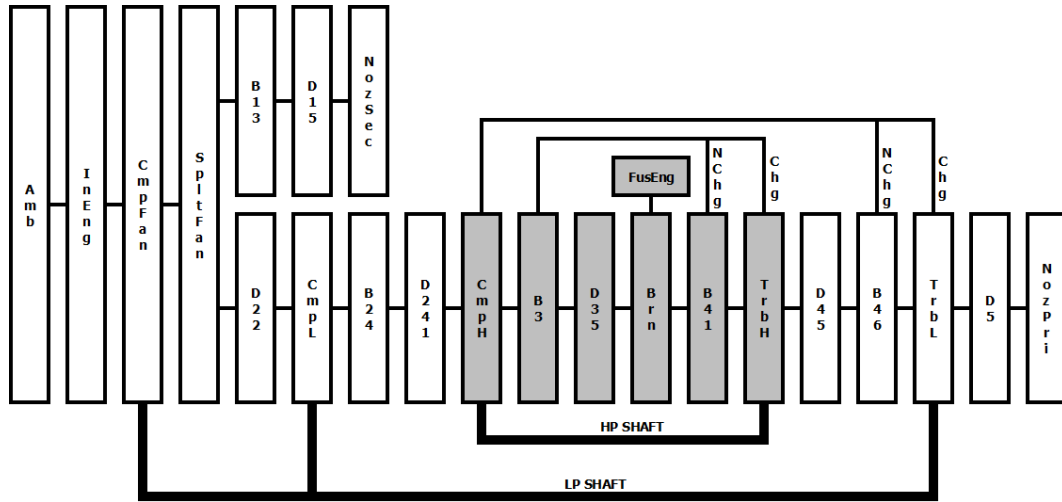


Figure 46: NPSS model components making up the physics based separate flow turbofan engine model, highlighting the engine core which remains geometrically common between design and variant engine applications.

Table 17: NPSS model components and their corresponding DESIGN or OFF-DESIGN status during Design and Variant engine simulations.

Component	Description	Design Sim Mode	Variant Sim Mode
InEng	Inlet	DESIGN	DESIGN
CmpFan	Fan	DESIGN	DESIGN
SpltFan	Splitter	DESIGN	DESIGN
NozSec	Bypass Nozzle	DESIGN	DESIGN
CmpL	LPC	DESIGN	DESIGN
TrbL	LPT	DESIGN	DESIGN
NozPri	Core Nozzle	DESIGN	DESIGN
ShL	LP Shaft	DESIGN	DESIGN
CmpH	HPC	DESIGN	OFF-DESIGN
Brn	Combustor	DESIGN	OFF-DESIGN
TrbH	HPT	DESIGN	OFF-DESIGN
ShH	HP Shaft	DESIGN	OFF-DESIGN

3.3.3.3 Modified Common Core with New LP System

When more core power is desired, but with as much core commonality as possible, the geometrically common core can be slightly modified to allow for more core working fluid. In order to increase core power the designer can either increase the operating temperatures of the core, increasing the thermal energy of the working fluid exiting the core components, or the core inlet flow can be increased, requiring an increase in the core overall pressure ratio at design. The goal of this design modification is to overcome the limitation of the fixed geometry core and increase the attainable core power beyond the capabilities of the previously sized definition. Modification of the power generator in order to increase the max amount of power gives the ability to generate increased power levels without surpassing imposed temperature limits.

Any modification to the core is an expensive design modification. Alteration of an existing core would likely be less of an undertaking compared to a completely new or photographically scaled core design. Increasing the design core flow is assumed to be accomplished through flaring of the HPC and/or zero-staging of the HPC, adding an additional rotor/stator compressor stage in front of the original core's first compressor stage. Flaring without the addition of a compressor stage is possible if the HPC was not originally sized to operate at max loading. However, if the necessary increase in the design overall pressure ratio in order to maintain the design level of HPC exit corrected flow at ADP cannot be achieved within HPC stage loading limits, then an additional compressor stage would be necessary. This is something to consider when making initial core design decisions. If future growth engines are projected to require significant core flow scaling compared to initial applications, the family of engines may benefit overall from a baseline core definition that is sized larger, with built in capabilities for increased loading for growth applications.

High flow growth applications would likely require reblading of the HPC due to achieve growth while being able to maintain high efficiency. Additionally, vane resets would likely be required to allow for HPT flow capacity to increase for growth applications. These modifications, although expensive, would aim to maintain core casing and flowpath geometry, requiring minimal modifications to be made to surrounding subsystems due to the fixed

casing geometry.

3.3.3.4 *Technology Infusion*

Just as with new engine products, new technology can be applied to existing engines through performance improvement packages. As targeted technologies become mature, they are applied to the various engines in operating and are built into new applications. In order to simulate such technology infusions, advanced technology impacts are applied to later projected engine applications, accounting for the projected maturation of an engine family's technology portfolio. Although somewhat trivial in its application within the COMMENCE method, the ability to account for future technological capabilities allows for more accurate simulation of later engine applications. As a result, the built in core size margins for future flow and power requirements can be minimized, allowing for technology development to fill the gap and achieve growth core power requirements.

3.3.4 **Common Core Variant Design Evaluation**

- *Research Question 5: How should a common core engine program consisting of multiple variant design applications be evaluated?*

Hypothesis 5: A common core variant engine should be evaluated based on the performance deviation from a benchmark, new centerline engine designed specifically for the application it is being designed for. In order to determine the amount of performance compromise made by utilizing a common core application, the benchmark engine should be designed for an identical set of requirements while under the same set of assumptions as were made for the common core variant design. A weighted sum of variant performance levels should be used in the overall evaluation of the engine program.

An engine's thrust specific fuel consumption (TSFC) is the most popular engine performance metric, relating the amount of fuel flow required per pound of net thrust achieved. TSFC is defined in Equation 23, usually at cruise conditions for some intermediate throttle

setting.

$$TSFC = \frac{\dot{m}_f}{F_{n,core} + F_{n,bypass}} \quad (23)$$

A candidate engine's overall efficiency (η_O) can be defined as a product of the engine's propulsive, thermal, and transmission efficiency at some condition, as shown in Equation 24. A typical condition for evaluating an engine's efficiency is at the aerodynamic design point (ADP), usually at cruise conditions and at full throttle. This design point is where the engine components are designed for maximum efficiency.

$$\eta_O = \eta_P \eta_T \eta_{Tr} \quad (24)$$

Kestner et al.[51] provide a convenient way of defining each of the engine efficiency terms in terms of *total available* kinetic energy levels, expanding all flow velocities to ambient conditions. This essentially converts the internal energy, or enthalpy, at each flow station into additional *available* kinetic energy. This approach is also convenient for studies interested in engine core performance, because it maps core performance to overall engine performance. The kinetic energy addition rate term ($\dot{K}E_{core}$ is defined in Equation 25, and the measurement for the overall engine $\dot{K}E_{engine}$) is defined in Equation 26.

$$\dot{K}E_{core} = \frac{1}{2} \left[\dot{m}_{43} v_{43}^2 - \dot{m}_{25} v_{25}^2 \right] \quad (25)$$

$$\dot{K}E_{engine} = \frac{1}{2} \left[\dot{m}_7 v_7^2 + \dot{m}_{17} v_{17}^2 - \dot{m}_2 v_2^2 \right] \quad (26)$$

The thermal energy addition rate terms (\dot{E}_{fuel} , $\dot{E}_{propulsive}$) required for efficiency calculation are defined in Equation 27 and Equation 28.

$$\dot{E}_{fuel} = \dot{m}_f LHV \quad (27)$$

$$\dot{E}_{propulsive} = \left[F_{n,core} + F_{n,bypass} \right] v_\infty \quad (28)$$

The propulsive efficiency (η_P) of the engine is defined in Equation 29, relating the propulsive energy addition by the engine to the kinetic energy added to the flow traveling through the engine.

$$\eta_P = \frac{\dot{E}_{propulsive}}{\dot{K}E_{engine}} = \frac{[F_{n,core} + F_{n,bypass}]v_\infty}{\frac{1}{2}[\dot{m}_7v_7^2 + \dot{m}_{17}v_{17}^2 - \dot{m}_2v_2^2]} \quad (29)$$

The thermal efficiency (η_T) of the engine is ratio of the resultant core energy addition to the thermal energy added to the core flow, and is defined in Equation 30.

$$\eta_T = \frac{\dot{K}E_{core}}{\dot{E}_{fuel}} = \frac{\frac{1}{2}[\dot{m}_{43}v_{43}^2 - \dot{m}_{25}v_{25}^2]}{\dot{m}_fLHV} \quad (30)$$

The transmission efficiency (η_{Tr}) term relates the energy addition to the core to the achieved energy addition by the engine to the engine gas flow, and is defined in Equation 31.

$$\eta_{Tr} = \frac{\dot{K}E_{engine}}{\dot{K}E_{core}} = \frac{\dot{m}_7v_7^2 + \dot{m}_{17}v_{17}^2 - \dot{m}_2v_2^2}{\dot{m}_{43}v_{43}^2 - \dot{m}_{25}v_{25}^2} \quad (31)$$

A useful metric that will assist in estimating how much thrust growth is available with an engine core is the *specific power* of the engine core, denoted as $P_{S,core}$. By measuring the specific power of the engine core while operating at the rated maximum temperature and pressure, the designer can determine how much additional specific power can be produced by the core when operating at the absolute maximum temperature and flow levels possible. The specific power of the engine at maximum power, usually measured at takeoff, can be defined using Equation 32. The available power produced by the core, equivalent to the available kinetic energy of the core defined above, is normalized by the core air flow entering the HPC. The normalized metric uses the HPC entrance mass flow because this contains both the working flow that will travel through the combustor as well as the cooling flow required for engine operation at the power level reached. This available specific power of the core is evaluated with no customer bleed or horsepower extraction from the engine. This will show the maximum specific power available at current operating conditions, of which the customer can reduce if bleed and power extraction are necessary.

$$P_{S,core} = \frac{\dot{K}E_{core}}{\dot{m}_{25}} = \frac{\frac{1}{2}[\dot{m}_{43}v_{43}^2 - \dot{m}_{25}v_{25}^2]}{\dot{m}_{25}} \quad (32)$$

It is helpful to define \dot{m}_{43} , the mass flow exiting the HPT in terms of \dot{m}_{25} . Equation 33 does this by also accounting for the fuel flow rate and any LPT cooling flow required from the HPC.

$$\dot{m}_{43} = \dot{m}_{25} + \dot{m}_f - \dot{m}_{cool,LPT} = \dot{m}_{25} \left[1 + \frac{\dot{m}_f}{\dot{m}_{25}} - \frac{\dot{m}_{cool,LPT}}{\dot{m}_{25}} \right] \quad (33)$$

Now the maximum specific power available can be restated using Equation 34.

$$P_{S,core} = \frac{\dot{K}E_{core}}{\dot{m}_{25}} = \frac{1}{2} \left[\left(1 + \frac{\dot{m}_f}{\dot{m}_{25}} - \frac{\dot{m}_{cool,LPT}}{\dot{m}_{25}} \right) v_{43}^2 - v_{25}^2 \right] \quad (34)$$

For a candidate engine core design, the maximum specific core power achievable can be found by calculating the specific core power when operating the core at the maximum possible temperature and air flow levels. The maximum achievable specific power and the corresponding excess specific core power can be found using Equation 35 and Equation 36.

$$(P_{S,core})_{max} = P_{S,core}(T_{3,max}, T_{4,max}, T_{41,max}, \dot{m}_{25,max}) \quad (35)$$

$$(P_{S,core})_{excess} = (P_{S,core})_{max} - P_{S,core} \quad (36)$$

In a similar fashion to how the current specific power was related to the maximum specific power achievable with a candidate engine core design, the growth potential can be determined for a given engine core. Growth potential is primarily concerned with the ability to achieve greater levels of thrust, both at the current technology level as well as with the performance improvement packages later in the engine program life. The maximum specific power of a fixed core design at the current technology level is determined based on the core geometry, achievable turbomachinery component efficiencies, cooling capabilities, and current temperature limits imposed on the design. However, the maximum specific power of an engine core design will increase as technologies are developed that improve component efficiencies, cooling capabilities, and temperature limits. Therefore, a *current* growth potential as well as a projected future growth potential can be found, mapping the maximum specific core power to the achievable thrust levels.

This mapping from maximum specific core power to achievable thrust growth can be done using the predefined efficiency terms, all evaluated at the maximum power design

point, usually the takeoff condition. The current maximum possible core kinetic energy addition at takeoff is found using Equation 37

$$(\dot{K}E_{core})_{max} = (P_{S,core})_{max}\dot{m}_{25,max} \quad (37)$$

With the values of transmission and propulsive efficiency evaluated when determining the maximum achievable specific power, the maximum propulsive energy addition achievable at the current technology level can now be found using Equation 38.

$$(\dot{E}_{prop})_{max} = \eta_P\eta_{Tr}(\dot{K}E_{core})_{max} \quad (38)$$

The resultant maximum achievable sustained thrust at takeoff can be found using Equation 39.

$$(F_n)_{max} = \frac{(\dot{E}_{prop})_{max}}{v_\infty} \quad (39)$$

This maximum achievable takeoff thrust is the growth potential of the common core engine. Whether the current or projected level is determined is based on the component efficiencies, cooling capabilities, and temperature limits used for the calculation of this growth potential.

Depending on the performance metrics of interest, common core variant applications can be evaluated. The amount of deviation in performance of each common core variant application can be determined. The individual application performance evaluations can then be combined to perform an overall common core engine family evaluation. This is shown in the following section.

3.3.5 Overall Engine Program Exploration and Evaluation

- *Primary Research Question: How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?*

Primary Hypothesis: Simultaneous simulation and evaluation of current and future common core applications should be performed when exploring the common core design space in order to quantitatively estimate the feasibility and likely performance levels of

program applications due to changes in the common core definition. If this is possible and implemented, the likelihood of achieving feasible, competitive common core variant designs will be increased while minimizing the amount of mitigation actions required later in the program.

Evaluation of how a common core definition performs throughout a range of variant engines aiming at meeting various requirements while having possible different technology limits will allow for a better selection of an engine program's common core design. Using the Performance Deviation Index (PDI) established by Simpson[104], the family of common core engine applications can be evaluated against each corresponding benchmark application. Evaluation of a single common core application is performed by taking the weighted sum of the deviations of the performance metrics of interest. In other words, the most desirable candidate designs are those that lie closest in performance to the corresponding benchmarks established for each application. An example of this is shown in Equation 40, where TSFC and engine weight are the two metrics of interest. Once each application is evaluated, the entire product line can be evaluated by taking the weighted sum of all applications' performance deviation levels, with increased weighting factors applied to the applications less compromised performance is more desired. This overall evaluation of a product line of common core evaluations is depicted in Equation 41.

$$y_i = \left(\frac{TSFC_{i,CC} - TSFC_{i^*}}{TSFC_{i^*}} \right) w_{TSFC} + \left(\frac{Wt_{i,CC} - Wt_{i^*}}{Wt_{i^*}} \right) w_{Wt} \quad (40)$$

$$Y_{CC} = \sum_{i=1}^{N_{apps}} w_i y_i \quad (41)$$

Relationships between the core definition and corresponding common core variant performance levels can be established after various candidate designs have been considered. In order to consider realistic core designs, an initial core defining design engine is explored. This avoids the need to independently vary design parameters related to an engine core geometry. This latter type of design exploration would likely result in many candidate cores considered being far from a realistic core definition.

Table 18: Typical control variables and corresponding ranges for a clean sheet turbofan engine design exploration.

Variable	Units	Min	Max
π_{Fan} @ ADP	-	1.3	1.7
π_{HPC} @ ADP	-	9.0	16.0
$\pi_{Overall}$ @ ADP	-	30.0	55.0
T_4 @ TKO	$^{\circ}R$	3,300	3,750
ER @ ADP	-	0.90	1.25
$NChg_{LPT}$	% W_{25}	0.00	0.06
F_N @ TKO	<i>lbf</i>	20,000	30,000
F_N @ TOC	<i>lbf</i>	5,900	8,850

Table 19: Control variables and corresponding ranges for variant engine exploration.

Variable	Units	Min	Max
$\pi_{Fan,Var}$ @ ADP	-	1.3	1.7
$\pi_{Overall,Var}$ @ ADP	-	30.0	55.0
$T_{4,Var}$ @ TKO	$^{\circ}R$	3,300	3,750
ER_{Var} @ ADP	-	0.90	1.25
$NChg_{LPT,Var}$	% W_{25}	0.00	0.06
sW_{25R} @ ADP	-	1.0	1.2
$N_{C,HPT,Var}$ @ TKO	%	100.	105.

Sampling of a design engine cycle is used to define the core design to be applied across common core variant applications. Such design cycle explorations can be done within a typical cycle design space as contained within the design ranges listed in Table 18. Then, for each independent exploratory core definition, a unique variant cycle design sample is considered within the design space contained in the variant cycle ranges listed in Table 19. The resultant core definition and variant cycle application is then simulated.

In the evaluation of both the common core variant designs as well as the overall common core program evaluation, the ERDS method is utilized, allowing for efficient exploration and evaluation of candidate common core variant designs under the assumed uncertainty scenario considered. Probabilistic performance predicting surrogate models are trained with the physics based engine model, and extensive explorations are then able to be more richly performed with the use of the surrogate models. Likely performance impacts of common core

variant designs are estimated based on the core defining engine control variable settings and uncertainty scenario shape parameter settings, the common core variant design variables and uncertainty scenario shape parameter settings, and the inputted level of confidence desired in the performance estimate(s).

The first priority when evaluating candidate common core program designs is the ability to have feasible variant designs for all applications considered. Once a high likelihood of achieving feasibility is achieved across all common core applications, TOPSIS evaluations are then used to evaluate the remaining candidate designs, ranking them based on the desired metrics and accompanying importance of each. The TOPSIS scores of candidate designs will be based on how close in performance the common core applications are with respect to their corresponding benchmark engines. The weighted sum of each common core application's TOPSIS performance measurement is then computed. The overall common core engine family then receives a TOPSIS score and rank within the population of feasible candidate common core families.

3.4 Summary of COMMENCE Method Development

The previous sections establish the COMMENCE method, the overall process, and the enabling techniques making the COMMENCE method a viable option for exploring common core engine family designs.

When considering a new common core engine family, the design-variant common core modeling and simulation environment is first developed to represent the types of engines considered and the relationship between the baseline core definition and a variant application. The physics based model is vetted, establishing a baseline collector onto which technology advancements can be applied. Simulations are performed throughout the design-variant design and technology space with the use of a DoE allowing for the most efficient use of the simulation budget. With the resultant data, two sets of deterministic surrogate representations of the two-engine physics based model are produced: One set represents the core defining design engine where each model takes the form in Equation 42, while the other set of models take the form in Equation 43 to represent the geometrically common core variant

engine.

$$Y_{Des} = f(\mathbf{CV}_{Des}, \mathbf{NV}_{Des}) \quad (42)$$

$$Y_{Var} = f(\mathbf{CV}_{Des}, \mathbf{NV}_{Des}, \mathbf{CV}_{Var}, \mathbf{NV}_{Var}) \quad (43)$$

Probabilistic assessments are then performed, accounting for any uncertainty scenarios that may be of interest. Another round of surrogate models are then regressed. These probabilistic models estimate likely performance levels at inputted confidence intervals for the design and common core variant engines under various uncertainty scenarios. The two sets of probabilistic surrogate models are functions of design variables, uncertainty distribution shape parameters, and the desired level of confidence associated with the metric response. The design and variant engine representations take the forms in Equation 44 and Equation 45.

$$Y_{Des} = f(C, \mathbf{CV}_{Des}, \mathbf{A}_{Des}, \mathbf{B}_{Des}, \mathbf{k}_{minmax,Des}) \quad (44)$$

$$Y_{Var} = f(C, \mathbf{CV}_{Des}, \mathbf{A}_{Des}, \mathbf{B}_{Des}, \mathbf{k}_{minmax,Des}, \mathbf{CV}_{Var}, \mathbf{A}_{Var}, \mathbf{B}_{Var}, \mathbf{k}_{minmax,Var}) \quad (45)$$

Once the probabilistic surrogate models are produced, the designer can perform design and technology space explorations. When integrated into the COMMENCE method, the probabilistic models are used for the explorations and selections of benchmark engine designs as well as for exploration of the multiple application common core engine design and technology space.

By investing the additional time required upfront to produce the resultant sets of probabilistic surrogate models, significant payoffs can be realized later in a development program. As long as the designer accounts for all possible variables and ranges that may be considered later, a valid set of probabilistic representations of the common core design and uncertainty space can be applied to a wide variety of deterministic or robust design exercises. Single application design problems, such as those to establish benchmark engines, can be considered with the use of the probabilistic representations of the core defining design engines.

The true power of the fully integrated method and resulting probabilistic representations is the enabled capability to explore the high dimensional, multiple-application common core design space under various uncertainty scenarios.

The first three experiments found in the following chapter are formulated to test the various developments made to enable the larger COMMENCE method. The first experiment tests the ERDS method's ability to select robust designs. The second experiment tests the common core modeling and simulation environment's ability to enforce commonality and apply upgrades to common core applications. The third experiment considers the common core design for a small set of engine applications, testing the COMMENCE method's ability to handle multiple sets of requirements all under the same uncertainty scenario.

The final experiment exercises the fully integrated COMMENCE method to perform a multi-application, robust common core design study that previously would be computationally infeasible to carry out. A large number of engine applications are considered, each having a unique uncertainty scenario. Geometric core commonality is implicitly enforced across all applications considered. The common core design space is explored with the goal of finding a common core design region that offers feasible variant engines for all applications considered. Demonstration of enabling capabilities will be presented as well, highlighting the range of studies that are enabled by the development and full integration of the COMMENCE method.

CHAPTER IV

EXPERIMENTAL IMPLEMENTATION AND RESULTS

The following experiments will be used to test the hypotheses established in the previous chapters. This will allow for the primary research question to be answered and corresponding hypothesis to be tested. The organization of the chapter serves as a stack-up of capabilities. Each subsequent experiment builds upon the previous established capabilities in order to reach the overarching goal of evaluating a common core design selection based on the overall engine program's performance.

Primary Research Question: *How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?*

Hypothesis: *Simultaneous simulation and evaluation of current and future common core applications should be performed when exploring the common core design space in order to quantitatively estimate the feasibility and likely performance levels of program applications due to changes in the common core definition. If this is possible and implemented, the likelihood of achieving feasible, competitive common core variant designs will be increased while minimizing the amount of mitigation actions required later in the program.*

In order to achieve the research objective, causality must be found that relates the engine core design decisions made early in a common core gas turbine engine program to the ability of the program to competitively offer solutions for current and future applications.

Research Objective: *Develop a gas turbine engine design and decision making process that aims to increase the useful competitive life and overall versatility of a common core engine family. The process should consider current and future competitive engine*

family performance, utilizing current and eventual technology improvements without the need for a core re-design.

The process framework, when used with virtual experimentation, offers the construction of a relationship to be established between engine core architecture and size selections to the resulting engine family performance across a range of capabilities and technology scenarios.

The first experiment serves to establish and test the capabilities of the ERDS method. A single application engine design problem will be used. A more traditional, deterministic design method will be used to explore and select a candidate design. Then, the ERDS method will be applied to the same design. Comparison of the favorable designs and the resulting likely performance levels of candidate selections made with each method will be compared.

The second experiment tests the common core modeling and simulation environment. Various common core upgrade options are explored with the goal of determining the maximum growth thrust achievable with the various options explored.

The third experiment is the first multiple application common core design problem posed. Three engine applications, each having unique thrust requirements are considered, all having the same technology level. Investigation into common core feasibility across multiple applications is performed, and selection of the common core design that provides the smallest total performance deviation with respect to benchmark engines established for each set of requirements.

The final experiment tests the suite of capabilities enabled by the fully integrated COM-MENCE method. Seven engine applications are considered, each having a unique uncertainty scenario. Exploration of the common core design space will be performed, aiming for a common core solution that provides feasible common core variants for all applications considered.

4.1 Training of Deterministic and Probabilistic Surrogate Models for Experiments

One key advantage of the ERDS method and the fully integrated COMMENCE method is the enabled flexibility of the probabilistic surrogate models produced. As was mentioned in Chapter 3, great care should be taken when producing training data for surrogate model regression. The number of input variables and their associated ranges should be selected such that a wide variety of probabilistic assessments can be performed with a single set of surrogate models.

The ERDS process contains the steps by which surrogate models should be trained to enable a wide variety of robust design probabilistic assessments. The process is now described in detail, providing the explicit steps that were carried out to produce the set of surrogate models used for all of the experiments to follow. The steps are consistent with the ones presented in the Chapter 3 discussion of the ERDS process, with added details intended to provide transparency and allow for process repeatability.

4.1.1 Step 1: Physics based model preparation

The baseline engine cycle definition, onto which technologies are applied and from which common core variant applications are simulated, is representative of a middle-of-market high bypass turbofan engine such as the CFM56-7B27. The baseline definition listed in Table 30 and visualized in Figure 51 serves as the baseline definition for all experiments discussed and performed in this chapter.

As was discussed in Chapter 3, logic must be added to the NPSS modeling and simulation environment to allow for the simulation of a core defining design engine and subsequent simulation of a common core engine variant. The NPSS code created by this author to enforce commonality and simulate a common core variant design can be found in Appendix A. The steps that are carried out in the model are presented here in words.

Two sets of input variables are first established, identifying all variables that are to be varied in all experiments to come. The first set of input variables apply to the core defining design engine. The second set of input variables correspond to the common core

Table 20: Core Defining Design Engine - Design Variable Ranges

Variable	Description	Lower Limit	Upper Limit
π_{Fan} at ADP	Design Fan Pressure Ratio	1.3	1.7
π_{HPC} at ADP	Design HPC Pressure Ratio	9.0	16.0
$\pi_{Overall}$ at ADP	Design Overall Pressure Ratio	30.0	55.0
$T_{4,max}$ (<i>degR</i>)	Maximum Combustor Exit Temperature	3,300	3,750
ER at ADP	Design Extraction Ratio	0.9	1.25
F_N at TKO (<i>lbf</i>)	Net Thrust at Takeoff	20,000	30,000
F_N at TOC (<i>lbf</i>)	Net Thrust at Top of Climb	5,900	8,850

engine variant engine. These sets of input variables first consist of design variables (also referred to as control variables) whose values will be selected when making design decisions. In addition to the inclusion of all possible design variables of interest, all possible technology, installation, and other noise variables that are to be accounted for in the following experiments should be included in the superset of input variables to explore.

For each of the input variables considered for all subsequent experiments, variable ranges must be established that envelop all possible values that may be used to evaluate a design. A total of 36 input variables corresponding to the core defining design engine are assigned ranges. The seven design variables for the core defining engine are listed in Table 20 with their corresponding desired ranges. To avoid extrapolation of the resultant surrogate models when evaluating designs near the edges of the input variable space, all variable ranges are extended on either side by one percent of the desired range. For example, the fan pressure ratio (FPR) has a desired range from 1.3 to 1.7. The actual ranges used to sample the physics based model for surrogate training vary from 1.296 to 1.704. This ensures that any future surrogate evaluations will have input variable values within the input region where the surrogate model accurately represents the physics based model.

In addition to the seven core defining engine design variables, ranges were applied to all variables whose values are assumed to be uncertain in one or more experiments to follow. Table 21, Table 22, and Table 23 contain the variables and corresponding ranges to account for during surrogate training.

Table 21: Core Defining Design Engine - Input Variables for Component Efficiency Impacts

Variable	Description	Lower Limit	Upper Limit
$\Delta\eta_{Fan}$	Delta Fan Efficiency	0.00	0.01
$\Delta\eta_{LPC}$	Delta LPC Efficiency	0.00	0.01
$\Delta\eta_{HPC}$	Delta HPC Efficiency	0.00	0.01
$\Delta\eta_{HPT}$	Delta HPT Efficiency	0.00	0.01
$\Delta\eta_{LPT}$	Delta LPT Efficiency	0.00	0.01

Table 22: Core Defining Design Engine - Input Variables for Component Weight Factors

Variable	Description	Lower Limit	Upper Limit
sWt_{FanBld}	Fan Blade Weight Factor	0.90	1.00
sWt_{FanSt}	Fan Stator Weight Factor	0.90	1.00
sWt_{LPCBld}	LPC Blade Weight Factor	0.90	1.00
sWt_{LPCSt}	LPC Stator Weight Factor	0.90	1.00
sWt_{HPTBld}	Fore HPC Blade Weight Factor	0.90	1.00
sWt_{HPCSt}	Fore HPC Stator Weight Factor	0.90	1.00
$sWt_{HPCBld2}$	Aft HPC Blade Weight Factor	0.90	1.00
sWt_{HPCSt2}	Aft HPC Stator Weight Factor	0.90	1.00
sWt_{HPTBld}	HPT Blade Weight Factor	0.90	1.00
sWt_{HPTSt}	HPT Stator Weight Factor	0.90	1.00
sWt_{LPTBld}	LPT Blade Weight Factor	0.90	1.00
sWt_{LPTSt}	LPT Stator Weight Factor	0.90	1.00
sWt_{NozPri}	Core Nozzle Weight Factor	0.90	1.00
sWt_{NozSec}	Bypass Nozzle Weight Factor	0.90	1.00
sWt_{ShH}	HP Shaft Weight Factor	0.90	1.00
sWt_{ShL}	LP Shaft Weight Factor	0.90	1.00

Table 23: Core Defining Design Engine - Remaining Input Variables with Possible Uncertainty

Variable	Description	Lower Limit	Upper Limit
$T_{41,max}$ (degR)	Max HPT Inlet Temperature	3,186	3,300
$NChg_{LPT}$ (pct W_{25})	Bleed Fraction for LPT Non-chargeable Cooling	0.00	0.06
$SCustBld$	Customer Bleed Factor	0.5	2.0
$SHPX$	Customer HP Extraction Factor	0.5	2.0
$CmpL_{PRmax}$	Max First Stage PR, LPC	1.28	1.54
$Load_{HPT}$	HPT GE Loading	0.34	0.41
$Load_{LPT}$	LPT GE Loading	0.25	0.30
S_{eRam}	Inlet Recovery Factor	0.99	1.00

Table 24: Common Core Variant Engine - Design Variable Ranges

Variable	Description	Lower Limit	Upper Limit
π_{Fan} at ADP	Design Fan Pressure Ratio	1.3	1.7
$\pi_{Overall}$ at ADP	Design Overall Pressure Ratio	30.0	55.0
$T_{4,max}$ (degR)	Maximum Combustor Exit Temperature	3,300	3,750
ER at ADP	Design Extraction Ratio	0.9	1.25
SW_{CHPC}	HPC Inlet Flow Scale Factor	1.0	1.2
$NcPct_{HPT}$ at TKO	HPT Map Corrected Speed	100.	105.

Similar to the core defining design engine input variables, a total of 35 input variables correspond to the common core variant design within the same physics based model. The variables are also assigned ranges that encompass all possible values of interest for the experiments to come, with an additional buffer of one percent of the desired range added to the minimum and maximum values needed. The desired ranges, before adding buffers, are found in Table 24 for the variant design variables. The noise variables and their desired ranges are found in Table 25, Table 26, and Table 27.

Before moving on to the construction of the DoE that will be used to sample the input variable design space, the designer should perform final checks that the input variables of

Table 25: Common Core Variant Engine - Input Variables for Component Efficiency Impacts

Variable	Description	Lower Limit	Upper Limit
$\Delta\eta_{Fan}$	Delta Fan Efficiency	0.00	0.01
$\Delta\eta_{LPC}$	Delta LPC Efficiency	0.00	0.01
$\Delta\eta_{HPC}$	Delta HPC Poly Efficiency	0.00	0.01
$\Delta\eta_{HPT}$	Delta HPT Efficiency	0.00	0.01
$\Delta\eta_{LPT}$	Delta LPT Efficiency	0.00	0.01

Table 26: Common Core Variant Engine - Input Variables for Component Weight Factors

Variable	Description	Lower Limit	Upper Limit
sWt_{FanBld}	Fan Blade Weight Factor	0.90	1.00
sWt_{FanSt}	Fan Stator Weight Factor	0.90	1.00
sWt_{LPCBld}	LPC Blade Weight Factor	0.90	1.00
sWt_{LPCSt}	LPC Stator Weight Factor	0.90	1.00
sWt_{HPCBld}	Fore HPC Blade Weight Factor	0.90	1.00
sWt_{HPCSt}	Fore HPC Stator Weight Factor	0.90	1.00
$sWt_{HPCBld2}$	Aft HPC Blade Weight Factor	0.90	1.00
sWt_{HPCSt2}	Aft HPC Stator Weight Factor	0.90	1.00
sWt_{HPTBld}	HPT Blade Weight Factor	0.90	1.00
sWt_{HPTSt}	HPT Stator Weight Factor	0.90	1.00
sWt_{LPTBld}	LPT Blade Weight Factor	0.90	1.00
sWt_{LPTSt}	LPT Stator Weight Factor	0.90	1.00
sWt_{NozPri}	Core Nozzle Weight Factor	0.90	1.00
sWt_{NozSec}	Bypass Nozzle Weight Factor	0.90	1.00
sWt_{ShH}	HP Shaft Weight Factor	0.90	1.00
sWt_{ShL}	LP Shaft Weight Factor	0.90	1.00

Table 27: Common Core Variant Engine - Remaining Input Variables with Possible Uncertainty

Variable	Description	Lower Limit	Upper Limit
$T_{41,max}$ (degR)	Max HPT Inlet Temperature	3,186	3,300
$NChg_{LPT}$ (pct W_{25})	Bleed Fraction for LPT Non-chargeable Cooling	0.00	0.06
$SCustBld$	Customer Bleed Factor	0.5	2.0
$SHPX$	Customer HP Extraction Factor	0.5	2.0
$CmpL_{PRmax}$	Max First Stage PR, LPC	1.28	1.54
$Load_{HPT}$	HPT GE Loading	0.34	0.41
$Load_{LPT}$	LPT GE Loading	0.25	0.30
S_{eRam}	Inlet Recovery Factor	0.99	1.00

interest are included in the superset of variables, and that the assigned ranges are large enough to account for all possible design exercises of interest. While the resultant surrogate models will provide accurate metric value estimates for designs with input variable values within the ranges used for training, significant error can be introduced when even a minor extrapolation takes place.

4.1.2 Step 2: Deterministic assessment and regression

Once the designer is sure that the physics based model is prepared and the input variable ranges to sample are adequate, the surrogate training DoE can be constructed. The design-variant engine exploration requires sampling within a total of 71 input variable dimensions: 36 core defining design engine variables and 35 common core variant engine variables. Since each simulation consisting of a design engine and subsequent common core variant simulation has an average runtime of less than 5 seconds, a large training set was produced, providing ample design samples for training and testing of the resultant surrogate models. For surrogate model training, a 3,600 case latin hypercube design was constructed. An additional 900 random cases were constructed for testing of the surrogate regressions.

Final checks were performed ensuring the physics based model provided the input and resultant output metrics of interest, and the DoE simulations were launched. The 4,500

case batch of simulations were carried out in less than 3 hours with four instances of the simulation environment running on a quad-core machine. After the completion of all simulations, it was verified that less than 10% of the simulations failed, and that the failures were randomly distributed within the sampled input variable space.

With the resultant input/output data of all the training simulations of the physics based model, the resultant surrogate models can be regressed. For each output metric of interest, an Artificial Neural Network (ANN) surrogate model was produced. For each model, the ANN architectures were explored. Regressions were constructed using the Neural Network Matlab toolbox, allowing ANN models to have between 5 and 15 hidden nodes. Each network architecture was allowed 60 seconds to achieve the best possible fit quality. The resultant network architecture with the best fit quality was selected for use to represent that particular output metric.

Surrogate models trained to estimate the core defining design engine metrics are functions of only the 36 input variables of the design engine. These surrogate models are of the form shown in Equation 46, and from now on will be referred to as *Surrogate Set A* or $SSA_{Des,Det}$.

$$\mathbf{SSA}_{Des,Det} = f(\mathbf{CV}_{Des}, \mathbf{NV}_{Des}) \quad (46)$$

Surrogate models trained to estimate the common core variant engine design are functions of all 71 input variables, showing the variant design dependency on the core defining engine whose resultant engine core definition is applied to the variant application. These surrogate models trained to predict common core variant performance are of the form shown in Equation 47, and from now on will be referred to as *Surrogate Set B* or $SSB_{Var,Det}$.

$$\mathbf{SSB}_{Var,Det} = f(\mathbf{CV}_{Des}, \mathbf{NV}_{Des}, \mathbf{CV}_{Var}, \mathbf{NV}_{Var}) \quad (47)$$

4.1.2.1 *Surrogate model evaluation metrics*

As was discussed in Chapter 3, in order to ensure adequate model fit quality, the following surrogate model fit requirements must be met:

- Model Fit Error (MFE), the distribution of surrogate model error that is determined

through the comparison of predicted performance to the actual performance levels used in the training of the surrogate model

- Shape of MFE distribution should resemble a normal distribution
 - Mean value of MFE distribution should be close to zero
 - MFE standard deviation (STD) should be less than one
- Model Representation Error (MRE), the distribution of surrogate model error that is determined through the comparison of predicted performance to the actual performance levels of the reserve data not used in the training of the surrogate model. MRE quality determined based on the same metrics used in the evaluation of the MFE distribution.
 - The quotient of MRE and MFE standard deviations should have a value less than two.
 - A high coefficient of determination (R^2) must be achieved, having a value greater than 0.99. A value close to one indicates the model is able to explain the vast majority of variation in the metric response of interest. The model's R^2 can be determined by referring back to Equation 16 provided in Chapter 3, which is repeated below.
 - When visualizing the actual-by-predicted plot, the data should not contain tails where actual-by-predicted values depart from the perfect fit line at low and high values of response data.
 - The residual error should be at least two orders of magnitude less than the actual response levels.

$$R^2 = 1 - \frac{SS_{error}}{SS_{total}} \quad (16a \text{ Revisited})$$

$$SS = \sum (Y - \bar{Y})^2 \quad (16b \text{ Revisited})$$

$$\bar{Y} = \frac{\sum Y}{N} \quad (16c \text{ Revisited})$$

Examples of the surrogate model fit qualities achieved are shown in Figure 47 and Figure 48 for the core defining design and common core variant engine, respectively. The

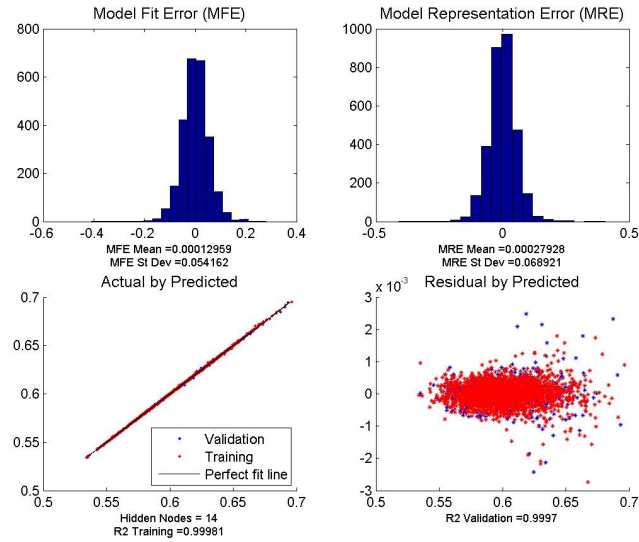


Figure 47: Fit quality achieved for deterministic surrogate model predicting TSFC at ADP for the core defining design engine.

remainder of surrogate fit quality statistics for the deterministic ANN surrogate models can be found in Appendix C.

4.1.2.2 Kernel based support vector machine generation

The deterministic ANN surrogate models produced from the deterministic assessment data proved to represent the training data and the test data very well. However, a small percentage of DoE design-variant samples failed to converge during the common core variant simulation within the NPSS model. These samples failed due to the combination of design and variant input variables having values whose initially sized core definition was not able to provide the core power necessary to achieve a converged common core variant cycle having the inputted characteristics.

This class of design-variant simulation failures necessitated the construction of a kernel based support vector machine classification model. Without the classification model, non-realistic common core variant performance estimates would unknowingly be made. The deterministic surrogate models lacked training data within the multidimensional region of the input variable space where common core power was not available for the variant design

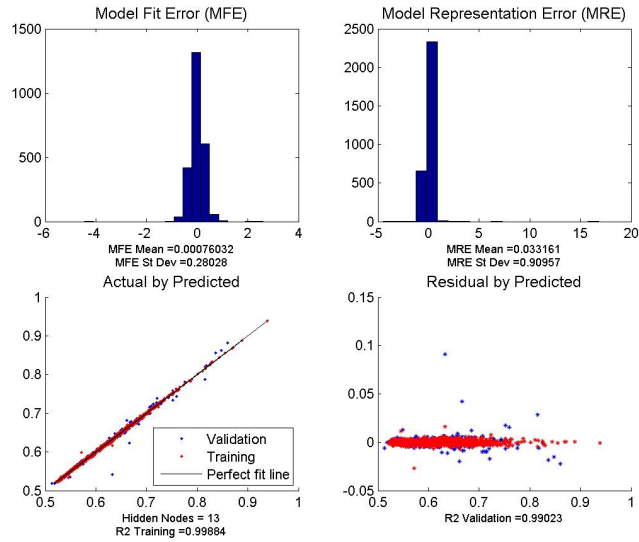


Figure 48: Fit quality achieved for deterministic surrogate model predicting minimum cruise TSFC for the common core variant engine.

under consideration.

In order to produce the kernel based SVM model, the design-variant simulation data was used, where each case was assigned a classification of whether or not the variant engine was able to achieve inputted characteristics with the common core definition. A resultant SVM model was trained within Matlab to achieve the resultant classification model.

An additional iteration of SVM model training was performed to ensure ample classification performance. Data points that were unable to achieve variant engine convergence were identified, and their closest neighboring data samples that were able to achieve variant convergence were also identified. Based on the input variable settings of the neighboring pairs of data points on either side of the boundary separating the two classes, an additional data point was produced to lie halfway in between the points within the input variable space.

Once these additional points were generated and then simulated within the physics based modeling and simulation environment, the training of the kernel based SVM model was again performed. The resultant SVM model was assumed to offer ample classification

accuracy, having the ability to accurately classify over 90% of the input-output data produced. If additional accuracy were desired, a similar technique could be used to more richly sample the suspected region where the boundary between the two classes of data was likely to exist. However, for the present work and corresponding probabilistic assessments favor designs that lie away from the suspected boundary.

As will be shown in the following section, designs whose variability in the input space due to uncertainty push portions of its probabilistic spread into this unfavorable region where ample variant core power is likely not available have accompanying degradation in its confidence interval performance levels. Such a design, when collecting confidence interval metric levels, have the confidence levels for a given performance levels scaled by the fraction of designs that were suspected to have ample variant core power. For example, if only 90% of Monte Carlo simulations of a candidate design have ample variant core power, then all confidence intervals of variable performance are scaled by 90%. The original 95% confidence interval performance estimate becomes the 85.5% confidence interval performance level. Since favorable high confidence interval performance levels are desired within the present work, these designs whose variability in performance cross the boundary where the common core variant application is likely to lack ample core power will be extremely unfavorable.

4.1.3 Step 3: Probabilistic assessment and regression

Now that two sets of deterministic surrogate models exist, representing both the core defining design engine and the common core variant engine, a probabilistic assessment can be performed. The goal of this assessment is again to provide input/output representative data, but this time to produce confidence interval metric estimates for designs under all possible uncertainty scenarios of interest.

For this assessment, a DoE of only control variables is constructed. The set of control variables consists of design variables and uncertainty distribution shape parameters for both the core defining design and common core variant engine. For the present work, the set of control variables consisted of eight core defining design variables, seven variant design variables, and alpha and beta shape parameters of the beta distributions representing the

uncertainty assumed present in the experiments to come. A total of 48 uncertainty shape parameters were selected to vary, 24 being applied to both the core defining design and common core variant engine. A latin hypercube DoE was constructed for the 63 input variable space. Each of the control variable ranges were consistent with the originally sampled ranges of the deterministic assessment. All alpha and beta uncertainty shape parameters were assigned ranges between 1 and 10, providing great flexibility in the distribution shapes that can be represented. The minimum and maximum uncertainty distribution impacts remained fixed at the originally sampled ranges of the noise variables during the deterministic assessment.

The resultant DoE of control variable settings was then launched. For each case of the DoE, 50,000 Monte Carlo simulations were performed, having the noise variable settings varied to represent the DoE case's inputted uncertainty scenario's distributions. Because a large number of uncertainty distributions are assumed present and the cost of each individual Monte Carlo simulation is very cheap, the large number of simulations was used to ensure accurate representation of all uncertainty distributions present. For each Monte Carlo simulation, all deterministic surrogate models, including the SVM model, were evaluated. For each metric, the 50,000 metric estimates were then sorted, and 13 confidence interval performance levels were calculated for the DoE case's control variable settings. After performing Monte Carlo analyses for each DoE case, the resultant data set was made up of control variable settings and 13 confidence interval performance levels for each of the metrics of interest. The quoted confidence levels were then scaled by the fraction of Monte Carlo simulations that were classified by the SVM to have ample variant core power.

The resultant input/output data was then used in the regression of probabilistic surrogate models. For each output metric of interest, ANN architectures were again explored, varying the number of hidden nodes between 5 and 15. The resultant probabilistic surrogate model architecture that achieved the best fit within the allotted training time was then saved.

Probabilistic surrogate models trained to estimate the core defining design engine metrics are functions of the 32 control variables corresponding to the core defining design engine

and one additional input variable for the desired confidence of the performance estimate. These surrogate models are of the form shown in Equation 48, and from now on will be referred to as *Surrogate Set C* or $SSC_{Des,Prob}$.

$$\mathbf{SSC}_{Des,Prob} = f(CI, \mathbf{CV}_{Des}, \mathbf{A}_{Des}, \mathbf{B}_{Des}) \quad (48)$$

Probabilistic surrogate models trained to estimate the common core variant engine design are functions of all 63 control variables and one additional input variable for the desired confidence level of metric estimate. This is again required because of the variant design dependency on the core defining engine whose core definition is applied to the variant application. These surrogate models trained to predict likely common core variant performance are of the form shown in Equation 47, and from now on will be referred to as *Surrogate Set D* or $SSD_{Var,Prob}$.

$$\mathbf{SSD}_{Var,Prob} = f(CI, \mathbf{CV}_{Des}, \mathbf{A}_{Des}, \mathbf{B}_{Des}, \mathbf{CV}_{Var}, \mathbf{A}_{Var}, \mathbf{B}_{Var}) \quad (49)$$

Just as was the case when evaluating the deterministic surrogate models produced, the same fit quality was required of the probabilistic surrogate models. Figure 49 and Figure 50 contain example probabilistic surrogate model fit statistics achieved for the core defining design engine and common core variant engine metrics, respectively. The fit statistics of the remainder of probabilistic models produced can be found in Appendix D.

4.1.4 Step 4: Robust design exploration and selection

At this point in the ERDS process, the designer is now able to perform many robust design explorations in a highly efficient manner with the resultant sets of surrogate models he now has at his disposal. After carrying out the above process, four sets of surrogate models have been produced. A set of deterministic surrogates exist for both the core defining design engine and the common core variant engine. Also, a set of probabilistic surrogate models also exists for both the core defining design engine and common core variant engine.

The actual Matlab code produced to perform the above ERDS probabilistic assessment can be found in Appendix B. By following the ERDS process, four sets of surrogate models were produced with wide enough input variable ranges that their capabilities can be utilized

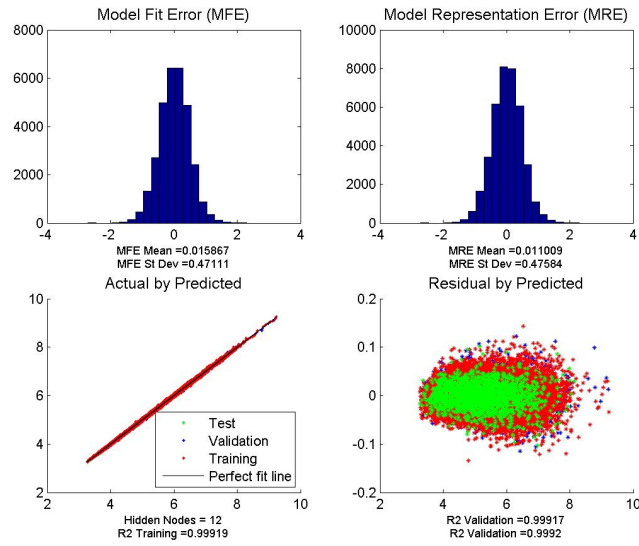


Figure 49: Fit quality achieved for probabilistic surrogate model predicting HPC exit corrected flow at ADP for the core defining design engine.

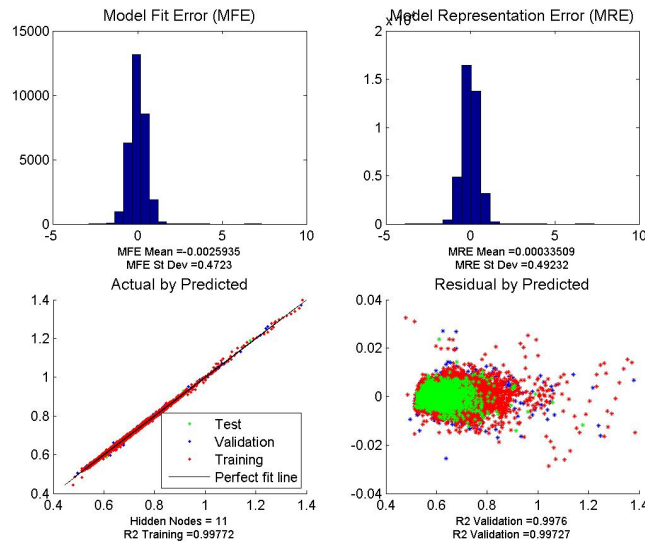


Figure 50: Fit quality achieved for probabilistic surrogate model predicting minimum cruise TSFC for the common core variant engine.

for all four of the experiments to follow. This again highlights the key contribution of ERDS method development that enables the fully integrated COMMENCE method to have such a powerful range of capabilities. If ample considerations are made at the onset of the process, the resultant surrogate representations can be used to perform a wide range of useful studies aiming to provide for more informed common core design decisions. This range in capabilities will be demonstrated by the experiments to follow.

4.2 *Experiment 1: Enhanced Robust Design Simulation Method Applied to Single Application Turbofan Design Problem*

- **Research Question 1:** *How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?*

Hypothesis 1: *Probabilistic performance levels of candidate cycle designs should be estimated with the use of surrogate models that predict likely performance under various inputted uncertainty scenarios for any desired confidence interval.*

4.2.1 Overview

This experiment serves to determine the feasibility of the Enhanced Robust Design Simulation (ERDS) method. It will show the advantages of the ERDS method when making design selections when compared to a traditional deterministic approach. A single application turbofan design and technology scenario will be considered when making design selections. A set of likely requirements and technology impacts will be established, and designs will be selected using two competing approaches.

The first design approach is the traditional deterministic approach. Deterministic surrogate models developed in the previous section, specifically the ones representing the core defining design engine, will be used to explore and evaluate candidate designs in the absence of uncertainty. Design selections will be made that maximize performance under the fixed set of design requirements and realized technology impacts.

The second design approach applies the ERDS method. Uncertainty distributions will be applied to installation requirements and realized technology impacts. Candidate designs will be evaluated under these assumed sources of uncertainty with the goal being to select the candidate design that has a high probability of satisfying all requirements and achieving competitive performance. Design selections will be made that maximize the *likely* confidence interval performance under uncertainty.

This experiment will establish and detail the ERDS process, comparing it to a more

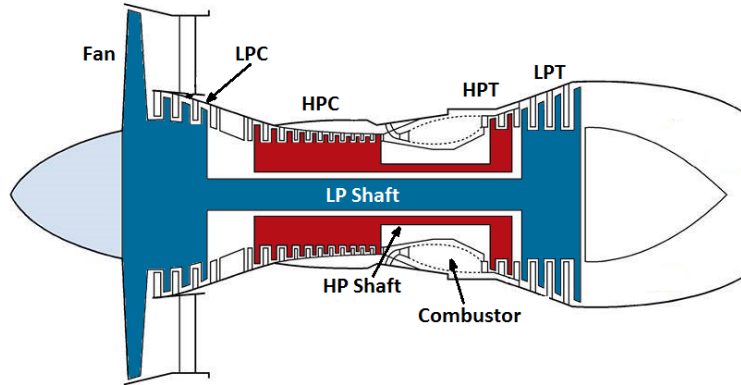


Figure 51: High bypass, two shaft boosted turbofan engine architecture.

conventional deterministic method. The experiment will conclude by comparing design selections made with the two competing methods. Various visualization techniques will be demonstrated when performing design space explorations. As the complexity of the design exploration increases, limitations of the initial visualization techniques will be identified. Alternative techniques will be introduced to better aid the designer in understanding the design space trends and corresponding trades being made for various design selections.

4.2.2 Requirements and Technology Scenario

This experiment aims to make parameter setting selections for the design of a high bypass direct drive turbofan engine, depicted in Figure 51. There are three design points considered in the NPSS engine model when designing and sizing the engine under consideration. The design points listed in Table 28 are uniformly used for all engine designs considered in this chapter. For this experiment, the candidate engine should be selected to ensure all requirements listed in Table 29 are satisfied. Additionally, pareto optimal performance should be achieved, offering a non-dominated solution that compromises between the engine pod weight and the achievable TSFC at Aerodynamic Design Point (ADP) operating conditions.

The baseline definition for this study aims to represent a middle-of-market turbofan engine such as the CFM56-7B27, sized for a large single aisle aircraft such as the 737-800.

Table 28: Experiment 1 - Engine Design Points

Design Point	Designation	Altitude, <i>ft</i>	Mach	dTs, $^{\circ}F$
Aerodynamic Design Point	ADP	35,000	0.80	0.0
Top of Climb	TOC	35,000	0.85	0.0
Hot Day Takeoff	TKO	0	0.25	27.0

Table 29: Experiment 1 - Engine Design Requirements

Requirement	Value	Units
Net Thrust at TKO	22,780	<i>lbf</i>
Net Thrust at TOC	5,960	<i>lbf</i>
Constant Customer Bleed	2.35	<i>lbm/s</i>
Constant Horsepower Extraction	250.	<i>shp</i>
TSFC at ADP	minimize	<i>lbm/lbf/hr</i>
Engine Pod Weight	minimize	<i>lbm</i>

The cycle parameter settings and corresponding performance of the baseline engine are tabulated in Table 30. For the present scenario of requirements and technology impacts, candidate design performance will be compared to the baseline. The designer is to make design parameter setting selections that provide for the best possible performance improvement. The design parameters allowed ranges can be found in Table 31. For this design study, technology improvement with respect to the baseline engine definition is assumed likely. The most likely turbomachinery efficiency and weight improvements are listed in Table 32.

To prevent the selection of candidate cycles that would exceed realistic temperature levels, three constraints are imposed. The maximum temperature limits listed in Table 33 are imposed on the HPC exit, the HPT inlet, and the LPT inlet temperature levels. The maximum allowed HPC exit temperature ($T_{3,max}$) corresponds to the HPC material limit, allowing for an uncooled HPC. The LPT inlet temperature limit ($T_{45,max}$) is imposed on the first LPT stage after introduction of nonchargeable cooling flow, which is fixed at a constant fraction of core inlet airflow (W_{25}). These first two temperature constraints are imposed after producing the design space exploration. Candidate designs will be filtered,

Table 30: Experiment 1 - Baseline Cycle, Technology Level, and Corresponding Performance

Category	Variable	Baseline Value
Cycle	π_{Fan} at ADP	1.685
	π_{HPC} at ADP	9.369
	$\pi_{Overall}$ at ADP	30.094
	$T_{4,max}$ ($degR$)	3,300 $degR$
	ER at ADP	1.018
Tech Level	η_{Fan} at ADP	0.895
	η_{LPC} at ADP	0.924
	η_{HPC} at ADP	0.871
	η_{HPT} at ADP	0.889
	η_{LPT} at ADP	0.900
Performance	TSFC at ADP	0.6640 $lbm/lbf/hr$
	Pod Weight	5,652. lbm

Table 31: Experiment 1 - Design Variable Ranges

Variable	Description	Lower Limit	Upper Limit
π_{Fan} at ADP	Design Fan Pressure Ratio	1.3	1.7
π_{HPC} at ADP	Design HPC Pressure Ratio	9.0	16.0
$\pi_{Overall}$ at ADP	Design Overall Pressure Ratio	30.0	55.0
$T_{4,max}$ ($degR$)	Maximum Combustor Exit Temperature	3,300	3,750
ER at ADP	Design Extraction Ratio	0.9	1.25

eliminating designs that violate these temperature limits. Conversely, the maximum HPT inlet temperature ($T_{41,max}$) limit is used to size the HPT nonchargeable cooling flow during the design simulation of the engine.

4.2.3 Experiment 1a: Deterministic Design Selection

This first sub-experiment will evaluate candidate designs with fixed technology impacts, assuming 100% certainty in the values that will be realized in the engine design. This traditional deterministic approach will be used first to understand the cycle design space of the present design problem. The method will be used to select a candidate design that meets requirements and performs best under the assumed technology impacts. The resultant

Table 32: Experiment 1 - Most likely engine technology improvements with respect to baseline engine.

Category	Variable	Likely Improvement
Turbomachinery Component Efficiency Improvements	$\Delta\eta_{Fan}$	+0.25%
	$\Delta\eta_{LPC}$	+0.25%
	$\Delta\eta_{HPC}$	+0.25%
	$\Delta\eta_{HPT}$	+0.25%
	$\Delta\eta_{LPT}$	+0.25%
Turbomachinery Blade/Vane Weight Reductions	sWt_{Fan}	-5.00%
	sWt_{LPC}	-5.00%
	sWt_{HPC}	-5.00%
	sWt_{HPT}	-5.00%
	sWt_{LPT}	-5.00%

Table 33: Experiment 1 - Temperature limits imposed on candidate designs.

Variable	Description	Maximum Value
$T_{3,max}$	Maximum HPC Exit Temperature	1,700 <i>degR</i>
$T_{45,max}$	Maximum LPT Inlet Temperature	2,300 <i>degR</i>
$T_{41,max}$	Maximum HPT Inlet Temperature	3,186 <i>degR</i>

design selection will then be compared to robust design selections made using the ERDS method.

4.2.3.1 Exploration of the Deterministic Engine Design Space

For a standalone design study, the ERDS process would be carried out to sample the physics based model to produce deterministic representations of the single application design space. Fortunately, the deterministic surrogate models of *Surrogate Set A* produced in Section 4.1.2 already offer accurate representations of the physics based model and design space considered for this experiment. Therefore, existing surrogate models can be used and the deterministic assessment can now be performed.

Using the deterministic representations of the physics based engine model, the engine design space was explored. Ten thousand unique design samples were randomly placed within the design space whose boundaries are defined by the variable ranges in Table 31.

For each set of design and fixed noise variable settings, the closed form equations of *Surrogate Set A* were evaluated, providing performance estimates for all 10,000 unique designs in a matter of seconds.

Visualization techniques can now be used to construct charts to show the trends and relationships present throughout the design space. The multivariate scatterplot matrix in Figure 52 displays all ten thousand designs considered. Each point in a given scatterplot represents one of the 10,000 unique design samples. The placement of each design in a given scatterplot is determined by its input or estimated metric values of the plot axes. Designs having temperature estimates that violate at least one of the imposed temperature limits are filtered out. The gray points in Figure 52 are those designs that violated at least one of the two temperature constraints. The black points represent the remaining feasible designs after filtering. For example, all high OPR designs are gray, indicating that they violated the $T_{3,max}$ limit at takeoff.

Each scatterplot shows a two-dimensional plane of the design space, allowing the designer to visualize the relationship between any two dimensions. For example, the designer can see in the bottom two rows of the multivariate plot the effects that the design variable settings have on TSFC and pod weight. Based on the plots, the design fan pressure ratio (FPR) is the key driver for both the TSFC and the pod weight. Low values of FPR allow for desirable TSFC levels by driving the designs to higher bypass ratios. However, lower FPR and high bypass ratio designs, drive the engine weight to higher levels due to the increased fan and nacelle diameters of higher bypass ratio turbofan engines.

The second row of the multivariate plot shows that the key drivers in the HPC exit and LPT inlet temperatures are the design values of the overall pressure ratio (OPR) and the design pressure ratio of the HPC (HPCPR). High OPR designs have high HPC exit temperatures accompanying the high pressure. Also shown in the second row of the multivariate plot is a relationship between OPR and HPCPR that causes the LPT inlet temperature at takeoff to go above the allowed level. This is due to the fact that low HPCPR designs require less work by the HPT to drive the HPC, allowing for excess thermal energy to leave the HPT and enter the LPT. This causes elevated levels of LPT inlet temperature,

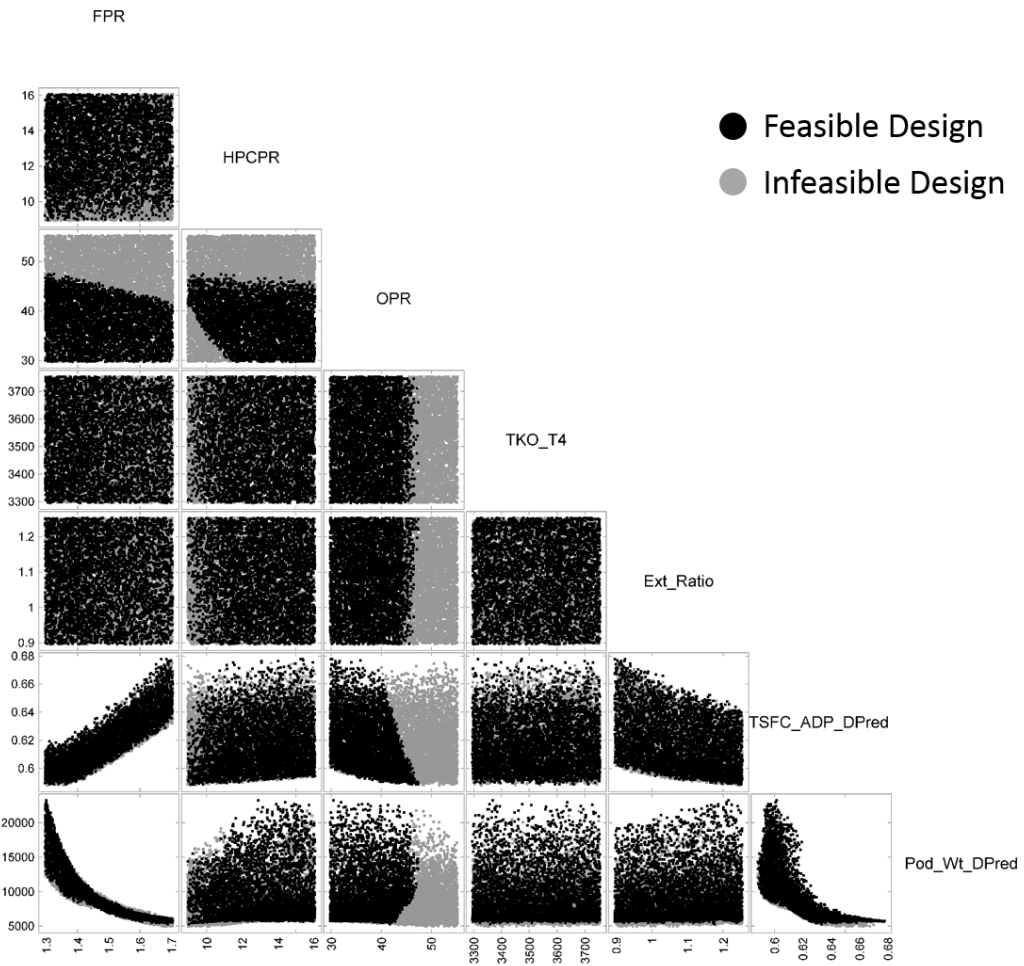


Figure 52: Experiment 1a - Constrained scatterplot of deterministic simulation data.

violating the constraints imposed on the design.

The bottom right cell of the multivariate plot shows the relationship of the TSFC at ADP to the engine pod weight, defining the pareto frontier on which candidate designs will be selected. Figure 53 shows the same multivariate plot as shown in Figure 52, but each feasible design is now colored based on its corresponding TOPSIS score. The TOPSIS scores were determined based on equal preference being given to low TSFC and low engine weight. This allows for the designer to see where in each design parameter setting range the top scores can be achieved. The plot shows that intermediate design FPR values are desirable that allow for compromised levels of both TSFC and pod weight.

While multivariate scatterplot matrices display candidate designs within various design planes of interest, parallel coordinate plots can help identify relationships on multiple design dimensions. The parallel coordinates chart in Figure 54 contains threads for all 10,000 designs explored. Each thread of connected line segments represents a unique design. The designs in black are both feasible and achieved a high score in the TOPSIS evaluation. The gray designs were either screened out or scored low in the TOPSIS evaluation. By plotting both sets of designs, the designer can again see where in the sampled space the candidate designs are feasible and achieve the best desired performance levels. This chart shows that low to intermediate FPR values within the sampled region score the highest while remaining feasible. Also apparent is that low HPCPR and mid-OPR designs score best while remaining feasible. Based on the deterministic assessment, the most desirable cycle settings can be easily identified with this chart.

4.2.3.2 Candidate Design Selection from Deterministic Assessment

After richly sampling the deterministic engine cycle design space and evaluating each candidate design using the surrogate models previously trained, a candidate design may be selected. At this point, further design considerations can be made. A typical additional design consideration made is the fan diameter of the candidate design. For simplicity, this consideration is not considered in the present deterministic design problem, only the temperature constraints imposed have been considered.

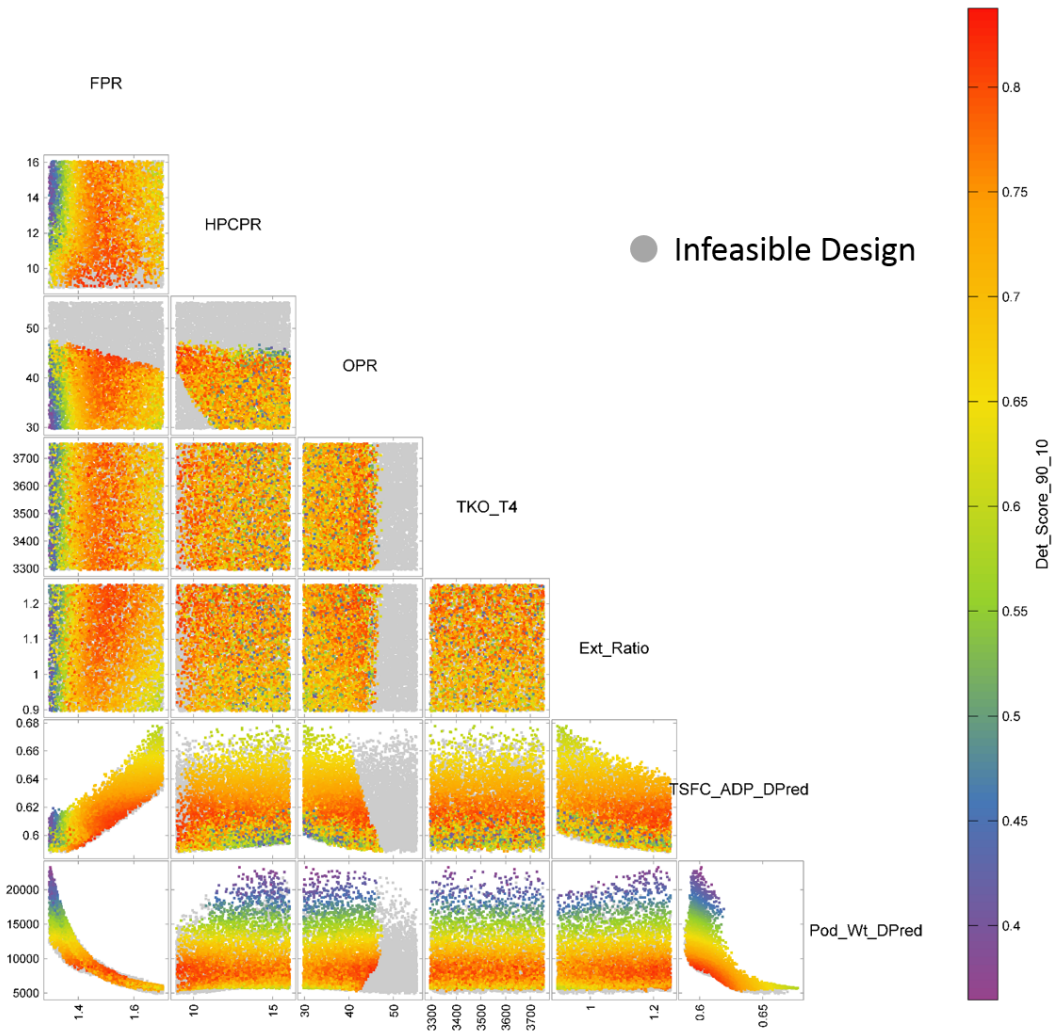


Figure 53: Experiment 1a - Constrained scatterplot of deterministic simulation data, colored by TOPSIS score.

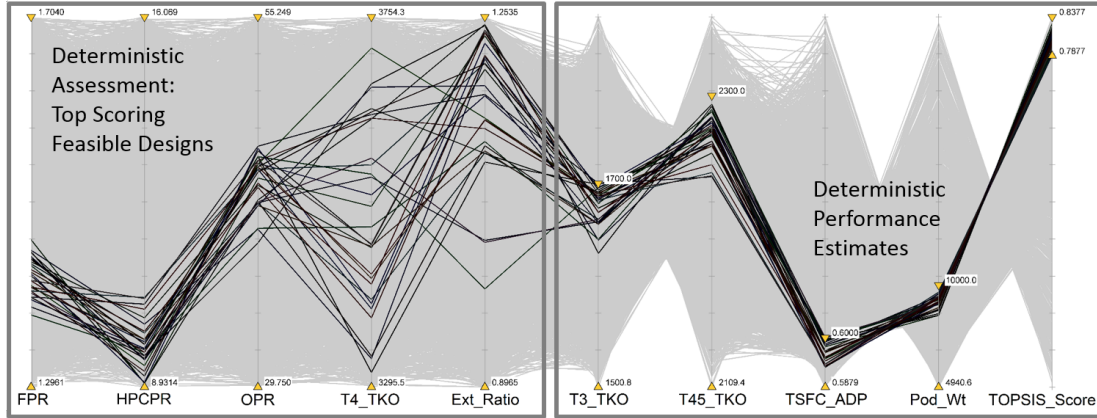


Figure 54: Experiment 1a - Constrained parallel coordinate chart of deterministic assessment with highest TOPSIS scores.

Of the feasible design candidates, the design with the highest TOPSIS score can be selected. The top feasible design parameter settings and its corresponding performance levels have been listed in Table 34. Note that both the $T_{3,max}$ and $T_{45,max}$ levels lie very close to the constraint levels. The candidate selection performs significantly better in terms of TSFC than the original baseline cycle (0.5978 compared to a baseline level of 0.6640), but the pod weight is significantly higher (8,500 compared to a baseline level of 5,652 lbm) due to the lack of any fan diameter constraint imposed on the design exploration. The candidate design was driven to a higher bypass ratio in order to improve the $TSFC$ achievable by the cycle. Also, consideration into whether engine weight or engine efficiency (i.e. TSFC) is the significant driver in the resultant engine fuel burn that the aircraft designer is really concerned about.

The candidate cycle selected from the deterministic assessment will be used as a benchmark to compare designs selected in the following robust design assessment. Keep in mind that the technology variable settings were assumed fixed in the present study, and no margin was introduced that pushed the temperature levels away from the constraints imposed. By considering uncertainty present in this design problem, candidate selections will have a high confidence associated with meeting the temperature constraints, giving the designer confidence that the resultant design will not surpass the temperature limits of the materials

Table 34: Experiment 1a - Candidate design selection with top TOPSIS score that maintained feasible temperature levels within imposed limits.

Parameter	Candidate Setting
π_{Fan}	1.458
π_{HPC}	9.004
$\pi_{Overall}$	44.15
$T_{4,max}$ ($^{\circ}R$)	3,520.
ER	1.235
$T_{3,max}$ ($^{\circ}R$)	1,680.
$T_{45,max}$ ($^{\circ}R$)	2,290.
TSFC at ADP ($lbm/lbf/hr$)	0.5978
Pod Weight (lbm)	8,500.

used in the engine design.

4.2.4 Experiment 1b: Robust Design Probabilistic Assessment and Candidate Design Selection

This experiment is the first to take advantage of the second portion of the ERDS process, the probabilistic RDS block of logic originally discussed in Chapter 3, shown in Figure 25, which has been displayed again for ease of access. The probabilistic models of *Surrogate Set C* that were trained in Section 4.1.3 are used here to predict the likely performance at some desired level of confidence with a set of assumed uncertainty distributions present in the design. Design selections can then be made that maximize the likely performance while ensuring high confidence that all imposed constraints are also met.

For the robust design exploration performed here, various sources of uncertainty are assumed present in the design. Table 35 contains the assumed uncertainty distributions for the turbomachinery component efficiency improvements. Note that the distributions have most likely impacts that corresponds to the deterministic impacts listed in Table 32. Also contained in Table 35 are the four parameter values defining the beta distributions shown: the minimum and maximum possible impacts and the alpha and beta shape parameters for the corresponding beta distributions. Similar to the efficiency improvement distributions, the distributions of likely component weight reductions considered in the present probabilistic assessment can be found in Table 36.

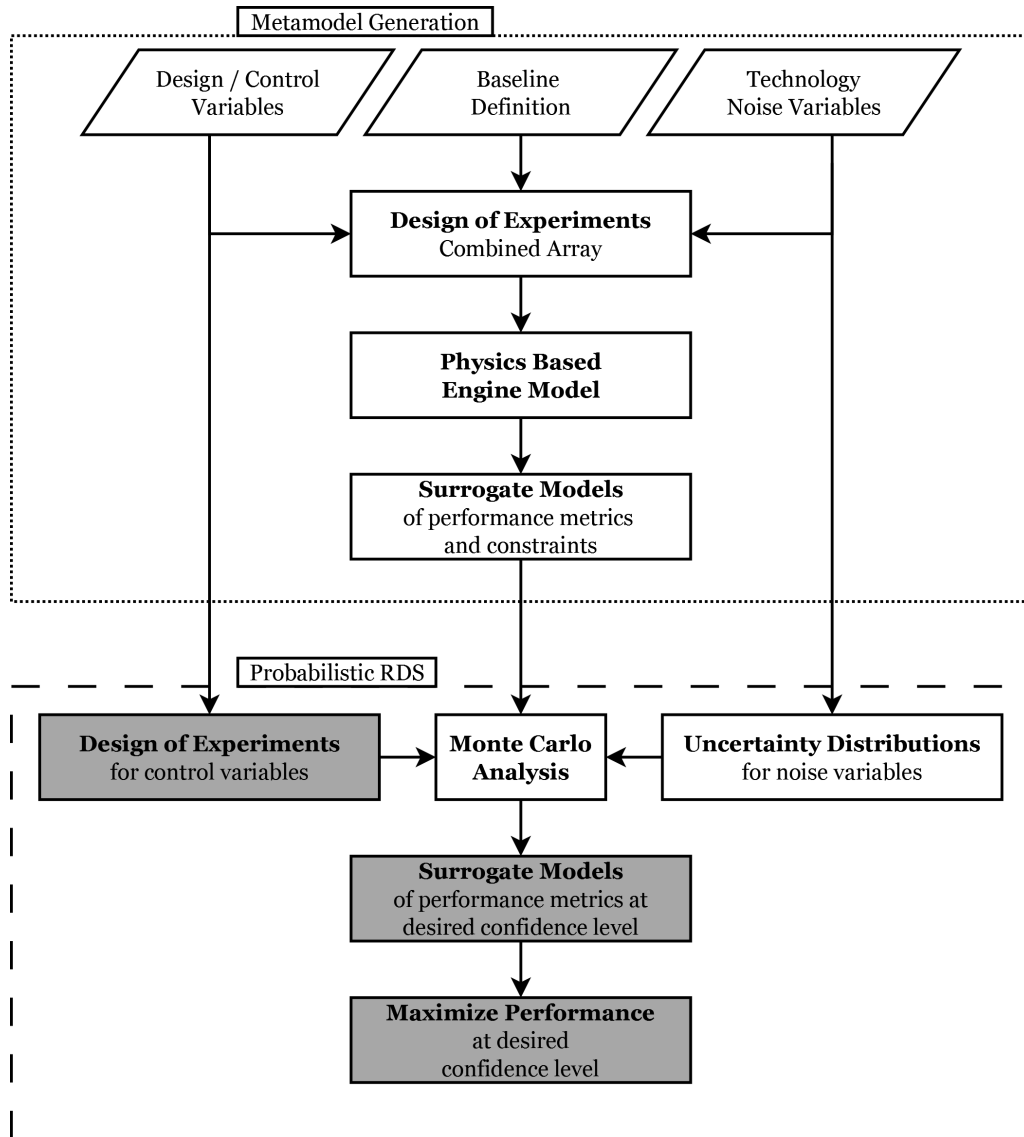


Figure 25 Revisited: Robust Design Simulation method modified for robust engine and technology design.

Table 35: Experiment 1 - Technology Impact Distributions - Component Efficiency

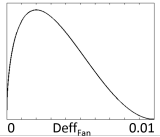
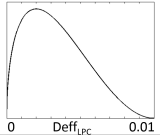
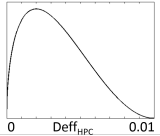
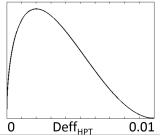
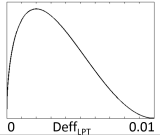
Impact	Min, Max, Alpha, Beta	PDF
$\Delta\eta_{Fan}$	0.00, 0.01, 1.5, 3.0	 <p>A plot of the probability density function for $\Delta\eta_{Fan}$. The x-axis is labeled 'Deff_{Fan}' and ranges from 0 to 0.01. The curve starts at (0,0), rises to a peak around 0.002, and then decays towards 0.01.</p>
$\Delta\eta_{LPC}$	0.00, 0.01, 1.5, 3.0	 <p>A plot of the probability density function for $\Delta\eta_{LPC}$. The x-axis is labeled 'Deff_{LPC}' and ranges from 0 to 0.01. The curve starts at (0,0), rises to a peak around 0.002, and then decays towards 0.01.</p>
$\Delta\eta_{HPC}$	0.00, 0.01, 1.5, 3.0	 <p>A plot of the probability density function for $\Delta\eta_{HPC}$. The x-axis is labeled 'Deff_{HPC}' and ranges from 0 to 0.01. The curve starts at (0,0), rises to a peak around 0.002, and then decays towards 0.01.</p>
$\Delta\eta_{HPT}$	0.00, 0.01, 1.5, 3.0	 <p>A plot of the probability density function for $\Delta\eta_{HPT}$. The x-axis is labeled 'Deff_{HPT}' and ranges from 0 to 0.01. The curve starts at (0,0), rises to a peak around 0.002, and then decays towards 0.01.</p>
$\Delta\eta_{LPT}$	0.00, 0.01, 1.5, 3.0	 <p>A plot of the probability density function for $\Delta\eta_{LPT}$. The x-axis is labeled 'Deff_{LPT}' and ranges from 0 to 0.01. The curve starts at (0,0), rises to a peak around 0.002, and then decays towards 0.01.</p>

Table 36: Experiment 1 - Technology Impact Distributions - Component Weight

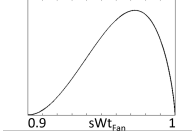
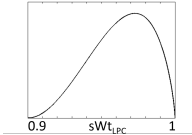
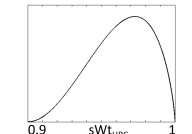
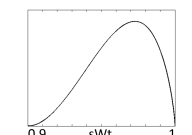
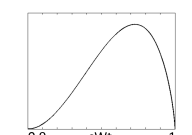
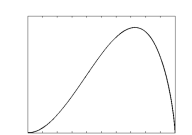
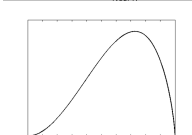
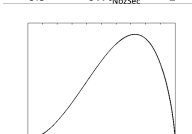
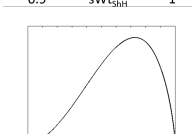
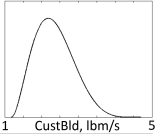
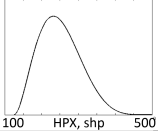
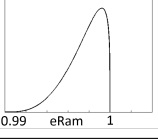
Impact	Min, Max, Alpha, Beta	PDF
sWt_{Fan}	0.90, 1.00, 3.00, 1.75	
sWt_{LPC}	0.90, 1.00, 3.00, 1.75	
sWt_{HPC}	0.90, 1.00, 3.00, 1.75	
sWt_{HPT}	0.90, 1.00, 3.00, 1.75	
sWt_{LPT}	0.90, 1.00, 3.00, 1.75	
sWt_{NozPri}	0.90, 1.00, 3.00, 1.75	
sWt_{NozSec}	0.90, 1.00, 3.00, 1.75	
sWt_{ShH}	0.90, 1.00, 3.00, 1.75	
sWt_{ShL}	0.90, 1.00, 3.00, 1.75	

Table 37: Experiment 1 - Technology Impact Distributions - Installations

Impact	Min, Max, Alpha, Beta	PDF
<i>CustBld</i>	1.175, 4.700, 3.00, 6.00	
<i>HPX</i>	125, 500, 3.00, 6.00	
<i>s_eRam</i>	0.99, 1.00, 4.00, 1.25	

For the deterministic assessment performed in Experiment 1a, many assumptions are made regarding the assumed installation requirements, the turbomachinery loading levels achievable, and engine cooling assumptions. In order to make robust selections in the present experiment that are insensitive to these assumptions, uncertainty distributions are also applied to these assumptions. The beta distribution parameter settings and corresponding distributions representing the uncertainty present in the installations, component loading, and engine cooling assumptions can be found in Table 37, Table 38, and Table 39, respectively.

4.2.4.1 Probabilistic Exploration of the Engine Design Space

With the set of probabilistic models trained in Section 4.1.3, referred to as *Surrogate Set C*, representing the probabilistic performance of a design engine under uncertainty, exploration of the robust design space can now be performed. For the present experiment the technology scenario is fixed, so the shape parameters of all uncertainty scenarios considered are fixed at their listed values representing the present technology scenario.

Table 38: Experiment 1 - Technology Impact Distributions - Component Loading

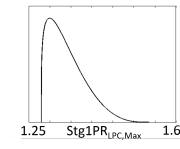
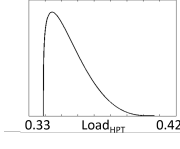
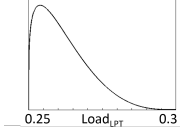
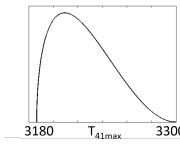
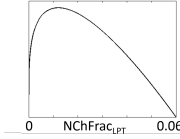
Impact	Min, Max, Alpha, Beta	PDF
$LPCstg1PR_{Max}$	1.28, 1.54, 1.25, 4.00	
$Loading_{HPT}$	0.339, 0.407, 1.25, 4.00	
$Loading_{LPT}$	0.25, 0.30, 1.25, 4.00	

Table 39: Experiment 1 - Technology Impact Distributions - Component Cooling

Impact	Min, Max, Alpha, Beta	PDF
$T_{41,Max}$	3186.5, 3300.0, 1.5, 3.0	
$NChBldFrac_{LPT}$	0.00, 0.06, 1.25, 2.00	

Just as with the deterministic assessment, exploration of the design space can be performed. However, candidate designs are now evaluated by mapping their design cycle settings to corresponding *likely estimates* of performance at various levels of confidence. The designer can input his desired level of confidence in any performance estimates and the surrogate evaluations instantaneously account for the uncertainty distributions implicit in the surrogate models and provide an estimation at the desired confidence interval. There is no need to further process any probabilistic data. The performance spread due to the uncertainty present is implicit in the surrogate model representations of the probabilistic design space. Repetitions of the design simulation are not necessary, allowing a computational budget to be better allocated to evaluate additional candidate designs, enabling the designer to more richly sample the design space under consideration. This eliminates any additional post-processing required to determine how the uncertainty distributions affect the spread in performance of candidate designs being evaluated.

Similar to what was done during the deterministic assessment, constraints are imposed on the designs explored. For the probabilistic assessment, the temperature constraints now have variability in likely levels to be experienced by each candidate design due to the technology uncertainty. Therefore, constraints are imposed on particular confidence interval level temperatures. For example, in the present exploration, the HPC exit temperature constraint ($T_{3,max}$) and the LPT inlet temperature constraint ($T_{45,max}$) are imposed for two confidence interval levels. Figure 55 shows the feasible probabilistic design space when the temperature constraints are imposed on the 50% confidence interval temperature estimates. This means that surrogate evaluations of the temperature estimates are made with an inputted desired confidence of 50%. As before, the figure shows the boundaries of the design space due to the temperature constraints imposed. The $T_{3,max}$ constraint obviously limits the higher OPR designs while the $T_{45,max}$ constraint limits the lower HPC pressure ratio designs. Since designs are filtered based on their 50% confidence interval temperature estimates, this means that the remaining designs have a 50% probability in satisfying constraints under the current uncertainty scenario, which may not offer the designer enough confidence in his decision.

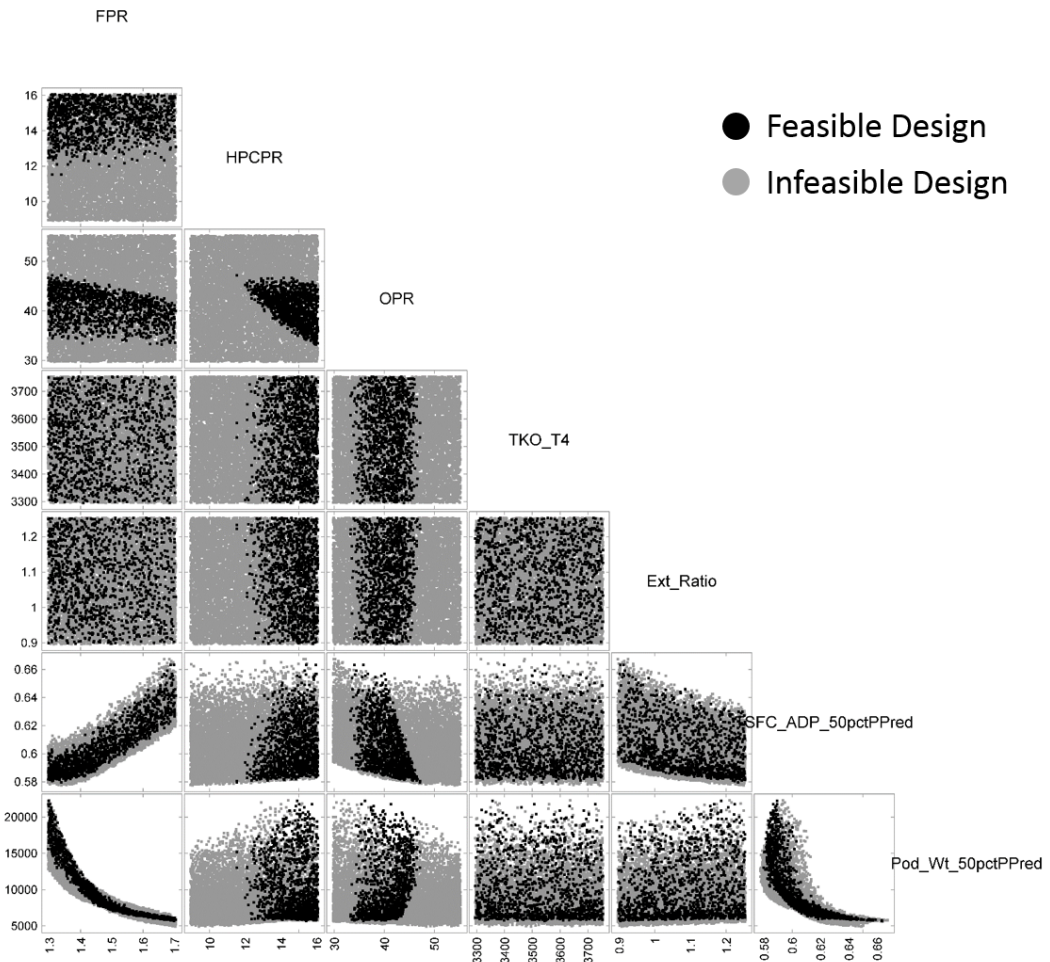


Figure 55: Experiment 1b - Constrained scatterplot of 50% likely performance data.

When making design decisions, the designer would like to have as high confidence as possible that mitigations, or redesigns, are not going to be required. These modifications not only cost development time and capital, but also incur performance compromises due to operating the design at conditions other than what was originally intended for the design. Therefore, constraints should be imposed based on high confidence level predictions. Figure 56 shows the feasible design space when the two temperature constraints are imposed on the 95% likely temperature estimates. Note that although the designer now has a high confidence in the temperature estimates for which constraints are imposed, the feasible design space has been greatly reduced. The small population of feasible designs have been highlighted with ellipses to better identify the very small feasible design region. A small band of candidate designs with intermediate OPR designs and high pressure ratio HPCs that provide feasibility with the high level of desired confidence. The designer now has a couple of options. He can either live with the temperature limits that are assumed for the HPC exit and LPT inlet and go with a design in the extremely narrow feasible design space, or the designer can explore options to increase the temperature limits in order to grow the feasible design space of the engine under consideration.

Figure 57 shows how the feasible design space can be increased in size by increasing the HPC exit temperature constraint ($T_{3,max}$) by 50 degrees Fahrenheit. Assuming that the 1,700 degree Rankine limit previously imposed was the current material limit, either new materials would be needed for the HPC, or a cooling technology such as an intercooler may need to be developed if this temperature limit push is desired.

Figure 58 shows the increase in the feasible design space for the engine design under consideration with a LPT inlet temperature limit increase of 50 degrees Fahrenheit. Assuming the LPT inlet temperature constraint ($T_{45,max}$) corresponds to an uncooled LPT, in order to allow for the increase in inlet temperature, either cooling flow would now be required for the first LPT stages or advanced material technologies would need to be pursued.

Even though increasing the level of confidence in constraint values decreases the feasible design space within which designs can be selected, it is still desirable to the designer. Decisions that have accompanying high levels of confidence often carry more weight than

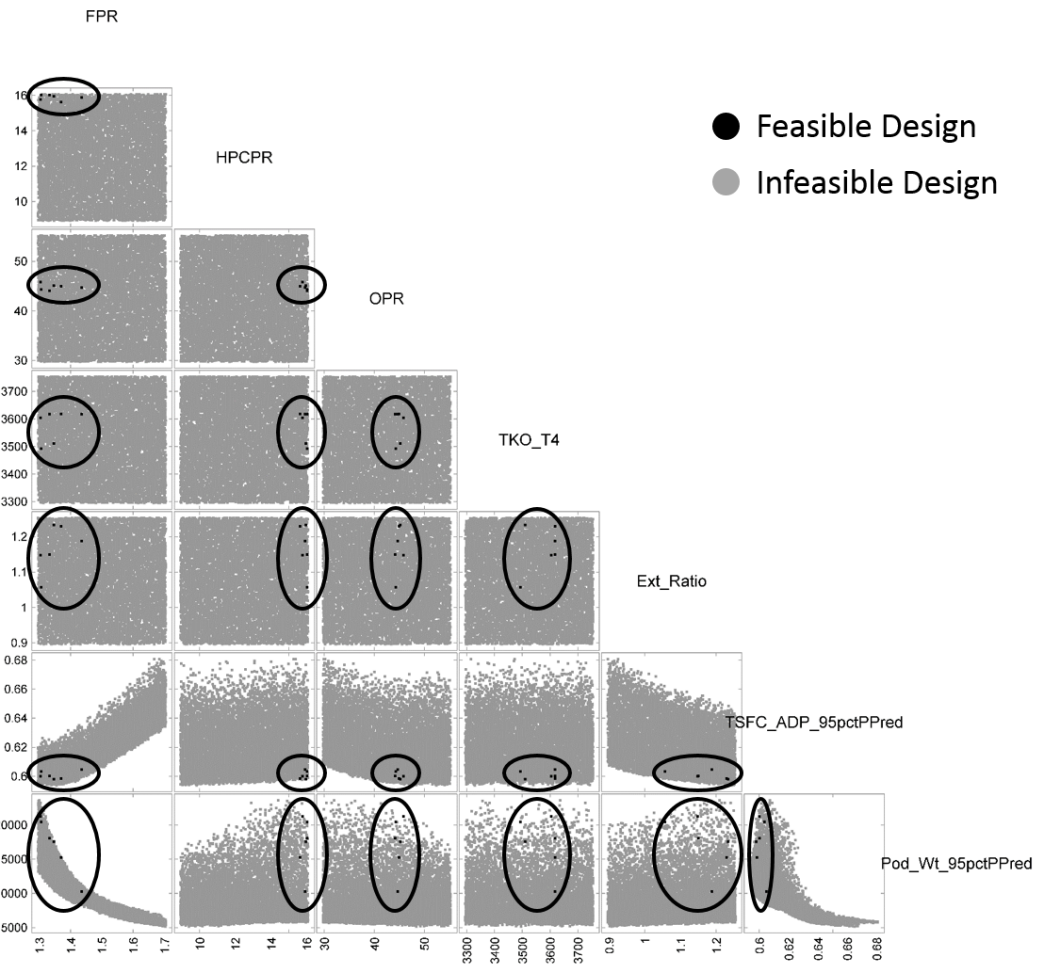


Figure 56: Experiment 1b - Constrained scatterplot of 95% likely performance data.

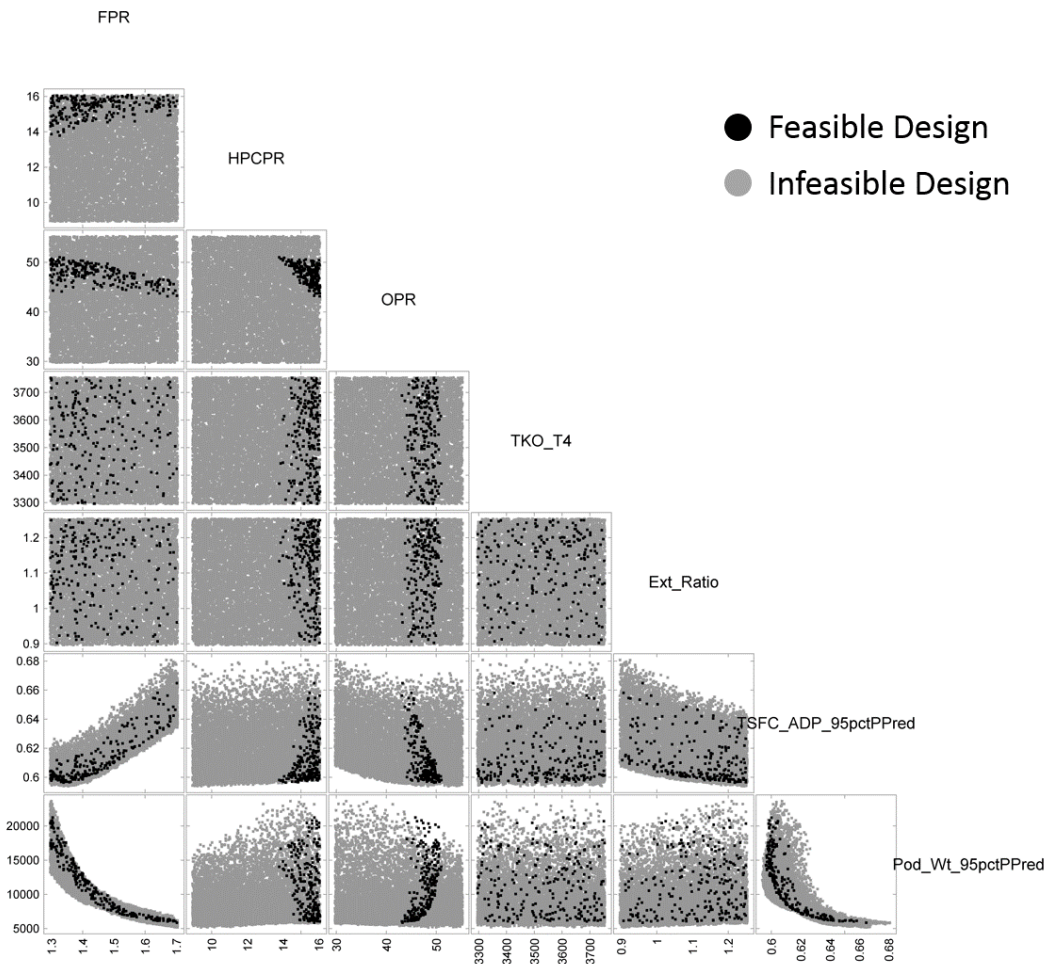


Figure 57: Experiment 1b - Constrained scatterplot of 95% likely performance data, HPC exit temperature limit increased by 50 degrees.

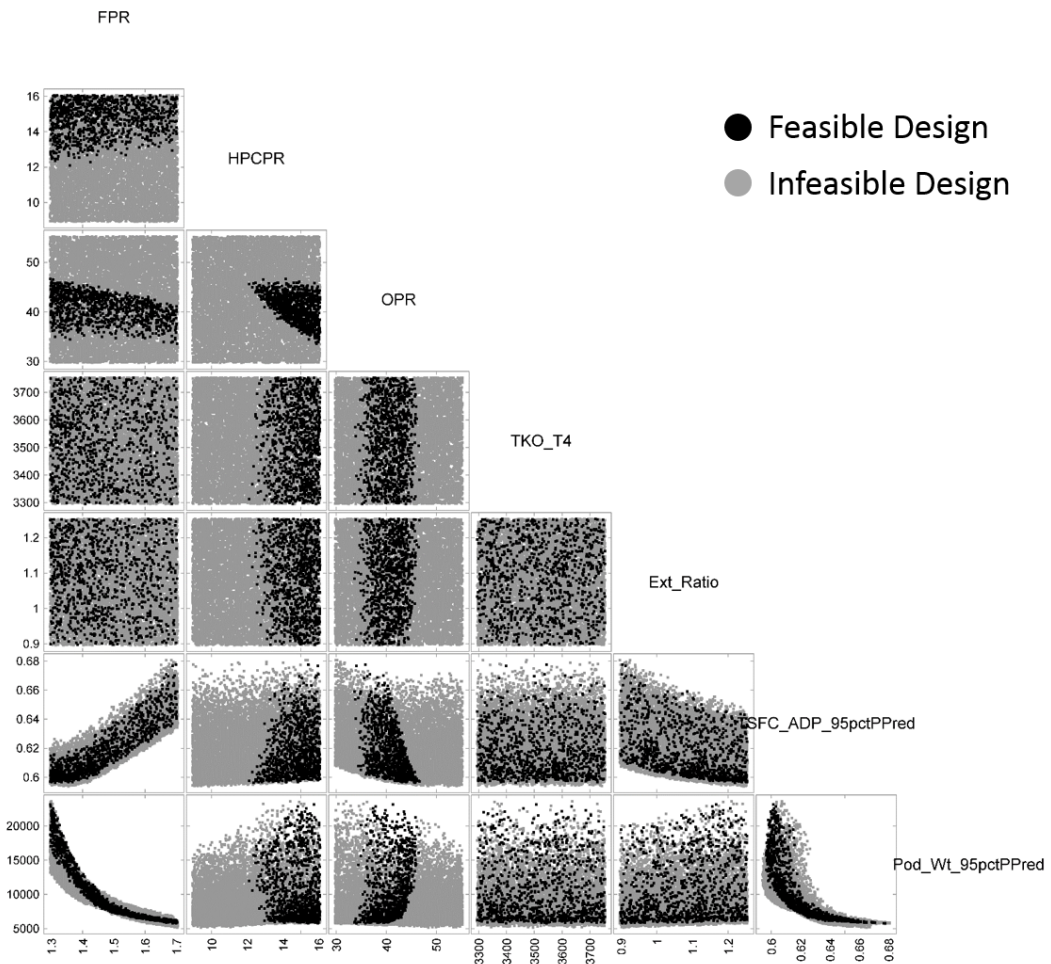


Figure 58: Experiment 1b - Constrained scatterplot of 95% likely performance data, LPT inlet temperature limit increased by 50 degrees.

decisions made that either have low confidence or no measure of likelihood associated. Mitigations such as technology infusion or alternative design approaches can be identified and explored earlier in the design process instead of after initial design selections have been made. The uncertainty assumed present can be addressed and attempts can be made to reduce the variability or improve the modal value.

4.2.4.2 TOPSIS Evaluation of Candidate Designs

After exploring the probabilistic design space considered and making any changes in the assumptions made in the technological capabilities considered in the exploration, TOPSIS evaluation and design selections can be made. For consistency with the deterministic assessment, the temperature limits imposed, although limiting the feasible design space and likely would be visited in reality, have been kept at the levels for which they were imposed for the deterministic study. Figure 59 displays the feasible design region when imposing the temperature limits on their corresponding 50% likely estimated levels, where each candidate design is colored based on its corresponding TOPSIS score. Again, the TOPSIS score is determined purely based on the TSFC at ADP conditions and the bare engine weight estimate, compromising between the two. The figure shows that the design fan pressure ratio (π_{Fan}) is the key driver in determining a candidate's TOPSIS score. An intermediate FPR design provides a compromise between TSFC and engine weight. Low FPR designs, although providing maximum thermal efficiency by resulting in high BPR designs, would require significant weight increases due to the fan and nacelle diameter increases. On the other hand, high FPR designs would allow for a more compact engine outer diameter, but would suffer in its propulsive efficiency during operation.

A similar multivariate scatterplot constrained by 95% likely temperature levels as shown in Figure 56 but colored by TOPSIS score is unnecessary due to the lack of feasible candidate designs. However, if allowing for both the HPC exit and LPT inlet temperature limits to increase by 50 degrees Fahrenheit each, a multivariate scatterplot can be constructed with a significant number of feasible designs colored by their TOPSIS score. This plot is shown in Figure 60. Comparison of the two multivariate scatterplots allows the designer

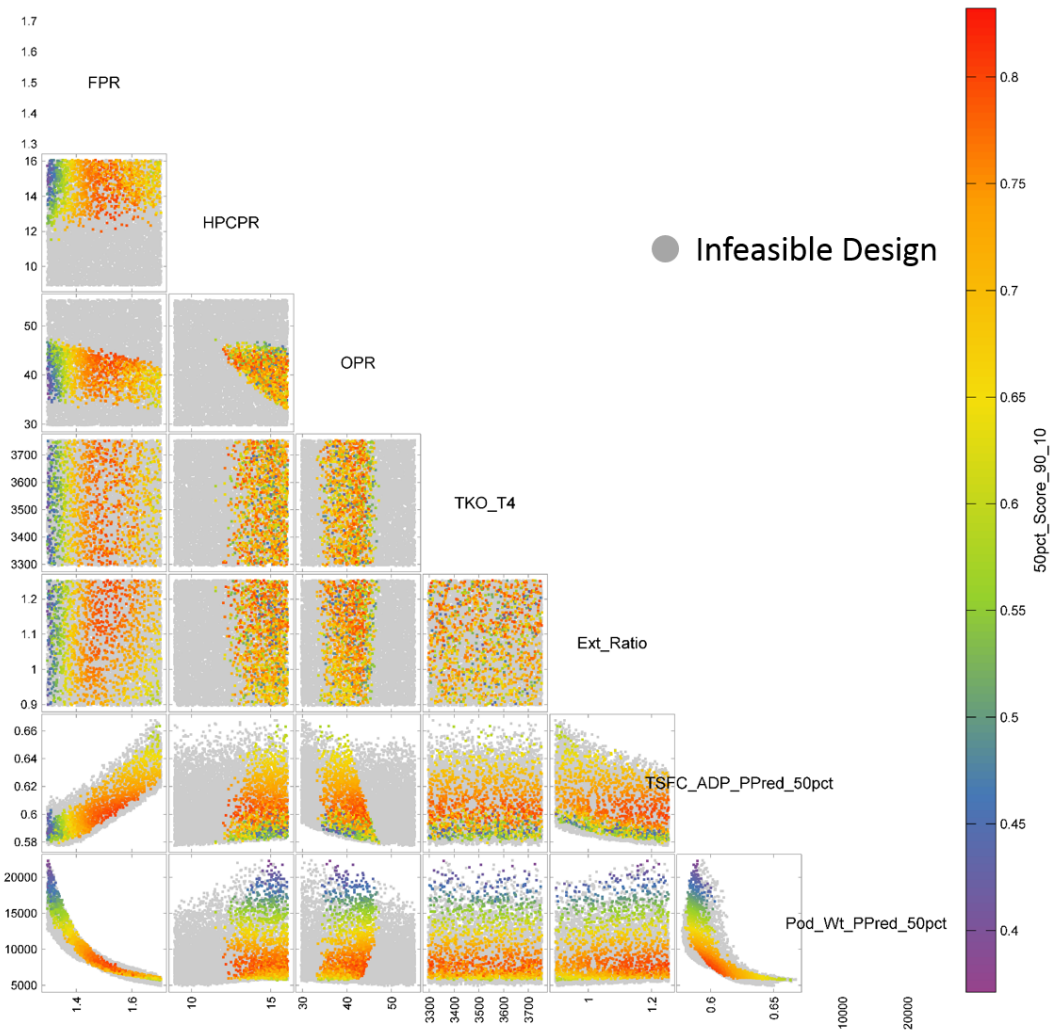


Figure 59: Experiment 1b - Constrained scatterplot of 50% likely performance data, colored by TOPSIS score.

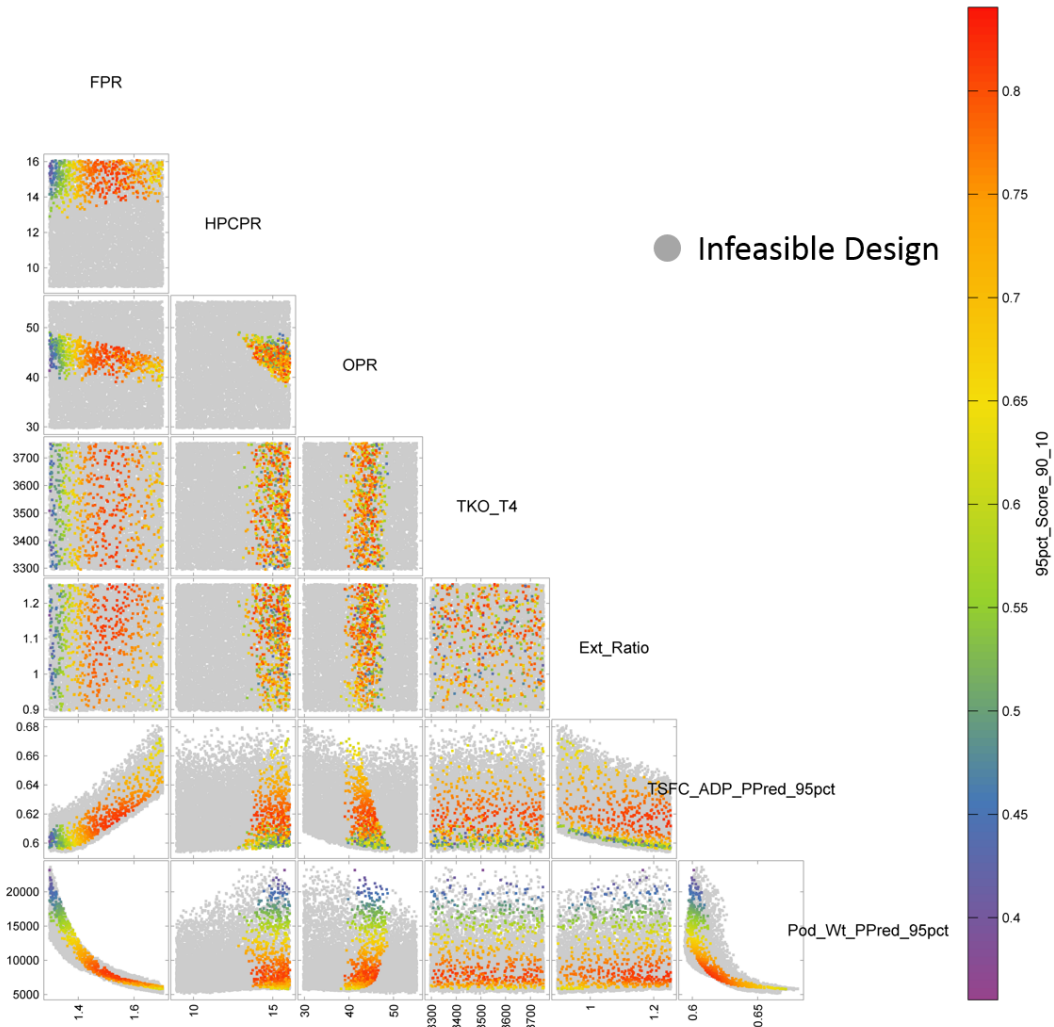


Figure 60: Experiment 1b - Constrained scatterplot of 95% likely performance data, colored by TOPSIS score, HPC exit and LPT inlet limits both increased by 50 degrees.

to see the trade in confidence and the temperature limits imposed on the feasible design space and the shift in peak TOPSIS scores in terms of design cycle settings. However, the multivariate plots only allow the designer to visualize two dimensional slices of the design space, preventing visualization of the combinatorial design space of all of the design variable settings that maximize the TOPSIS score within the feasible design region.

Multivariate plots offer designers the ability to easily see the trends and tradeoffs between design parameter setting and corresponding performance on a given plane of two design parameters. But when analyzing and selecting designs, parallel coordinate plots

allow the designer to visualize complete sets of design parameter settings and those candidates' corresponding performance. Looking back at Figures 59 and 60, the plots show the change in the feasible design region and shifts in design parameter settings that offer high TOPSIS scores, but they do not allow the designer to visualize complete design cycles that maximize likely performance. Figures 61 and 62 are parallel coordinate plots showing the same feasible designs that are shown in the multivariate scatterplots above. Figure 61 shows the top scoring designs after constraining the feasible design region using the original temperature limits using their corresponding 50% likely estimates. Figure 62 similarly shows top scoring designs that are feasible based on each design's 95% likely temperature limits that were relaxed by 50 degrees Fahrenheit each.

Interesting observations can be made when comparing the feasible design space shifts when changing the confidence interval of temperature estimates on which constraints are imposed, and the temperature limits themselves. The first coordinate plotted in each is the design fan pressure ratio. The range of intermediate values does not change when the method of imposing temperature limits is changed. However, there are significant shifts in the remainder of the design variables. Increasing the temperature limits and the level of confidence desired in the temperature levels shifts the mean levels of HPCPR, OPR, and T4 at takeoff to higher levels for the top scoring (feasible) design candidates. The impact of increasing the HPC exit and LPT inlet temperatures on the most optimal design variable settings shows that pursuit of technologies that increase those limits may also have accompanying need to mature material temperature limits in other regions of the engine, such as the first stage of the HPT which experiences the hottest temperature levels experienced in the entire engine.

Table 40 summarizes the favorable design regions based on the type of assessment performed. For the deterministic assessment, the probabilistic assessment when 50% confidence interval estimates were made, and the probabilistic assessment when 95% confidence interval estimates were made and the two temperature limits were increased by 50 degrees, the ranges of design variable settings are listed that offer top scoring feasible candidate designs.

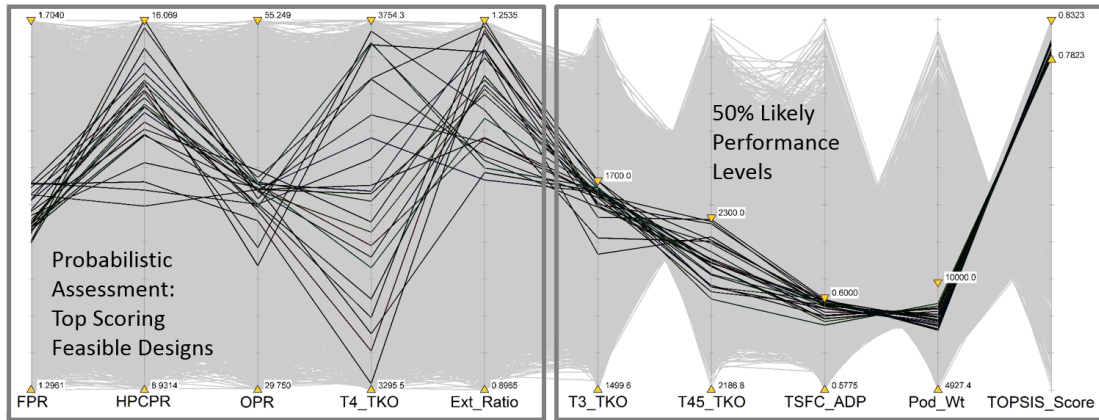


Figure 61: Experiment 1b - Constrained parallel coordinate chart of 50% likely performance data with highest TOPSIS scores.

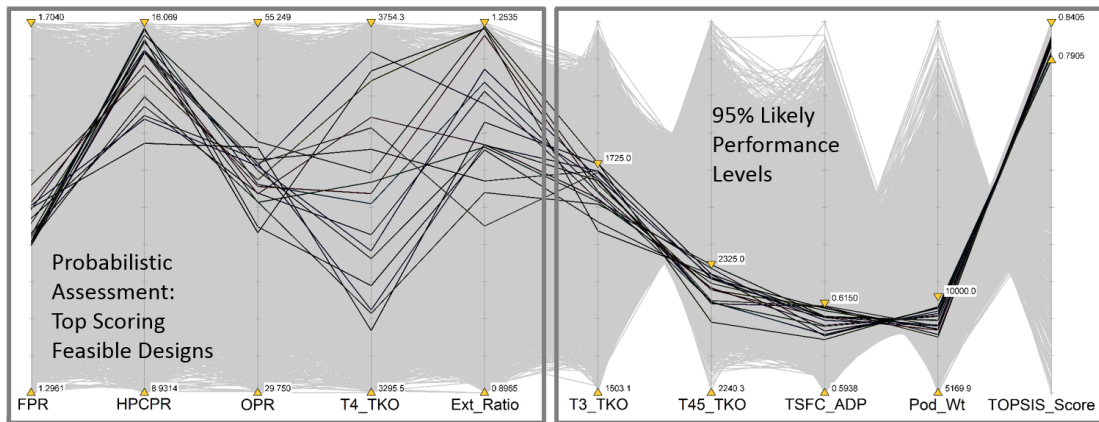


Figure 62: Experiment 1b - Constrained parallel coordinate chart of 95% likely performance data with highest TOPSIS scores, HPC exit and LPT inlet limits both increased by 50 degrees.

Table 40: Experiment 1 favorable design regions that provided top scoring engines for the deterministic assessment, probabilistic assessment with 50% CI metric estimates, and probabilistic assessment with 95% CI metric estimates.

Parameter	Det Select		Prob Select 50% CI		Prob Select 95% CI Tpush	
	min	max	min	max	min	max
π_{Fan}	1.37	1.62	1.46	1.62	1.42	1.64
π_{HPC}	8.97	16.07	12.00	16.09	11.29	16.07
$\pi_{Overall}$	31.98	46.34	36.81	45.19	34.69	50.69
$T_{4,max}, degR$	3,296	3,754	3,304	3,754	3,296	3,754
ER	0.92	1.25	1.04	1.25	0.99	1.25
95% CI TSFC at ADP, $lbm/lbf/hr$	0.5961	0.6264	0.6075	0.6252	0.5989	0.6274
95% CI Wt, lbm	6,535	10,433	6,568	9,316	6,305	9,813

The shifts in the favorable design regions show design responses due to changes in the assumed uncertainty and the desired confidence interval estimates that were used to evaluate designs under uncertainty.

4.2.4.3 Comparison to Deterministic Design Selection

This experiment has allowed for comparison between the methods of design exploration and selection and highlighted the ERDS method and its added capabilities when applied to a single engine design problem with a fixed set of technology assumptions. The added steps of the ERDS process allows for more information to be drawn for exploration of the design space by accounting for uncertainty present in the design. A traditional deterministic method of exploration and design selection was used to select a design in the previous experiment. The ERDS probabilistic method was then used for to make design selections with two levels of associated likelihood. The design selections made using the two methods are shown in Table 41.

The design cycle settings of the deterministic selection were used to evaluate the probabilistic surrogate models predicting the likely performance of the candidate design, which is shown in the first column of data. Two candidate designs were selected that had the highest TOPSIS scores of the feasible designs remaining after imposing temperature constraints.

Table 41: Experiment 1b Candidate design selections from probabilistic assessment with top 95% likely TOPSIS score that maintained a 95 % likelihood of having feasible temperature levels within imposed limits, compared to Experiment 1a Deterministic Selection. Prob Select 1 within original temperature limits, Prob Select 2 within relaxed temperature limits.

Parameter	Det Selection	Prob Select 1	Prob Select 2
π_{Fan}	1.458	1.435	1.522
π_{HPC}	9.004	15.90	14.39
$\pi_{Overall}$	44.15	44.68	42.15
$T_{4,max}$ ($^{\circ}R$)	3,520	3,617.	3,309.
ER	1.235	1.188	1.245
95% likely $T_{3,max}$ ($^{\circ}R$)	1,680.	1,700.	1,679.
95% likely $T_{45,max}$ ($^{\circ}R$)	2,418	2,298.	2,328.
95% likely TSFC at ADP	0.6021	0.6048	0.6134
95% likely Pod Weight (lbm)	8,978	10,240	7,981.

The design labeled *Prob Select 1* was selected after imposing the original temperature constraints consistent with the limits used to constrain the deterministic design space. The design labeled *Prob Select 2* is the top scoring feasible design after relaxing the two temperature limits by 50 degrees Fahrenheit. The major difference between the deterministic design selection and the probabilistic design selection as far as the cycle settings are concerned is in the design pressure ratio of the HPC. In order to satisfy the LPT inlet temperature constraint ($T_{45,max}$), the core of the engine must be loaded up in order to increase the required expansion of the HPT, reducing the remaining thermal energy, and corresponding temperature, of the flow entering the LPT.

In terms of likely performance, the deterministic candidate is significantly better in terms of both TSFC and engine weight when compared to the probabilistic candidate with the same temperature limits imposed. However, the deterministic candidate violates the LPT inlet constraint by more than 100 degrees. By ensuring that the temperature levels likely will not violate the limits imposed, a compromise in performance is made. This highlights a key tradeoff in the present design. Either mitigate the candidate design selection made by the deterministic method by somehow allowing for the elevated LPT inlet temperature or live with the performance compromise for the current LPT inlet temperature limit.

However, if mitigation is an option, then the probabilistic ERDS method should be used to account for the increase in temperature limits so that the designer can take full advantage of the temperature limit increase and a wise selection of a new candidate can be made. Comparing candidates labeled *Prob Select 1* and *Prob Select 2*, by allowing for the imposed temperature limits to be relaxed by 50 degrees, the top score cycle shifts to a slightly lower pressure ratio HPC. Recall that the HPCPR was limited by the LPT inlet temperature limit, so by relaxing the limit, the core can be unloaded, allowing for the LP system to perform more work to achieve the design OPR. This in turn takes a small performance hit on TSFC but significant reduction in engine weight is achieved by moving to a lower bypass ratio design.

4.2.4.4 Capabilities Enabled and Presently Displayed by the ERDS Process

The ERDS method enables many capabilities when compared to a deterministic method for design evaluation and selection. Primarily, it enables the designer to have more confidence in his design decisions by accounting for the variability in design performance due to sources of uncertainty present in the design. Exploration of the design space can be performed just as during deterministic assessments, but now performance estimates have accompanying desired levels of confidence associated with those estimate. This experiment also highlights the iterative process of evaluating designs and considering further technology developments. The present experiment showed that possible increases in temperature limit constraints can enable better performance designs. However, the present design did not explore the possibility of advancing the technology impacts assumed in the technology scenario considered. A subsequent experiment found in Appendix E shows another enabling capability of the ERDS process: the ability to account for changes in the technology scenario on design selections and performance estimates. By having the ability to simulate various uncertainty scenarios, the designer can consider many development strategies to reduce uncertainty or push modal values of distributions to more desirable levels. Competing strategies can then be evaluated based on the implications on the design and resultant performance levels. This flexibility of the COMMENCE method is also highlighted in Experiment 4 where the fully

integrated method is applied to a multi-application, multi-tech level common core design problem.

4.2.5 Conclusions

This experiment serves to show the capabilities and advantages of the Enhanced Robust Design Simulation (ERDS) method when considering design decisions under uncertainty. Significant sources of technology and requirements uncertainty were accounted for in the cycle design of a high bypass direct drive turbofan engine. A deterministic assessment was performed under the technology scenario where all the most likely technology impacts are assumed to be realized, and a resulting candidate cycle selection was made that met the imposed temperature constraints while also offering a top performance score through TOPSIS evaluation. Considering the same technology scenario, but accounting for the sources of uncertainty assumed present, a probabilistic assessment was performed using the ERDS method and a candidate design was selected that offered a high likelihood of satisfying the imposed temperature constraints, while also providing the best likely cycle performance.

The cycle selections were compared in terms of their design cycle settings, and it was shown that the robust selection preferred cycle settings that ensured a high probability of not violating the imposed constraints. As an example, the design π_{HPC} of the deterministic design selection favored a low value, loading up the LPC in order to achieve the desired high $\pi_{Overall}$, allowing for higher design thermal efficiency. However, the robust design selection favored a high π_{HPC} design, loading up the HP shaft and requiring significant work to be performed by the HPT. This high HPT work reduced the HPT exit temperature, in turn lowering the LPT inlet temperature well below the imposed limit, giving the robust design a high probability of remaining below the LPT inlet temperature constraint, minimizing the likelihood of needing a costly design modification later in the engine development.

Comparison of the likely performance levels of each of the design selections were also performed, showing how the robust design selection traded likely performance assurance that the design will remain under the imposed temperature constraints. The deterministic

cycle selection, when evaluated using the probabilistic surrogate models, showed that it would likely violate the LPT inlet temperature constraint ($T_{45,max}$) under the assumed uncertainty distributions present in the scenario. This shows the value of using the ERDS method, allowing the designer to have a high level of confidence that candidate design selections will simultaneously meet the requirements and satisfy all constraints imposed on the design.

When integrated into the COMMENCE process, the ERDS method is used to perform two specific tasks. First, the ERDS method is employed to explore and evaluate clean sheet benchmark designs, which common core variant engine performance levels are evaluated against. This allows for quantification of likely performance penalties ensued by employing a common core design instead of producing a completely new design in response to a set of customer requirements. The ERDS method is also utilized in the COMMENCE method for the simultaneous design exploration of the common core defining cycle and all common core variant applications considered. The benefits of the ERDS method are taken advantage of, enabling rapid probabilistic performance estimation of many common core applications under various uncertainty scenarios to be performed. This allows the designer to account for and make selections for any number of common core variant applications projected to be introduced throughout the life of an engine program.

4.3 *Experiment 2: Engine Variant Upgrade Option Exploration*

- **Research Question 2:** *For a given gas turbine engine core, how should a common core engine variant design be simulated? What parameter(s) must be held to consistent values in order to maintain geometric and aerothermodynamic commonality between engine applications?*

Hypothesis 2: *In order to simulate a common core engine variant, design rules must be established and enforced that maintain the design level of corrected flow exiting the high pressure compressor. Maintaining HPC exit corrected flow at design conditions will ensure geometric similarity between common core applications. In order to provide significance to maintaining design HPC exit corrected flow, the design rules must also ensure that the compressor map design operating point is also fixed between common core applications.*

- **Research Question 3:** *What design options should be considered for common core engine variant applications in order to distribute development capital across the engine program by taking advantage of commonality, while also offering more design freedom when needed for more demanding applications?*

Hypothesis 3: *In order to provide a wide range of capabilities with common core applications, a range of design options should be available with differing levels of upgrade cost and design freedom. Geometrically fixed core and modified common core options should exist to allow for significant core power growth if needed. The common core applications should have design freedom in the LP system in order to be sized for a new set of customer requirements, and technology infusion should be considered for the core and/or the LP engine components in order to provide feasible, competitive common core solutions while placing preference on less expensive upgrade options when at all possible.*

- **Research Question 4:** *What range of capabilities can various common core design options achieve without significant compromises made in application performance?*

Hypothesis 4: *A common core variant engine is able to provide a specific range*

of capabilities while maintaining acceptable performance levels, based on the technology level of the variant design, the core size, the overall engine architecture, and the amount of design freedom permitted for the particular common core application considered.

- **Research Question 5:** *How should a common core engine program consisting of multiple variant design applications be evaluated?*

Hypothesis 5: *A common core variant engine should be evaluated based on the performance deviation from a benchmark, new centerline engine designed specifically for the application it is being designed for. In order to determine the amount of performance compromise made by utilizing a common core application, the benchmark engine should be designed for an identical set of requirements while under the same set of assumptions as were made for the common core variant design. A weighted sum of variant performance levels should be used in the overall evaluation of the engine program.*

4.3.1 Overview

This experiment serves to explore the various common core design options when faced with a new set of customer requirements. Each common core upgrade option allows for a different amount of design freedom when aiming to achieve a competitive common core solution for the new engine application. Exploring varying levels of design freedom will identify the tradeoff between the attainable variant engine performance and the savings due to having product commonality. This feasibility study also serves to test the logic used to model and simulate geometrically common engine core variant engines, a central requirement of the COMMENCE method. Deterministic design explorations will be performed for five discrete common core upgrade scenarios, which are listed in Table 42. It is assumed that the engine core which is to be utilized for a new application was designed and previously selected for a past application, and no design freedom is granted in the baseline core definition. The five scenarios grant the designer various levels of design freedom aiming to achieve thrust growth for the new engine application. The goal for each scenario is to attain as much thrust growth

as possible with minimal compromise made in cruise specific fuel consumption (TSFC).

It will be shown that the amount of thrust growth achievable by the variant engine application is limited by the amount of design restriction imposed on the new design. Scenarios 2.1 - 2.3 require a geometrically fixed core to be used, while Scenarios 2.4 and 2.5 allow the core to be modified while still remaining geometrically common. The impacts of technology infusion into various regions of the engine will be examined as well. The three fixed geometry core scenarios assume various levels of technology infusion, from Scenario 2.1 where no technology advancement is allowed in the new application, to Scenario 2.3 where component efficiency and weight improvements are applied to both the low pressure and high pressure engine components. Similarly, Scenarios 2.4 and 2.5 allow for the comparison of common core responses with and without technology infusion of the engine variant.

While the first three scenarios considered have fixed geometry core components, Scenarios 2.4 and 2.5 allow for modification of the common core to be performed, enabling increases in the core flow capacity. This is expected to achieve significant thrust growth with less compromise in TSFC performance. Although modifications are allowed for these two upgrade scenarios, limitations will be imposed on the core design in order to enforce the core to remain geometrically common between the baseline core definition and the variant engine's core. The term *common core* in this context refers to the HPC, combustor, and HPT being geometrically similar. Flow areas and component stage counts remain consistent

Table 42: Experiment 2 - Variant engine upgrade scenarios.

Scenario	Design Change(s)	Technology Level
2.1	LP System Redesign	Parent Tech Level
2.2	LP System Redesign	LP System Tech Infusion
2.3	LP System Redesign	LP and HP System Tech Infusion
2.4	LP System Redesign and Flared HPC	Parent Tech Level
2.5	LP System Redesign and Flared HPC	LP and HP System Tech Infusion

between common core applications, with slight modifications when permitted. The turbomachinery maps of the baseline core's HPC and HPT are also used for the common core variant cycle, with alterations made when allowed, while operating within the rules imposed by the COMMENCE method to maintain and ensure commonality between common core engine applications.

4.3.2 Scenarios Considered

As shown in Table 42, there are five scenarios which are considered in the present experiment. They are ordered by the level of upgrade complexity of the engine variant design upgrade. The first scenario applies a fixed engine core to a new low pressure system and the variant engine is kept at its parent technology level. This represents a realistic scenario where design responses are to be provided for two unique sets of customer requirements within the same time frame. In reality, knowing the engine core is to be immediately applied to two engine applications, the designer would likely make considerations when selecting a baseline engine core in order to offer the best compromise for the two applications. However, another realistic decision would be to make a design selection that all but guarantees one application will successfully make it to market, while secondary considerations are made for the less likely application. The present experiment more closely follows the latter program decision, assuming that the core definition has been previously established by the baseline engine, whose characteristics are listed in Table 43.

The second scenario considered is very similar to the first, but allows for technology advancement of the low pressure system. The engine core remains fixed between the design and the variant engines, but the low pressure system is designed for the new application and has a more advanced technology level than the baseline engine for which the core was sized and selected. This scenario represents the case when an off-the-shelf engine core is applied without modification to a new application, while taking advantage of the latest technology development by utilizing advanced low pressure components in the new engine application.

The third scenario allows for technological advancement of both the low pressure and high pressure components, but the engine core geometry remains fixed. This is an example

Table 43: Experiment 2 - Baseline cycle used to size the common engine core.

Category	Variable	Baseline Value
Cycle	π_{Fan} at ADP	1.685
	π_{HPC} at ADP	9.369
	$\pi_{Overall}$ at ADP	30.094
	$T_{4,max}$ (<i>degR</i>)	3,300 <i>degR</i>
	ER at ADP	1.018
Tech Level	η_{Fan} at ADP	0.895
	η_{LPC} at ADP	0.924
	η_{HPC} at ADP	0.871
	η_{HPT} at ADP	0.889
	η_{LPT} at ADP	0.900
Attributes	W_{25R} at ADP	49.6 <i>lbm/s</i>
	W_{3R} at ADP	6.68 <i>lbm/s</i>
	BPR at ADP	5.11
	Fan Diameter	62.1 <i>in</i>
	Takeoff Thrust	22,783 <i>lbf</i>
	Top of Climb Thrust	5,962 <i>lbf</i>
Performance	TSFC at ADP	0.6640 <i>lbm/lbf/hr</i>
	Pod Weight	5,652. <i>lbm</i>

of a fixed core that has been re-bladed with more advanced materials or a better 3D design of the blades to improve the core thermal efficiency. This is a common scenario that is seen in industry: the company would like to make small changes to an existing design with which they have much experience. However, the company has advanced their state-of-the-art since designing the original core. Therefore, in order to take advantage of their latest developments, they infuse their latest technologies within the fixed geometry core design, gaining efficiency and pushing the amount of power they can achieve with the fixed geometry core design. These technology infusions can be applied to both new engine applications as well as existing off-the-shelf engines.

The fourth and fifth scenarios are common core applications, but with significant core design modifications aiming for more extensive power growth of the core. A *flared* or *zero-staged* HPC is allowed, offering the core more working fluid with which it can add thermal energy to the flow in order to do more work on the LPT, producing more propulsive power. Flaring or zero-staging the HPC requires significant modification to the core components.

While the only flow area changing is that of the HPC inlet, this option would require re-blading of both the HPC and HPT components, requiring significant development costs. Although, it is assumed that this modification would still be significantly less expensive than the development of an all-new core design. The fourth scenario allows for a flared HPC, while remaining at the parent technology level of the design engine from which the original core is sized. The fifth scenario allows for a modified core, a redesigned LP system, and additional technology advancement of both the LP and HP systems.

For each scenario, any technological advancements allowed will be applied to the variant engine design, and design space explorations will be performed within each corresponding set of rules, aiming to achieve the most thrust growth possible with the smallest compromise in SFC performance. The following section describes how the modeling and simulation environment is setup to allow for common core engine variant design simulations to be performed while ensuring the new application's engine core is geometrically common with the baseline core definition.

4.3.3 Common Core Engine Variant Design - Modeling and Simulation Environment

This experiment is the first to utilize the design-variant two engine model simulation within a single modeling environment. The two engines, the *core defining design engine* and the *common core variant engine*, utilize the same NPSS components making up their architectures. The core components, the ones shaded in Figure 63, are designed during the core defining design engine simulation and then applied to the variant engine simulation. This enforces a common core between the design and variant engine simulations. Table 44 contains the major NPSS components and their DESIGN or OFF-DESIGN modes during the simulations of the design and variant engine cycles. When in *DESIGN* mode, the components are allowed to be sized to exactly meet requirements imposed at the design points considered. Turbomachinery maps are scaled accordingly to provide the necessary flow levels at peak efficiency to meet the requirements, such as desired thrust levels. When in *OFF-DESIGN* mode, the components are not allowed to scale, and they operate at speeds and corresponding efficiency levels that offer a steady state flow solution at the inputted

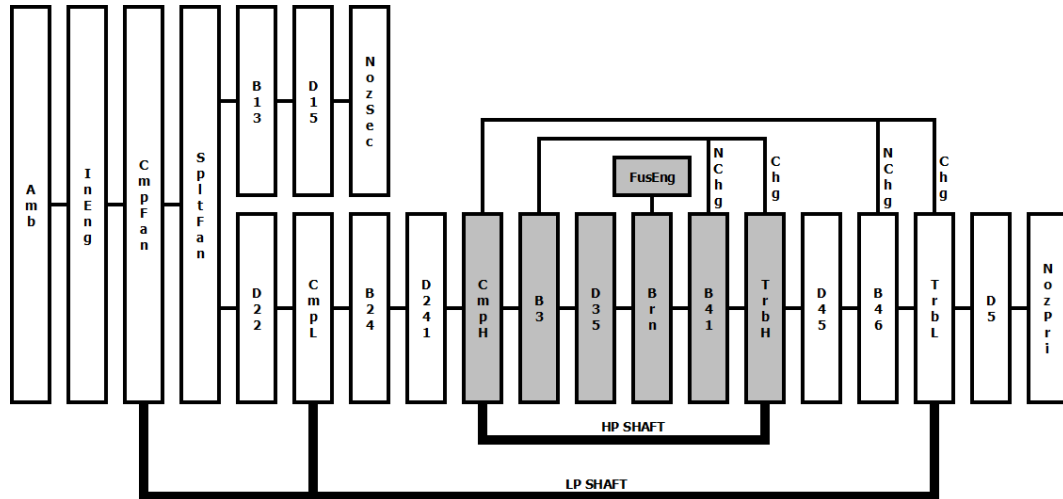


Figure 63: NPSS model components making up the physics based separate flow turbofan engine model. Gray components make up the engine core which remains geometrically common between design and variant engine applications.

engine power setting. The turbomachinery components use the same performance maps that were previously scaled during a design run. Its operation is allowed to vary during off-design simulations, but the map is not allowed to scale. For the common core variant simulation, this is how the core components operate, even at design point conditions.

Figure 64 represents the flow of information during the engine simulations within the modeling environment. First, the design engine is simulated. In the case of the present experiment, the design engine is fixed to the baseline definition in Table 43. Once the design engine is sized and evaluated, the core definition is set, in some cases modified by technology infusion and/or HPC flaring, and is then used for the variant engine simulation.

In addition to having the core geometry fixed during the simulation of the common core variant design, the four design rules presented in Chapter 3 are used to enforce commonality in the variant performance characteristics. These rules are described below.

- At the ADP design point, the HPC inlet corrected flow must be consistent between the design and variant run. When employing a flared HPC, the inlet corrected flow is scaled by a desired amount.
- At the ADP design point, the HPC exit corrected flow must be consistent between

Table 44: Experiment 2 - Major NPSS model components and their DESIGN or OFF-DESIGN status during Design and Variant engine simulations.

Component	Description	Design Sim Mode	Variant Sim Mode
InEng	Inlet	DESIGN	DESIGN
CmpFan	Fan	DESIGN	DESIGN
SpltFan	Splitter	DESIGN	DESIGN
NozSec	Bypass Nozzle	DESIGN	DESIGN
CmpL	LPC	DESIGN	DESIGN
TrbL	LPT	DESIGN	DESIGN
NozPri	Core Nozzle	DESIGN	DESIGN
ShL	LP Shaft	DESIGN	DESIGN
CmpH	HPC	DESIGN	OFF-DESIGN
Brn	Combustor	DESIGN	OFF-DESIGN
TrbH	HPT	DESIGN	OFF-DESIGN
ShH	HP Shaft	DESIGN	OFF-DESIGN

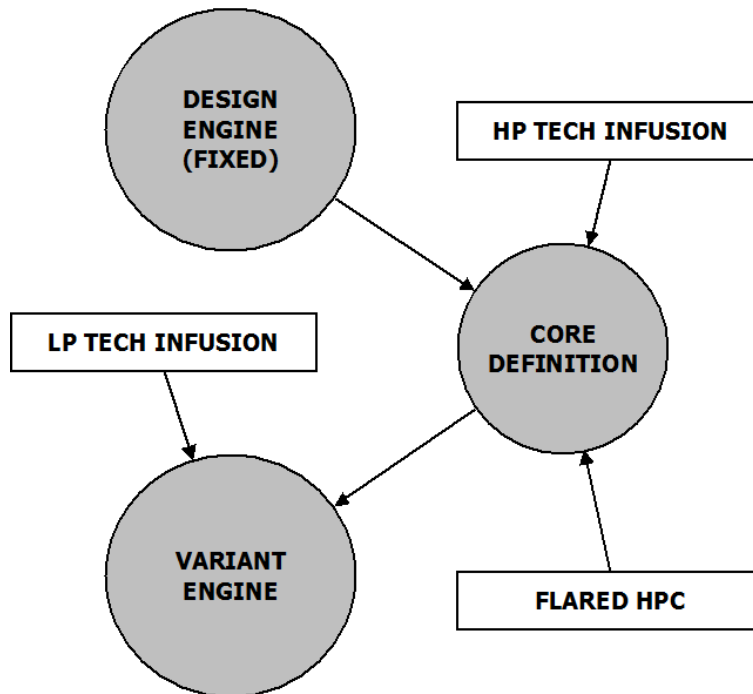


Figure 64: Experiment 2 - Common core modeling and simulation environment representation.

the design and variant run at max design power.

- The design map operating point of the HPC must remain consistent between the design and variant run.
- At ADP, the design HPC polytropic efficiency is held consistent between the design and variant run. When technology infusion is allowed on the core, the polytropic efficiency is increased by the permitted amount.

The NPSS code used to prepare the common core variant simulation and enforce geometric commonality of the engine core between the design and variant simulations can be found in Appendix A.

4.3.4 Common Core Technology Infusion

For the present experiment, there are two particular technology infusion sets which are to be applied during the design of the common core variant engine. The first is infusion of advanced technology into the low pressure system of the engine. This technology package impacts are listed in Table 45. For each of the turbomachinery components, the efficiency is increased by a quarter of a point with respect to the design engine efficiency at the ADP design point. In addition to efficiency improvements, each of the components of the LP system have weight reductions of 5 percent each, representing either material or manufacturing improvements. Again, these improvements are applied during the design of the common core variant engine, not after sizing of the LP system components. For the scenarios that include LP system technology infusion, the technology improvements are applied and the LP system design is selected.

In addition to the LP system technology package, two of the five scenarios considered in the present experiment allow for HP system technology infusion. Similar to the LP system technology package, each of the high pressure system components, making up the core, have efficiency improvements of a quarter of a point and the component weight reductions of five percent. Again, these efficiency improvements are applied to the engine components before exploring the design space and making selections. Also of note, the efficiency improvement

Table 45: Experiment 2 - Low pressure system technology infusion - scenario impacts on variant design.

Variable	Units	Impact
$\Delta\eta_{Fan,Var}$	-	+0.0025
$\Delta\eta_{LPC,Var}$	-	+0.0025
$\Delta\eta_{LPT,Var}$	-	+0.0025
$sWt_{Fan,Var}$	-	$\times 0.95$
$sWt_{LPC,Var}$	-	$\times 0.95$
$sWt_{LPT,Var}$	-	$\times 0.95$
$sWt_{ShL,Var}$	-	$\times 0.95$
$sWt_{NozPri,Var}$	-	$\times 0.95$
$sWt_{NozSec,Var}$	-	$\times 0.95$

of the HPC is applied as a polytropic efficiency improvement. Due to the possibility of the HPC having increased flow capacity, as enforced by the variant design rules previously established, consistent technology level efficiency across common core applications is enforced using the polytropic efficiency. This allows for efficiency levels of baseline or advanced variant HPC designs to be held consistent across applications, regardless of whether variant engines employ HPC designs with increased flow capacity and accompanying pressure ratio increases. The HPC efficiency impacts are the exception, with all other component efficiency improvements realized on the design adiabatic efficiency of the components. For the single fixed core upgrade scenario that contains core technology improvements, Scenario 2.3, the core design is only allowed efficiency improvements and weight reductions. No geometry or design characteristics are allowed to the core for Scenario 2.3. However, Scenario 2.5 allows both technology improvements and the HPC flow capacity is allowed to grow in order to increase design core air flow.

4.3.5 Utilization of Existing Surrogate Models

Just as with the previous experiment, the surrogate models produced at the beginning of this chapter are utilized for this experiment. By considering this study before sampling the physics based modeling and simulation environment for surrogate training, it was ensured that the input variable ranges sampled encompassed the input values for the fixed core

Table 46: Experiment 2 - High pressure system technology infusion - scenario impacts on variant design.

Variable	Units	Impact
$\Delta\eta_{HPC,Var}$	-	+0.0025
$\Delta\eta_{HPT,Var}$	-	+0.0025
$sWt_{HPC,Var}$	-	$\times 0.95$
$sWt_{HPT,Var}$	-	$\times 0.95$
$sWt_{ShH,Var}$	-	$\times 0.95$

defining design engine used in this study. Also, the variant design and noise variable ranges were selected for trained such that all possible values considered in this study fall within the originally sampled ranges for surrogate training.

Whenever utilizing surrogate models, it is important that when making design selections that the surrogate models are accurately predicting the performance characteristics of the candidate design. This experiment serves as a good application for testing surrogate model fit quality for design selections. For all candidate designs selected in this experiment, it was ensured that all surrogate models accurately predicted the metric responses of interested within one percent of the physics based model value. This provides added confidence that the deterministic and probabilistic surrogates used throughout this chapter provide accurate responses.

For this deterministic experiment, the deterministic surrogate models produced to represent the common core variant engine, *Surrogate Set B*, are utilized. Recall from Equation 47 that these surrogate models are functions of all control and noise variables for both the core defining design engine and the common core variant engine.

$$\mathbf{SSB}_{\text{Var,Det}} = f(\mathbf{CV}_{\text{Des}}, \mathbf{NV}_{\text{Des}}, \mathbf{CV}_{\text{Var}}, \mathbf{NV}_{\text{Var}}) \quad (47 \text{ Revisited})$$

4.3.6 Variant Engine Design Space Exploration

For each of the five scenarios considered in this experiment, the variant engine design space is explored. As was discussed in Chapter 3, a typical set of control variables explored for a new centerline engine design is contained in Table 18, which has been repeated for clarity. Design cycle parameter settings are explored as well as thrust levels, sizing the engine to

Table 18 Revisited: Typical control variables and corresponding ranges for a clean sheet turbofan engine design exploration.

Variable	Units	Min	Max
π_{Fan} @ ADP	-	1.3	1.7
π_{HPC} @ ADP	-	9.0	16.0
$\pi_{Overall}$ @ ADP	-	30.0	55.0
T_4 @ TKO	$^{\circ}R$	3,300	3,750
ER @ ADP	-	0.90	1.25
$NChg_{LPT}$	% W_{25}	0.00	0.06
F_N @ TKO	<i>lbf</i>	20,000	30,000
F_N @ TOC	<i>lbf</i>	5,900	8,850

meet the customer requirements. The common core variant control variables presented in Chapter 3 are again listed in Table 19, along with the ranges to be explored. In contrast to how the engine design space was explored in the previous single application design problem in Experiment 1, common core variant design explorations use variables that control the off-design operation of the core rather than variables than engine sizing variables. This is directly related to the fact that the variant engine design is not a clean sheet design. Its capabilities are limited by the lack of engine core design freedom. So instead of inputting performance levels like thrust that a common core variant engine may not be able to achieve, a power setting parameter such as the core corrected speed or combustor exit temperature is used, and the resulting variant capabilities (i.e. the achievable thrust levels) are determined. Notice also that the design pressure ratio of the HPC is no longer a control variable for the variant cycle. The variant design HPC pressure ratio at its aerodynamic design point (ADP) is determined based on how much core pressure build up is necessary to reach the same design HPC *exit* corrected flow as the core defining engine (also at its ADP design point) while having increased the HPC *inlet* corrected flow based on the inputted amount of core flow scaling (sW_{25R}).

The surrogate models referred to as *Surrogate Set B* were trained to provide deterministic performance estimates of a common core variant design. By taking the time to consider the ranges of design and noise variable settings of interest before running simulations for surrogate training, it was ensured that the resultant surrogate models would be valid for

Table 19 Revisited: Control variables and corresponding ranges explored for the present common core variant engine design explorations.

Variable	Units	Min	Max
$\pi_{Fan,Var}$ @ ADP	-	1.3	1.7
$\pi_{Overall,Var}$ @ ADP	-	30.0	55.0
$T_{4,Var}$ @ TKO	$^{\circ}R$	3,300	3,750
ER_{Var} @ ADP	-	0.90	1.25
$NChg_{LPT,Var}$	% W_{25}	0.00	0.06
sW_{25R} @ ADP	-	1.0	1.2
$N_{C,HPT,Var}$ @ TKO	%	100.	105.

the present experiment. With these surrogate models, efficient exploration of the variant design spaces for each upgrade option can be performed. Many tens of thousands of candidate designs can be evaluated with function calls of the surrogate models within a few seconds.

For each upgrade option considered, 10,000 candidate designs were explored within the scope of the upgrade option and within the variant design variable ranges considered. The resultant design data is then filtered to ensure design feasibility and the feasible design that achieves maximum growth takeoff thrust for each upgrade option is selected.

4.3.7 Geometrically Fixed Core - Design Space Explorations for Various Technology Scenarios

The least complex common core variant design scenarios do not allow any scaling of the design HPC inlet corrected flow. This represents a geometrically fixed engine core. The design restriction will likely also restrict the amount of thrust growth achievable by the variant design selections within this category of scenarios. Three specific scenarios will be considered, each with increased amounts of technology infusion. The first scenario, Scenario 2.1, does not have any technology infusion beyond the core defining engine's technology level. Scenario 2.2 has technology infused into the LP engine components, improving the component efficiency and weight levels by the factors listed in Table 45. Scenario 2.3 considers the LP component technology infusion of Scenario 2.2 as well as core component advancements, which are listed in Table 46.

Figures 65 through 67 contain multivariate scatterplots showing the feasible design candidates considered in the design space explorations performed for each of these three geometrically fixed core scenarios. All of the scatterplots shown henceforth have the following constraints imposed, allowing for consistent comparison of the feasible design spaces for each:

- $T_{3,max} < 1,700 \text{ } ^\circ R$
- $T_{45,max} < 2,350 \text{ } ^\circ R$
- Fan Diameter $< 100 \text{ } in$
- Pod Weight $< 10,000 \text{ } lbm$

Comparison of the feasible design regions of common core variant design scenarios 2.1, 2.2, and 2.3, it can be seen that the infusion of technology does not visually change the feasible design space of the engine variant design significantly. This shows that the major contributor in the definition of the feasible space is the core design being restricted to its baseline definition. For each of the geometrically fixed core scenarios, candidate design selections were made that maximize the takeoff thrust, while simultaneously meeting the HPC exit temperature and LPT inlet temperature limits, and also having fan diameters less than 70 inches. This fan diameter limit was chosen to align with the fan sizes allowed in the following two experiments, and limits the fan to reasonable sizes typically allowed for this thrust class of engine.

Table 47 lists the feasible design samples for each scenario that achieve the maximum amount of takeoff thrust growth within a population of 10,000 random variant design samples. With no additional technology infusion, the Scenario 2.1 engine variant with a new LP system design selected to produce the maximum takeoff thrust within the temperature and fan diameter limits imposed is able to achieve 22.95% growth in takeoff thrust. The most thrust growth achievable after infusing LP system technology for Scenario 2.2 is 24.28% takeoff thrust growth. By bumping up the LP component efficiency levels, more work is able to be performed by a given core flow capacity.

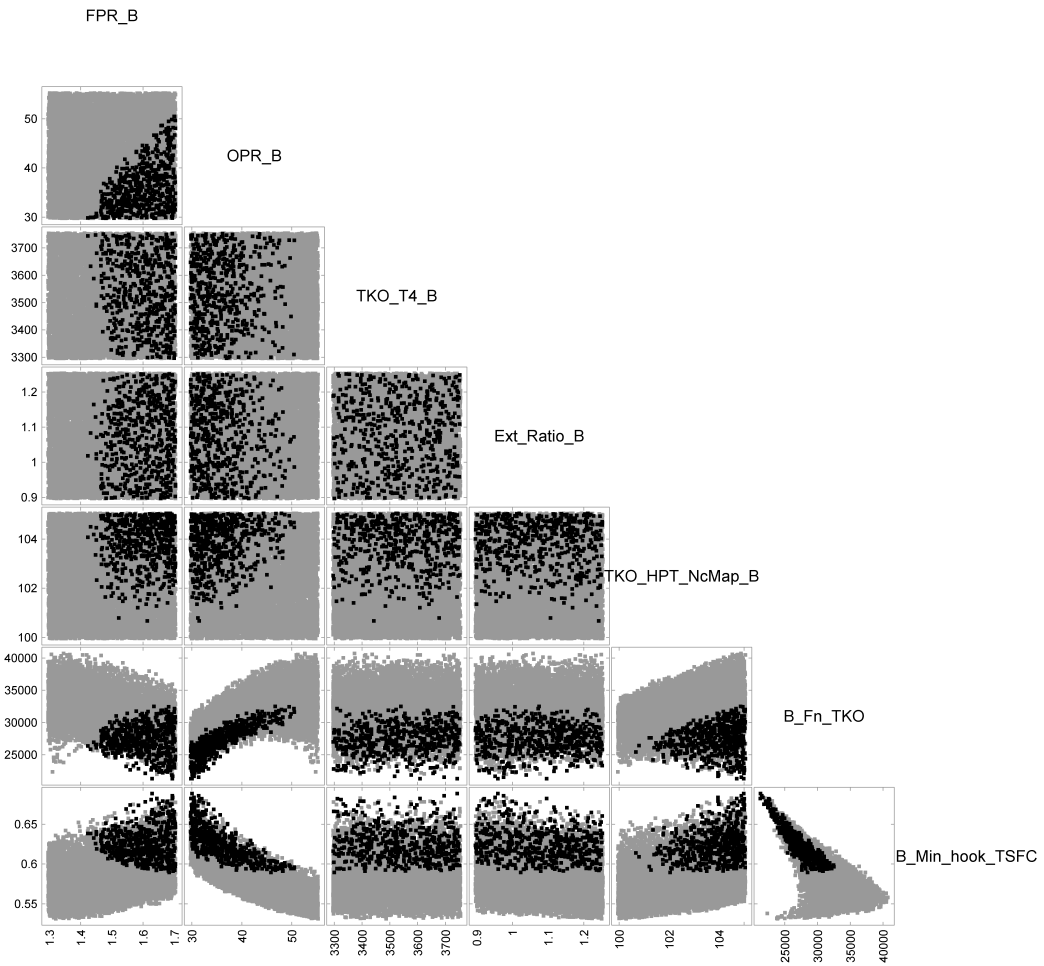


Figure 65: Scenario 2.1 multivariate scatterplot of feasible candidate variant engine designs.

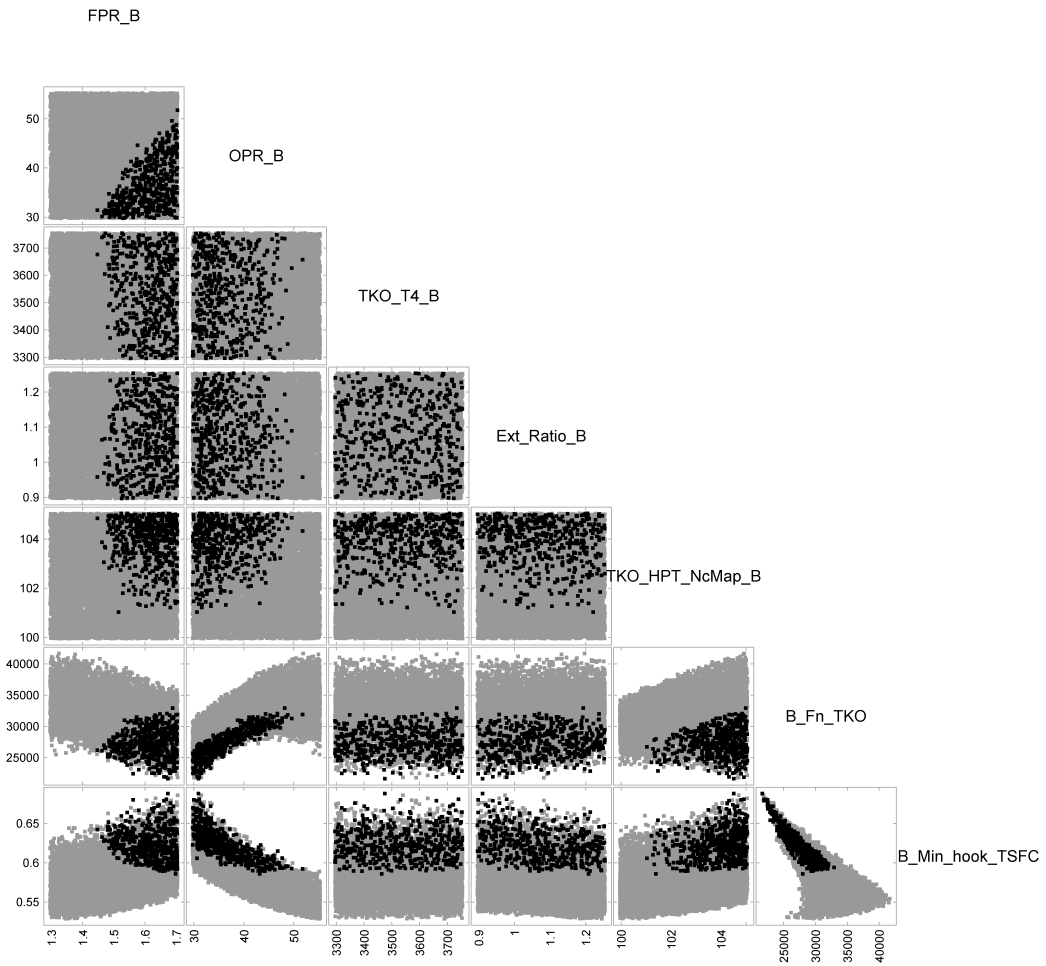


Figure 66: Scenario 2.2 multivariate scatterplot of feasible candidate variant engine designs.

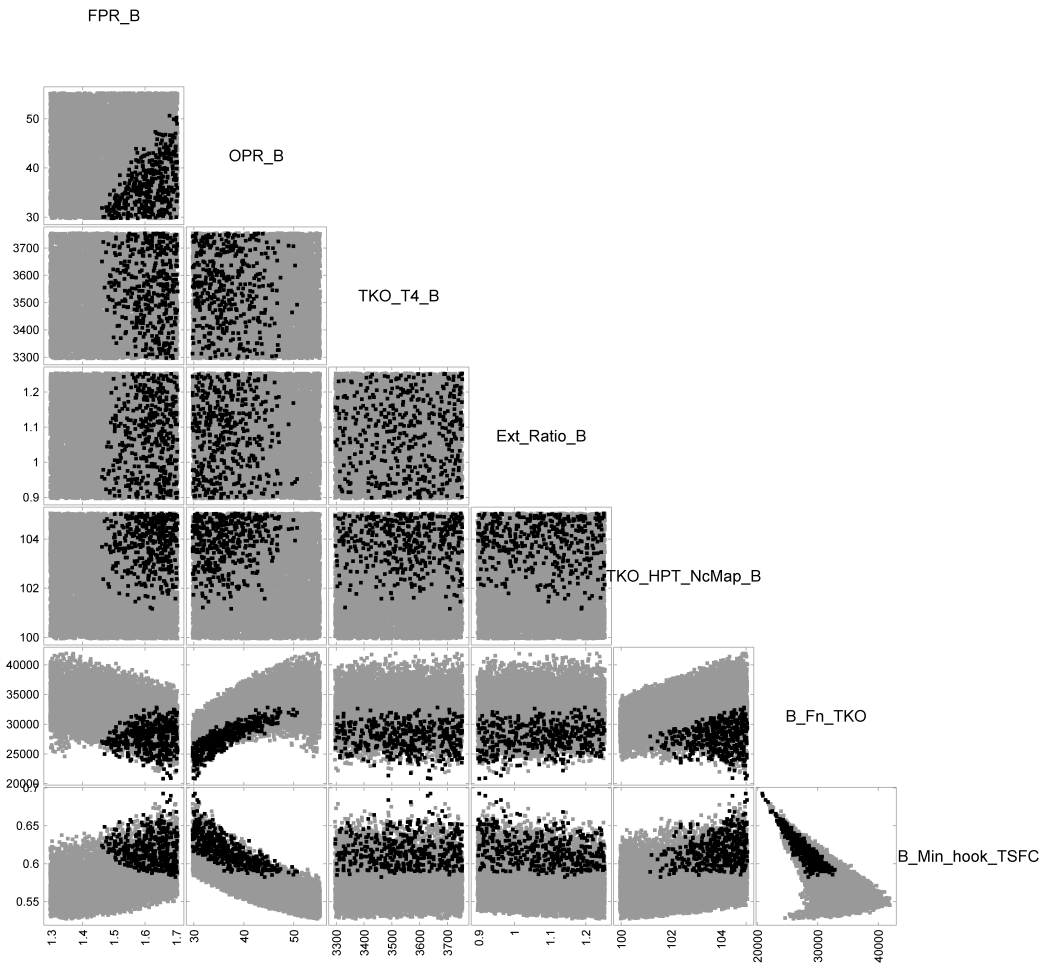


Figure 67: Scenario 2.3 multivariate scatterplot of feasible candidate variant engine designs.

Table 47: Experiment 2 - Geometrically Fixed Core Scenarios - Feasible design selections achieving maximum takeoff thrust growth.

Parameter	Technology Scenario		
	2.1	2.2	2.3
$\pi_{Fan,Var}$ at ADP	1.68	1.70	1.66
$\pi_{Overall,Var}$ at ADP	38.87	39.11	38.35
$T_{4,Var}$ at TKO, $^{\circ}R$	3,318.	3,640.	3,396.
ER_{Var} at ADP	0.90	0.98	0.94
sW_{25R} at ADP	1.00	1.00	1.00
$N_{C,HPT,Var}$ at TKO, %	104.1	104.7	105.0
Fan Diam, in	69.8	69.4	69.7
$T_{3,max}$, $^{\circ}R$	1,647.	1,678.	1,680.
$T_{45,max}$, $^{\circ}R$	2,382.	2,395.	2,394.
BPR at ADP	5.31	5.29	5.37
W_{25R} at ADP, lbm/s	49.6	49.6	49.6
W_{3R} at ADP, lbm/s	6.68	6.68	6.68
TKO Core Power Growth, %	25.52%	34.20%	35.32%
TKO F_N Growth, %	22.95%	24.28%	25.36%
TOC F_N Growth, %	29.89%	27.74%	23.86%
TSFC at CRZ, $lbm/(lbf \cdot hr)$	0.6384	0.6358	0.6344
Pod Weight, lbm	6,513.	5,986..	6,226.

Similar to what was shown with LP technology infusion, the infusion of both LP and HP technology to improve component efficiency and component weight for Scenario 2.3 offers 25.36% growth in takeoff thrust. Again, increases in component efficiency levels allow for more energy to be added to the core flow on the compression side, and similarly more energy is able to be extracted from the core flow on the expansion side. This results in additional power and thrust available for a fixed geometry core and corresponding flow capacity.

For the fixed geometry core engine variant scenarios, infusion of technology offers minor additions to the amount of thrust growth achievable. The redesign of the LP system allows for most of the thrust growth, but more efficient components are able to provide further increases in capabilities. The additional design freedom of the next upgrade scenarios will offer further increases in common core variant capabilities, but with accompanying increases in upgrade complexity and development cost.

4.3.8 Modified Common Core - Design Space Explorations for Various Technology Scenarios

When employing a fixed geometry core definition to a new LP system design in response to a new set of thrust and/or power requirements, the range of growth capabilities can be limited, as was shown in the previous section. This section considers a modification to the otherwise fixed engine core, allowing for the core inlet flow, or HPC inlet corrected flow (W_{25R}), to be increased by up to 20% when considering a new common core variant application. Core flow scaling can be accomplished through zero staging and/or flaring of the HPC inlet. The following two scenarios allow for core flow scaling up to 20% beyond the baseline engine design flow capacity. The common core variant design rules developed in Chapter 3 are used to impose the core design restrictions on the off-design core definition while allowing the core inlet flow to scale by the desired amount during exploration of the variant cycle design space. Scenario 2.4 attempts to achieve significant thrust growth while remaining at a consistent technology level with the baseline engine, while Scenario 2.5 provides the engine with advanced LP and HP components that improve their design efficiency while having weight reductions as well.

Figure 68 and Figure 69 show the feasible variant design regions for Scenarios 2.4 and 2.5. The scatterplots again have the following constraints imposed, allowing for consistent comparison of the feasible design spaces:

- $T_{3,max} < 1,700 \text{ } ^\circ R$
- $T_{45,max} < 2,350 \text{ } ^\circ R$
- Fan Diameter $< 100 \text{ } in$
- Pod Weight $< 10,000 \text{ } lbm$

Just as with the fixed geometry common core variant scenarios, it is hard to distinguish between the feasible design regions with and without technology infusion. Nevertheless, candidate selections were made for each scenario that were within the imposed limits, including having a fan diameter less than or equal to $70 \text{ } in$, and was the remaining feasible

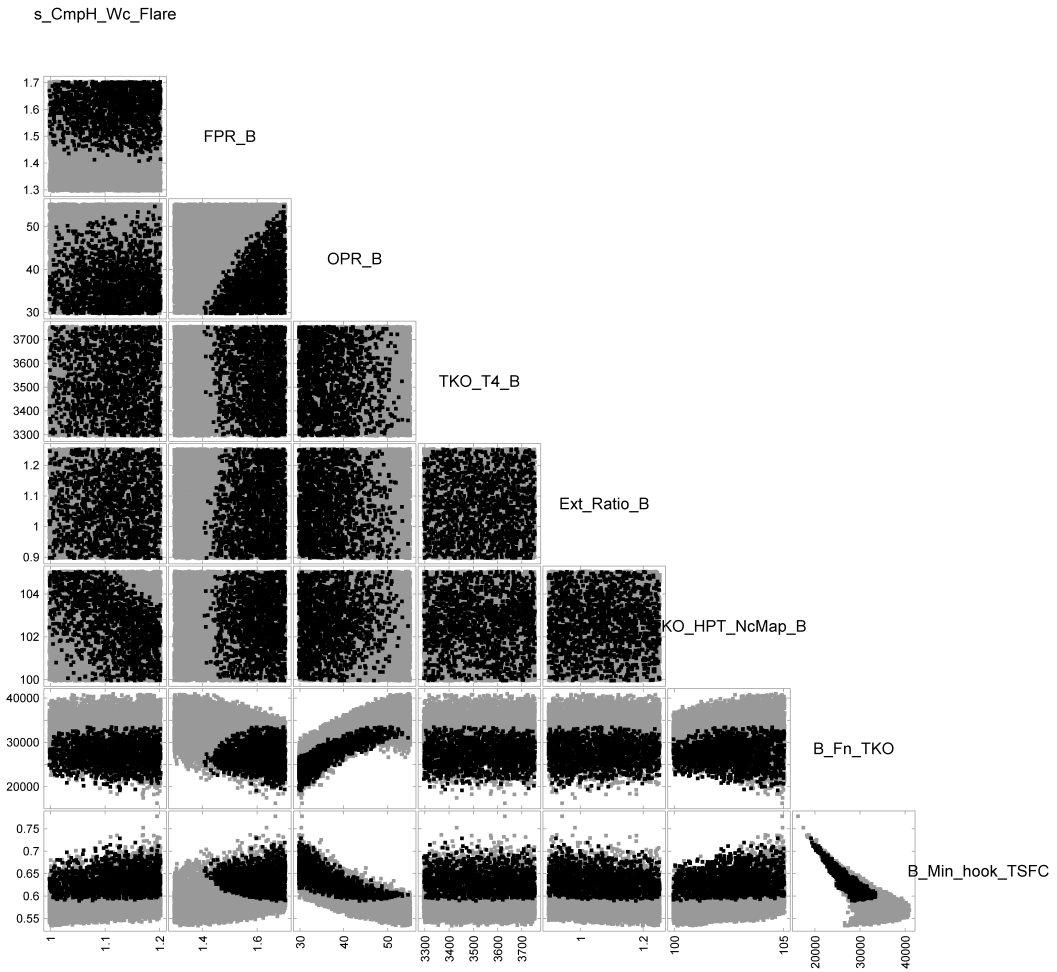


Figure 68: Scenario 2.4 multivariate scatterplot of feasible candidate variant engine designs.

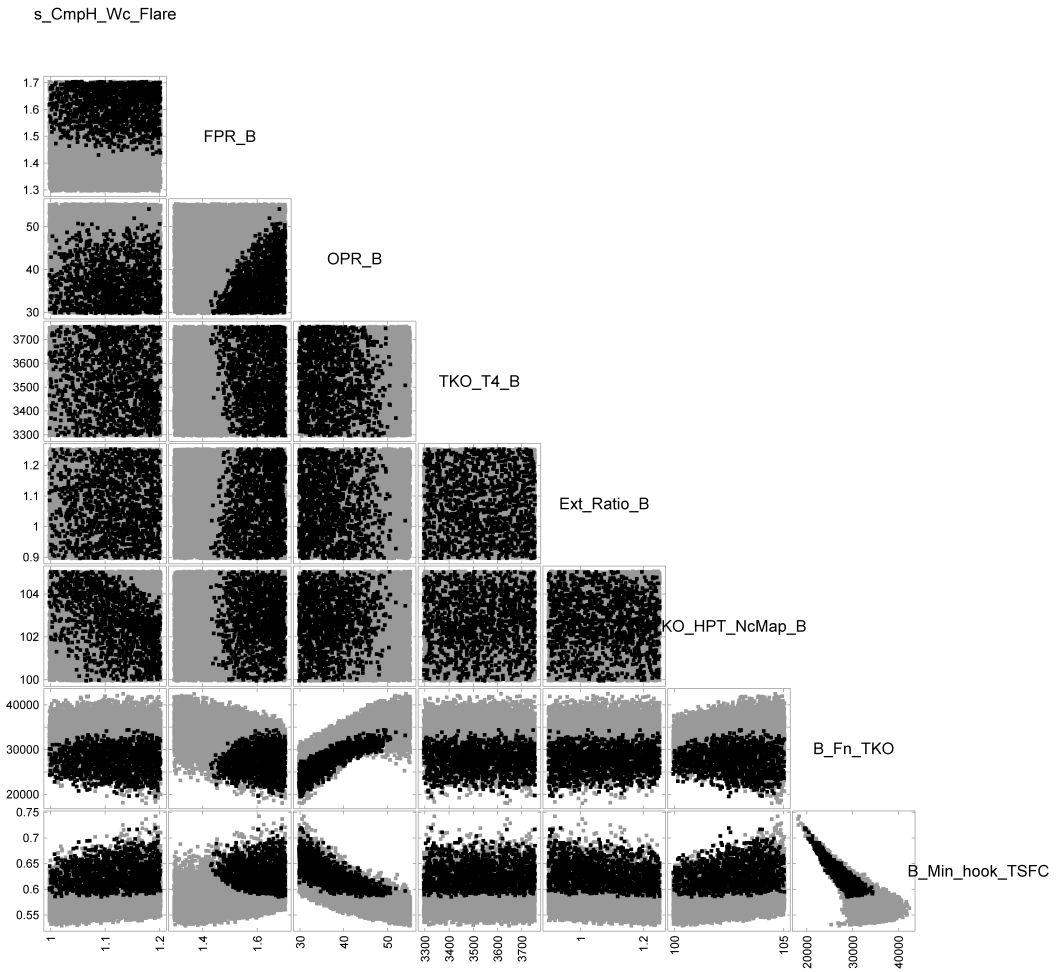


Figure 69: Scenario 2.5 multivariate scatterplot of feasible candidate variant engine designs.

Table 48: Experiment 2 - Modified Core Scenarios - Feasible design selections achieving maximum takeoff thrust growth.

Parameter	Technology Scenario	
	2.4	2.5
$\pi_{Fan,Var}$ at ADP	1.66	1.69
$\pi_{Overall,Var}$ at ADP	39.28	39.58
$T_{4,Var}$ at TKO, $^{\circ}R$	3,640.	3,655.
ER_{Var} at ADP	0.91	0.97
sW_{25R} at ADP	1.16	1.13
$N_{C,HPT,Var}$ at TKO, %	102.2	103.0
Fan Diam, <i>in</i>	69.1	69.6
$T_{3,max}$, $^{\circ}R$	1,680.	1,682.
$T_{45,max}$, $^{\circ}R$	2,350.	2,355.
BPR at ADP	5.22	5.25
W_{25R} at ADP, <i>lbm/s</i>	57.7	55.9
W_{3R} at ADP, <i>lbm/s</i>	6.68	6.68
TKO Core Power Growth, %	42.70%	41.64%
TKO F_N Growth, %	22.91%	24.10%
TOC F_N Growth, %	24.91%	27.41%
TSFC at CRZ, <i>lbm/(lbf · hr)</i>	0.6460	0.6371
Pod Weight, <i>lbm</i>	6,556.	6,287.

candidate sample that achieved the maximum takeoff thrust. These common core variant cycle selections can be found in Table 48 along with their corresponding performance characteristics.

Allowing the core inlet flow to scale provides more available core power growth than was attainable with a completely fixed geometry core design. However, limiting the maximum fan diameter to 70 inches and increasing the core flow capacity results in lower bypass ratio engine applications with lower thrust growth than the fixed geometry options. Scenario 2.4, without having additional technology infusion, was able to achieve 22.91% takeoff thrust growth while maintaining competitive cruise TSFC levels. This growth in thrust required the core inlet corrected flow to be increased by 16%, while also taking advantage of the freedom to over-speed the HPT at takeoff if necessary, over-speeding the core by 2.2% to achieve maximum thrust, while remaining within the imposed design limits. The maximum top of climb thrust grew by a similar amount. The takeoff core power necessary to produce

this 23% increase in takeoff thrust was almost 43% greater than the core power generated by the baseline engine definition at takeoff. The technology infused feasible candidate common core variant design considered in Scenario 2.5 that allowed for maximum takeoff thrust growth was able to reach 24.10% more thrust than the baseline engine. Through efficiency improvements of all turbomachinery components, increasing the core flow by 13%, and over-speeding the HPT at takeoff by 3%, the advanced technology common core design was able to generate slightly more core power while staying within the allotted fan diameter and temperature limits.

When applying a previously designed engine core definition to a new engine application and allowing the core flow to be scaled, the resultant engine design was able to produce up to 8% more power than was possible when utilizing the geometrically fixed core. Comparing the feasible common core variant design spaces with and without restricting the core inlet flow to its baseline design level, Figure 70 shows the feasible design region of the geometrically fixed, technology infused common core application in Scenario 2.3, and Figure 71 shows the feasible variant design region when allowing core flow increases in addition to technology infusion as was considered in Scenario 2.5. The main difference between the two highlights the impact of allowing increases in the core working fluid. Figure 70 shows that without core flow scaling, the HPT must operate at levels at or above 102% corrected speed, while Figure 71 shows that with core flow increases of at least 5% allow the HPT to operating at design takeoff speed of 100%. The figure also shows that significant flow scaling in addition to over-speeding of the core at takeoff prevents variant designs from meeting the limits imposed. Consideration of core flow scaling allows for the trade to be made between core flow increases and HPT over-speeding to achieve power and thrust growth.

The multivariate scatterplots offer the cycle designer a way to see certain slices of the feasible design space, showing performance trends with variant cycle changes. Parallel coordinate plots are often more helpful in visualizing feasible and high performance regions of a design space made up by several design dimensions, enabling the designer to see all cycle input settings of particularly high performing design candidates all at once. Figure 72 and Figure 73 display the same feasible variant design candidates as are shown in the

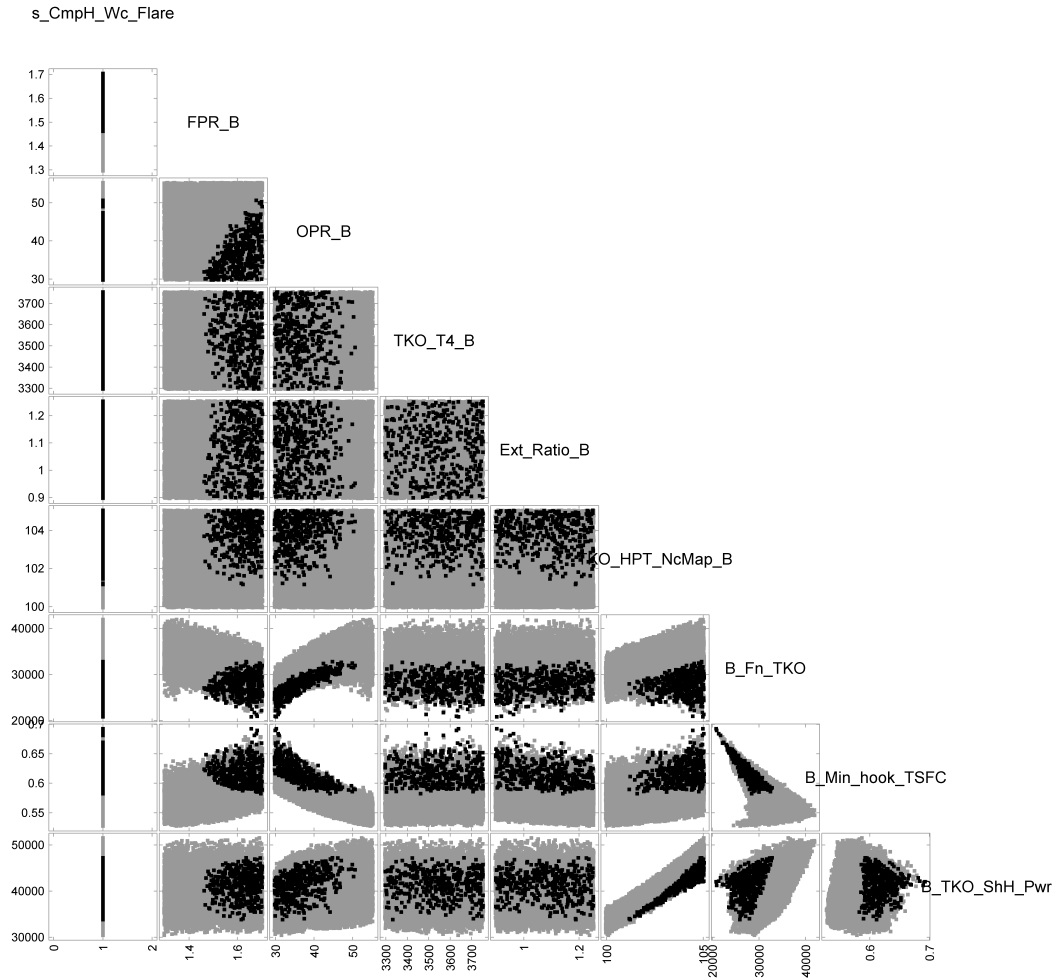


Figure 70: Scenario 2.3 multivariate scatterplot of feasible candidate variant engine designs. High technology variant engine design space with fixed geometry core.

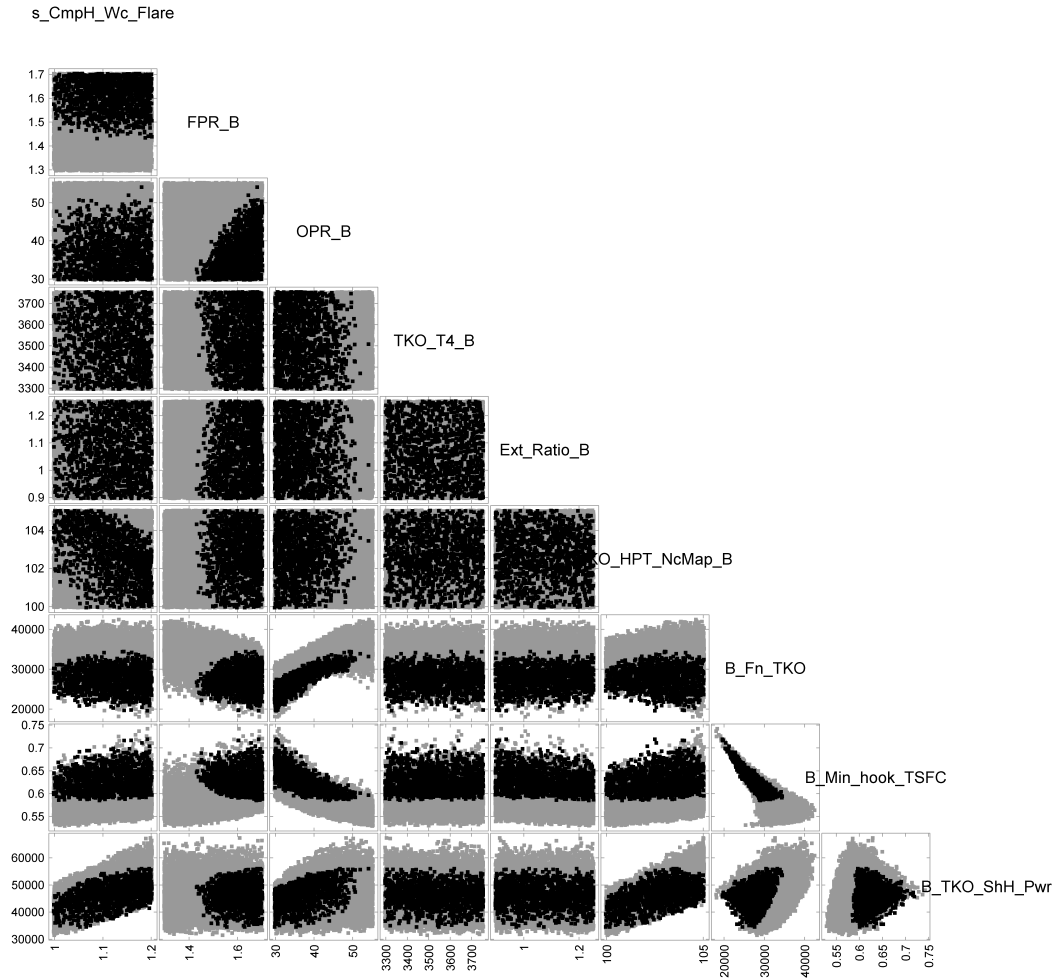


Figure 71: Scenario 2.5 multivariate scatterplot of feasible candidate variant engine designs. High technology variant engine design space with flared HPC core.

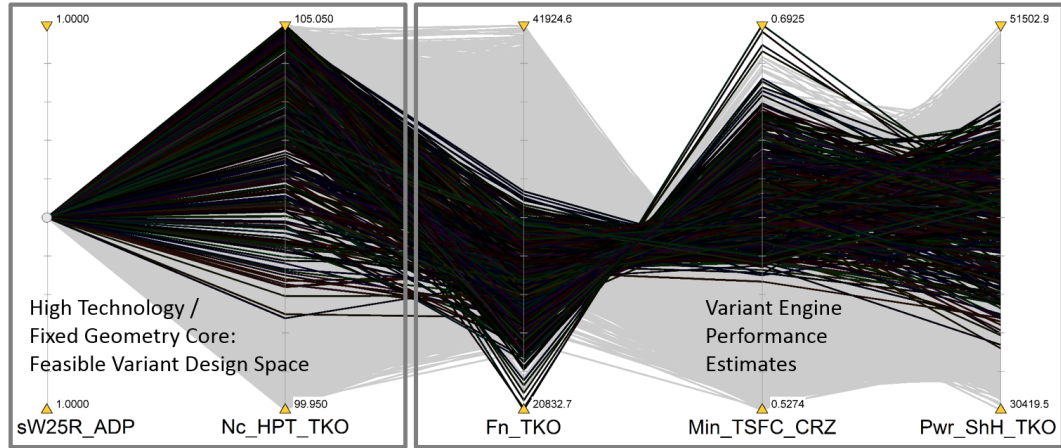


Figure 72: Scenario 2.3 parallel coordinate plot of feasible candidate variant engine designs. High technology variant engine design space with fixed geometry core.

multivariate scatterplots, but have them displayed in parallel coordinate plots.

After removing the 70 *in* fan diameter constraint, and filtering out the variant designs with compromised performance levels and only showing the best performing candidates of the Scenario 2.3 variant designs that produced at least 31,000 *lbf* of takeoff thrust (at least 36% growth when allowing fans larger than 70 *in*) while also achieving cruise TSFC levels below 0.61 *lbm/(lbf · hr)*, Figure 74 shows that significant over-speeding of the HPT at takeoff is necessary to provide significant growth in takeoff thrust to a point while also providing the best cruise TSFC levels. Figure 75 displays the common core variant design candidates for Scenario 2.5 that had the best performance levels, applying the same filters as were used in Figure 74 for Scenario 2.3. There are many more variant design possibilities that offer feasible, high performing solutions when allowing the core flow be to increased, providing more takeoff thrust with less increases in takeoff core power necessary to produce the increase levels of thrust.

Now also considering the remainder of the common core variant cycle design parameters, Figure 76 shows the high performing candidate cycle settings for Scenario 2.3, while Figure 77 displays the best performing cycle settings for Scenario 2.5. Both scenario explorations show that high fan pressure ratio variant designs are desirable, allowing for as much energy addition to the high bypass flow as possible. Medium to high OPR designs are also

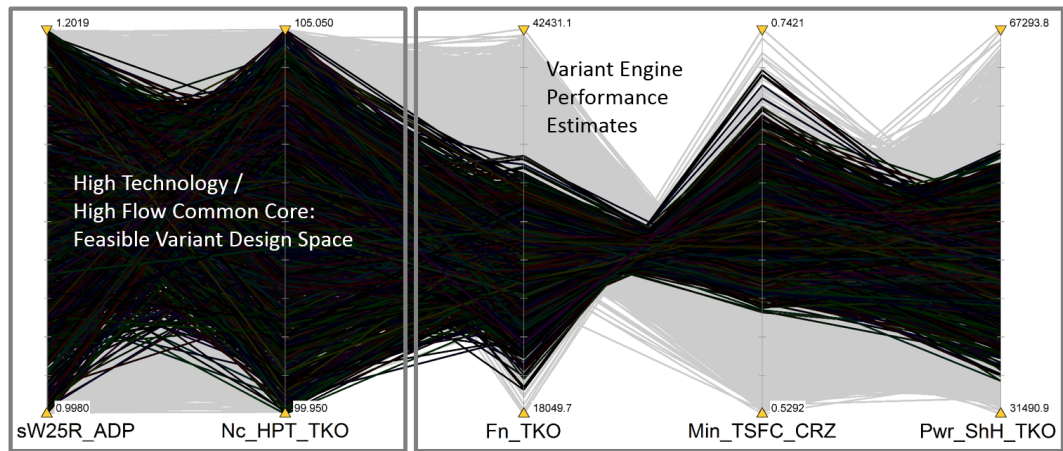


Figure 73: Scenario 2.5 parallel coordinate plot of feasible candidate variant engine designs. High technology variant engine design space with flared HPC core.

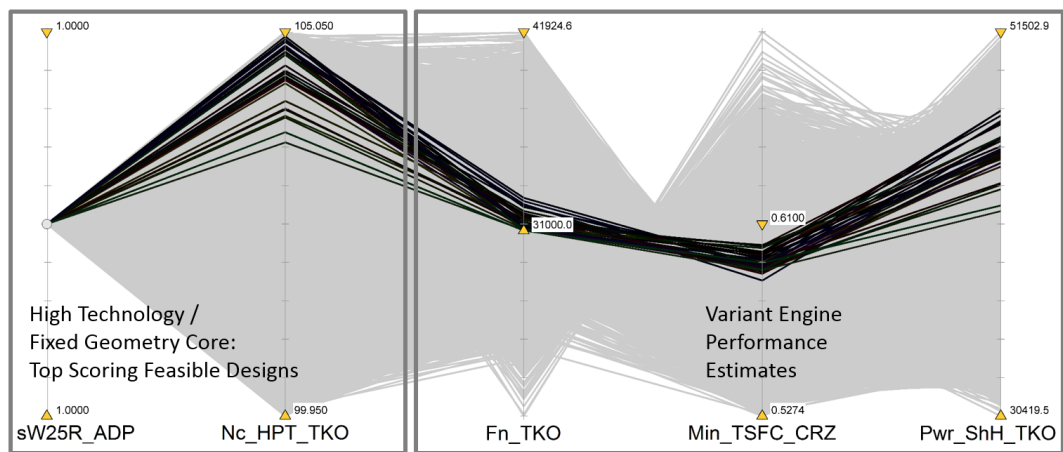


Figure 74: Scenario 2.3 parallel coordinate plot of best candidate variant engine designs. High technology variant engine design space with fixed geometry core.

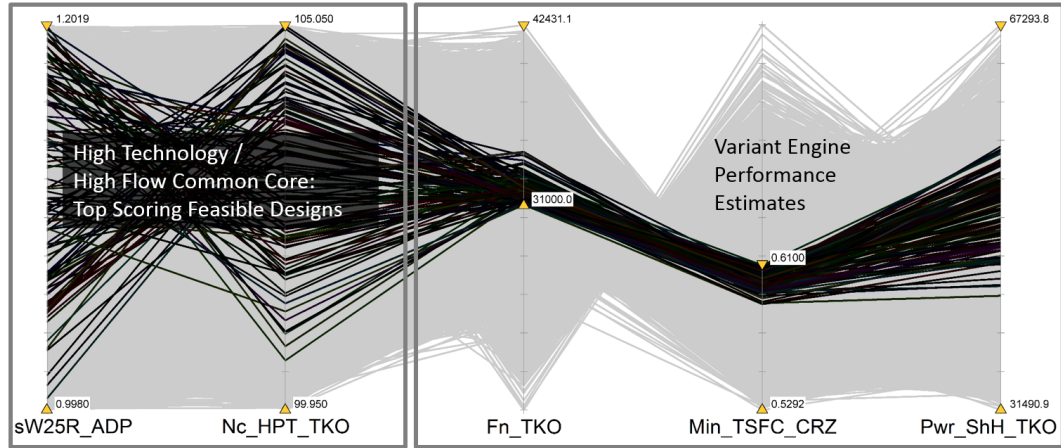


Figure 75: Scenario 2.5 parallel coordinate plot of best candidate variant engine designs. High technology variant engine design space with flared HPC core.

favorable, while the highest OPR designs start to violate the imposed $T_{3,max}$ constraint. High thrust growth can be achieved pretty much anywhere within the sampled ranges of takeoff $T_{4,Var}$ and variant design extraction ratio (ER_{Var}). As noted earlier, the key shift in favorable variant designs with and without core flow scaling is that by scaling the core flow, significant over-speeding of the HPT is no longer necessary to produce large growth levels of takeoff thrust. However, zero-staging the HPC and/or flaring the HPC inlet as well as all the necessary design changes that accompany the upgrade (such as possible re-blading of the HPC and/or HPT, HPT vane reset, new variable guide vane schedules, etc.) would likely require significantly more resources to develop and bring to market.

4.3.9 Conclusions

For the common core variant design scenarios where the core geometry was not allowed to be modified at all, significant thrust growth was achievable, with further increases achievable through the infusion of technology improving component efficiency. This highlights a key consideration that must be addressed when attempting to employ common core engine applications. Once a core definition is established and is to be applied to a new application, the designer has two choices. The core design can remain completely fixed, but will have a high likelihood of having significant performance compromises due to the fixed core being

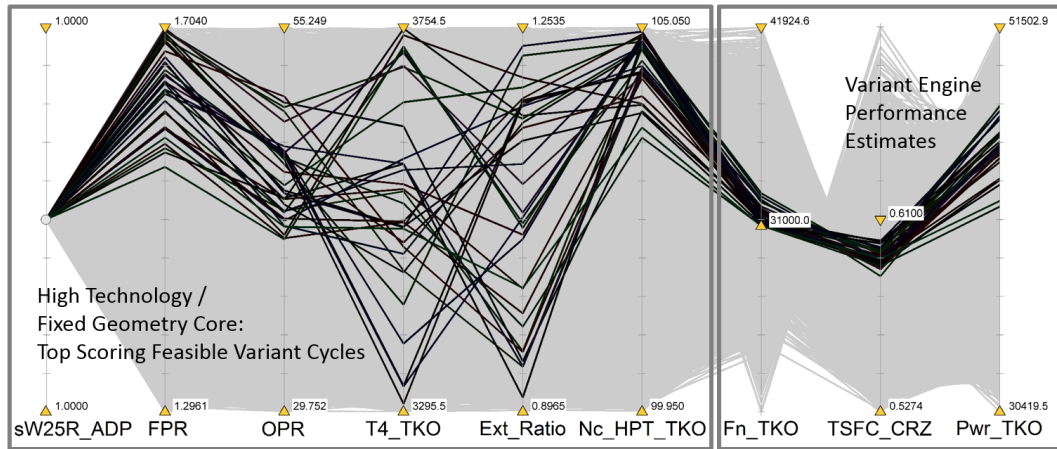


Figure 76: Scenario 2.3 parallel coordinate plot of best candidate variant engine cycles. High technology variant engine design space with fixed geometry core.

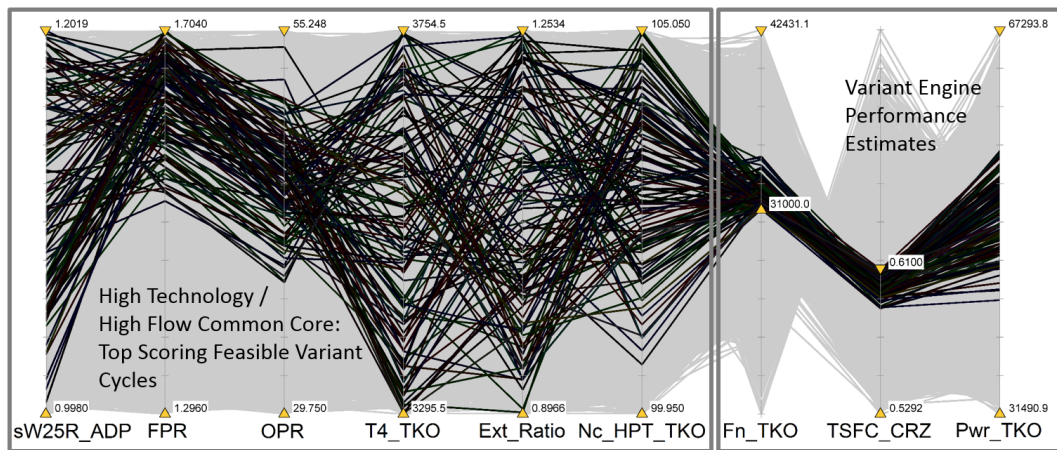


Figure 77: Scenario 2.5 parallel coordinate plot of best candidate variant engine cycles. High technology variant engine design space with flared HPC core.

a mismatch for the new set of customer requirements. Alternatively, the designer can make core design modifications, incurring significant development costs, but allowing for a more flexible range of capabilities. However, if there are a significant number of engine applications being considered, and many unique core design modifications are necessary to apply the available core definition across the program, development costs could potentially go through the roof.

This provides the rationale behind the development and use of the COMMENCE method. Consideration of possible common core applications are made upfront, early in the core development program. This allows for implications of various sets of requirements on the required core definition to be drawn before making core design down selections. Also, limitations of candidate core designs can be quantified early in the process, allowing for decision makers to decide whether the limitations are acceptable, or if actions must be taken to extend the engine program capabilities. In addition to the simultaneous consideration of possible applications, impacts of uncertainty present in the initial and projected applications can be accounted for upfront, allowing for required design margins to be determined quantitatively instead of subjectively in order to improve the likelihood of success throughout the family of engine applications. By allowing the designer to make more informed core design decisions, the likelihood of engine family success can be increased.

The following experiment integrates both the ERDS method demonstrated in Experiment 1 and the common core variant design logic established and demonstrated in this experiment into the overall COMMENCE method, and the first simultaneous multiple application common core design space exploration will be performed. Consideration of multiple sets of customer requirements will be made while accounting for significant technology and installation requirements uncertainty, leading to the exploration and selection of a common core definition and corresponding common core variant engine designs that achieve a high likelihood of meeting all their requirements and constraints while also achieving competitive performance levels.

4.4 *Experiment 3: Common Engine Core Design for Multiple Applications*

- **Primary Research Question:** *How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?*

Hypothesis: *Simultaneous simulation and evaluation of current and future common core applications should be performed when exploring the common core design space in order to quantitatively estimate the feasibility and likely performance levels of program applications due to changes in the common core definition. If this is possible and implemented, the likelihood of achieving feasible, competitive common core variant designs will be increased while minimizing the amount of mitigation actions required later in the program.*

- **Research Question 1:** *How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?*

Hypothesis 1: *Probabilistic performance levels of candidate cycle designs should be estimated with the use of surrogate models that predict likely performance under various inputted uncertainty scenarios for any desired confidence interval.*

The previous experiment explored the amount of thrust growth achievable through common core engine applications of increasing upgrade complexity. Different sets of engine variant design restrictions and technology infusion were considered, and the corresponding core power and thrust levels achievable with minimum SFC compromise were determined. The experiment demonstrated the capabilities enabled through the modeling and simulation environment constructed to explore common engine core design problems.

The present experiment takes the next step in exploring gas turbine engine family designs. The rules established to tie the design engine, the core definition, and the common core engine variant application together are now duplicated to simultaneously consider

multiple engine applications of the same common engine core design. These relationships are implicitly contained within the surrogate models representing the deterministic and probabilistic common core variant design space. When simultaneously considering multiple common core variant designs, the same common core representative surrogate models ($SSB_{Var,Det}$ and $SSD_{Var,Prob}$) are utilized for all variant designs. For each application, the same core defining design engine variable settings are used while unique variant design variable values are used to evaluate multiple applications of the same geometrically common core family of engines. The goal of this study is to test the capabilities of the COMMENCE method when applied to a multiple application exploration and common core family design selection. For multiple sets of application requirements, the method will be used to search for a feasible core design region that offers feasible solutions for all engine applications considered. If a feasible core design region is identified, selection of a family of designs will be made that offers competitive performance with respect to benchmark engine designs established for each application.

4.4.1 Utilization of Existing Surrogate Models

The probabilistic surrogate model sets trained at the beginning of this chapter will again be utilized in the present experiment, both for the design engine which establishes the common engine core definition ($SSC_{Des,Prob}$), as well as for the variant engine performance ($SSD_{Var,Prob}$) which applies the common engine core definition to a new low pressure system. When exploring and making single engine design selections to establish benchmark engines for each set of requirements, $SSC_{Des,Prob}$ will be utilized to perform the single application robust design explorations and resultant benchmark selections. No additional training of surrogate models is needed for the following experiments. Furthermore, no additional repetitions of design simulations are needed, only single model evaluations are necessary to predict confidence interval performance under a given uncertainty scenario. All variability due to the ranges of uncertainty distributions considered is already implicitly accounted for in the probabilistic surrogate models. This unique capability of the probabilistic surrogate models produced by the ERDS method enables robust design explorations

of the benchmark engines and common core applications to be performed similarly to how typical engine cycle explorations are performed. Once the surrogate representations are produced, probabilistic explorations now offer the designer the ability to account for and design under uncertainty without any additional computational effort.

4.4.2 Multi-Application Common Core Design - Problem Setup

Three discrete sets of engine requirements are established for each engine application considered in the present common core design study. For the present study, a single technology scenario is assumed for all applications. A single technology scenario applied to all engine applications is representative of a common core family of engine applications that are all designed to enter the market within the same time frame.

To establish the performance compromises made by employing a common core definition across applications, new centerline benchmark engine designs will be explored and selections will be made for each application using the ERDS process exactly as it was applied in Experiment 1. Exploration of the common core design space will then be performed and a common core design region offering maximum application feasibility will be identified. Finally, simultaneous exploration of the engine core and each engine variant design will be performed to arrive at common core applications that achieve minimal performance deviations from each corresponding benchmark design.

This experiment aims to demonstrate the ease of using the COMMENCE method to consider any number of common core engine applications when making common core design selections. The unique formulation of the probabilistic surrogate models produced by the ERDS process when integrated into the COMMENCE method enables design explorations to be performed in a similar fashion to how engine cycle designers currently perform deterministic single design explorations. Familiar cycle trends result from the enabled robust design explorations, but performance estimates now account for the assumed uncertainty present in the designs considered. Multiple engine applications can now be considered simultaneously, and robust design selections can be made knowing the implications the selections have on the resultant common core variant application designs. For the design problem

Table 49: Experiment 3 - Engine Requirements and cycle temperature limits for each of the three applications considered in the present common core engine design problem.

Engine Requirement	App 3.A	App 3.B	App 3.C
Takeoff Thrust, <i>lbf</i>	23,000	27,000	30,000
Top of Climb Thrust, <i>lbf</i>	6,000	6,000	8,000
Horsepower Extraction, <i>shp</i>	250	250	250
Customer Bleed, <i>lbm/s</i>	2.35	2.35	2.35
Maximum HPC Exit Temperature, $^{\circ}R$	1,700	1,700	1,700
Maximum LPT Inlet Temperature, $^{\circ}R$	2,300	2,300	2,300
Cruise TSFC, <i>lbm/lbf/hr</i>	minimize	minimize	minimize
Pod Weight, <i>lbm</i>	minimize	minimize	minimize

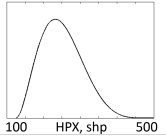
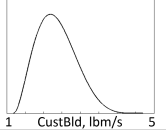
considered in this experiment, examination of the resultant common core selection will be performed, comparing the core definition to the core designs independently selected for each individual application considered.

4.4.2.1 Requirement Sets for Multiple Applications Considered

A common engine core and corresponding variant designs will be selected for three sets of requirements, listed in Table 49. Among the applications, there are three unique levels of takeoff thrust required. The low and medium takeoff thrust applications require the same thrust at top-of-climb conditions. The highest takeoff thrust application requires 33% more thrust at top-of-climb conditions. All three applications have the same constant horsepower and air bleed extraction installation requirements and associated uncertainty in the actual amounts required. The uncertainty distributions of these installation requirements are found in Table 50.

The three engine applications considered represent what would be required from similar aircraft variants, each of which having unique requirements for various specific design missions. Between the lowest and highest takeoff thrust levels required, the family of common core applications requires more than 30% growth in achievable thrust.

Table 50: Experiment 3 - Power and air extraction requirements uncertainty distributions assumed for the present multiple application experiment.

Impact	Min, Max, Alpha, Beta	PDF
<i>HPX, shp</i>	125, 500, 3.0, 6.0	
<i>CustBld, lbm/s</i>	1.18, 4.70, 3.0, 6.0	

4.4.2.2 Technology Scenario

This multiple application design problem assumes that all three engine applications are to enter the market within the same time frame. Therefore the technology scenario, containing the uncertain impacts of technology improvements is assumed to be the same for all three common core engine applications. The technology impact distributions of the assumed scenario are all contained in Table 51. For each category of technology, the minimum and maximum impacts, and the alpha and beta shape parameters of the assumed beta distributions are listed. This scenario is similar to the one assumed in Experiment 1.

4.4.3 Common Core Engine Design for Multiple Applications

This experiment is the first to carry out the entire COMMENCE process, originally shown in Figure 21 and redisplayed for ease of access. The sets of requirements for the discrete variant applications have been established. Robust design selections of benchmark (new centerline) engines will be established using the ERDS process under the same uncertainty scenario. This allows for determination of performance deviations purely due to the use of a geometrically common core design for all applications.

Next, the multiple application common core design space will be explored in order to identify any feasible regions of the common core design space. Simultaneous exploration of the core design definition as well as each of the three engine variant designs will be

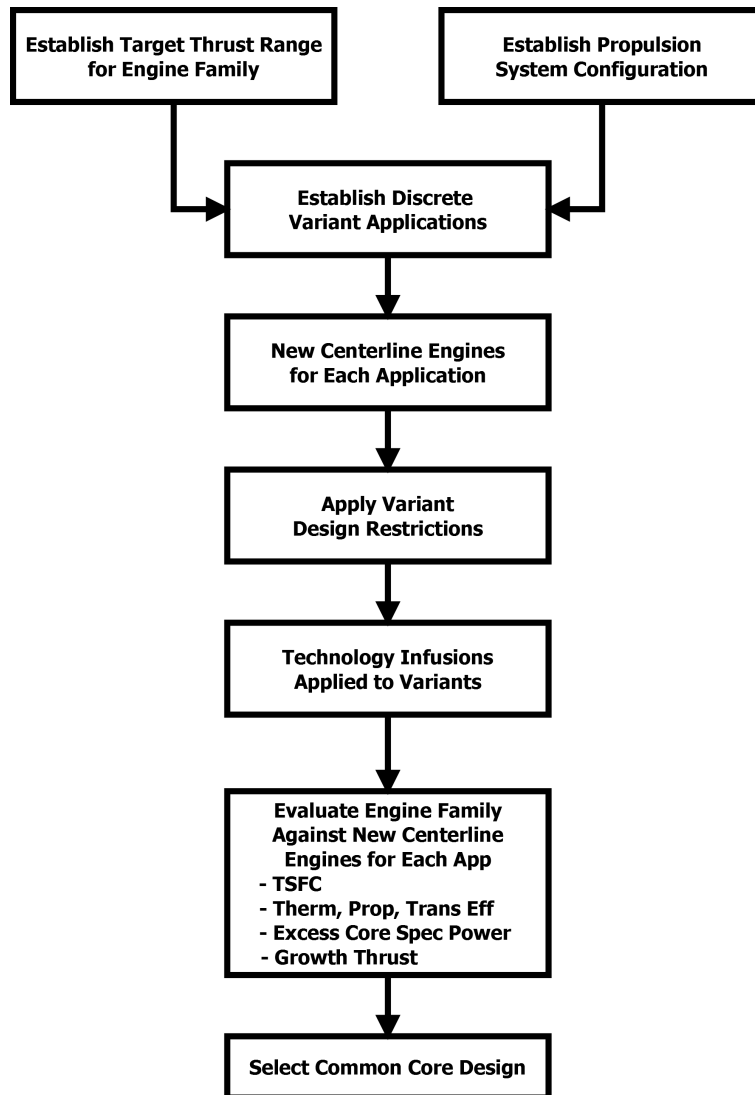
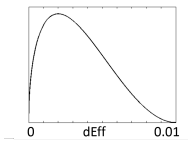
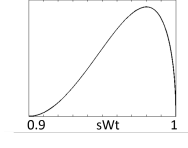
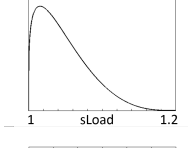
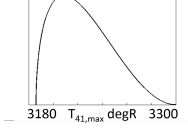


Figure 21 Revised: Common Engine Core Evaluation (COMMENCE) Method.

Table 51: Experiment 3 - Technology impact uncertainty distributions assumed for the present technology scenario.

Impact	Min, Max, Alpha, Beta	PDF
$\Delta\eta$	0.00, 0.01, 1.5, 3.0	
sWt	0.90, 1.00, 3.0, 1.5	
$sLoad$	1.00, 1.20, 1.25, 4.00	
$T_{41,max}, degR$	3,186, 3,300, 1.5, 3.0	

performed in order to maintain feasibility while also improving variant performance as much as possible. Final evaluation of the candidate common core engine family design will be performed, comparing each common core application’s performance to their corresponding benchmark engines. If the performance levels are deemed acceptable, then the design core definition is selected for multiple engine applications.

4.4.3.1 Benchmark Engine Selections for Each Application

Before performing any common core design space explorations, single application robust design selections are made for each of the sets of requirements established for this experiment. The ERDS method is used identically to how it was used in Experiment 1, utilizing the surrogate set $SSC_{Des,Prob}$ to perform design explorations and make selections. The same sources and distributions of uncertainty assumed for the common core design problem are assumed in the simulation and selection of each single application benchmark design. Exploration throughout the ranges of design variable settings listed in Table 18 is performed for

Table 18 Revisited: Clean sheet engine control variables and corresponding ranges explored during benchmark design space explorations. For consistency, identical ranges are used to explore the core defining engine design space.

Variable	Units	Min	Max
π_{Fan} @ ADP	-	1.3	1.7
π_{HPC} @ ADP	-	9.0	16.0
$\pi_{Overall}$ @ ADP	-	30.0	55.0
T_4 @ TKO	$^{\circ}R$	3,300	3,750
ER @ ADP	-	0.90	1.25
$NChg_{LPT}$	% W_{25}	0.00	0.06
F_N @ TKO	<i>lbf</i>	20,000	30,000
F_N @ TOC	<i>lbf</i>	5,900	8,850

each application’s benchmark engine design, and selections are made that meet all requirements and constraints imposed (based on 95% confidence interval performance estimates) while achieving competitive performance in terms of cruise TSFC and estimated engine pod weight.

Each new centerline engine selection offers the designer valuable information for the present design problem. The benchmarks established the performance levels achievable at the current technology level if design selections were made independently for each set of requirements. These benchmark engines represent designs located on the TSFC-Weight pareto frontier for each engine application. By comparing common core variant designs against these benchmark engines, quantification of the performance compromises incurred by employing common core variants can be determined, allowing the designer to decide whether each application is worth pursuing with the common core program.

The benchmark engine cycle selections made for each of the three applications considered in this experiment are found in Table 52 along with their corresponding performance estimates. For this and for all future design explorations and selections, 95% confidence interval performance estimates will be used in the evaluation of designs. This requires all designs considered to meet all requirements and achieve competitive performance levels with a an associated 95% probability of doing so. For performance levels whose direction of improvement is a reduction in the value, such as temperature levels, TSFC levels, and weight

Table 52: Experiment 3 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for each of the three applications considered.

Design Cycle Parameter	Bench 3.A	Bench 3.B	Bench 3.C
π_{Fan} at ADP	1.57	1.47	1.67
π_{HPC} at ADP	15.57	15.87	14.16
$\pi_{Overall}$ at ADP	37.49	32.17	33.56
T_4 at TKO, °R	3,319.	3,321.	3,334.
ER at ADP	0.90	1.00	0.90
$NChg_{LPT}$, % W_{25}	0.12%	0.00%	4.69%
95% CI Metric Value	Bench 3.A	Bench 3.B	Bench 3.C
F_N at TKO, <i>lbf</i>	23,220	26,970	30,230
F_N at TOC, <i>lbf</i>	6,021	6,042	7,940
$T_{3,max}$, °R	1,691.	1,682.	1,687.
$T_{45,max}$, °R	2,300.	2,300.	2,300.
W_{3R} at ADP, <i>lbm/s</i>	5.91	7.09	8.74
TSFC at CRZ, <i>lbm/lbf/hr</i>	0.6525	0.6440	0.6525
Pod Weight, <i>lbm</i>	3,685	3,633	6,458
Fan Diameter, <i>in</i>	67.0	73.8	70.5

estimates, the 95% confidence interval estimate will have an associated 95% probability of achieving the value listed or at a lower, more favorable level. For performance estimates whose direction of improvement is an increase in the value, such as thrust capabilities, the 95% confidence interval estimate will have a 95% probability of being at the listed value or at a higher, more favorable level.

The existing probabilistic surrogate models of *Surrogate Set C* ($SSC_{Des,Prob}$) and *Surrogate Set D* ($SSD_{Var,Prob}$) are utilized in this and the following multiple application experiment. For these surrogate models, the desired confidence interval of performance estimates is a user input, allowing the designer to predict and provide estimates anywhere on the cumulative distribution function of likely performance levels for a particular design if he or she desires to do so. For simplicity and for consistent comparison between designs in this experiment and the next, only 95% confidence interval estimates will be provided, providing the high likelihood of feasibility and performance typically desired when making robust design selections.

The characteristics shown in Table 52 indicate that all three benchmark engine designs

favor relatively low pressure levels, shown by the design $\pi_{Overall}$ levels selected. Additionally, all benchmark selections favor design π_{HPC} levels close to the maximum value allowed. These design pressure ratio selections allow for minimal levels of nonchargeable LPT cooling (as shown by the low levels of $NChg_{LPT}$) while allowing the cycles to remain under the LPT inlet temperature limit ($T_{45,max}$) of $2,300^{\circ}R$. By having relatively low OPR levels, the cooling air available is at relatively low temperature, increasing the cooling effectiveness and reducing the secondary flow required to bring the LPT inlet temperature to within the imposed limit. The relatively high HPCPR levels favor the HP system in the pressure ratio split, the difference in the fraction of the overall pressurization required of the low pressure and high pressure compression systems. Having the HPC provide most of the pressurization requires significant mechanical work to be supplied by the HPT, requiring more flow expansion through the HPT. This reduces the temperature of the gas exiting the HPT, reducing the amount of cooling flow required to bring the LPT inlet flow temperature within the imposed material limit. Also accompanying the low $\pi_{Overall}$ levels of the selections are relatively low design values of takeoff combustor exit temperature (T_4 at TKO). Just as with the LPT cooling, these low T_4 designs require less HPT nonchargeable cooling flow to bring the HPT inlet temperature (T_{41}) within the allowed levels present in the uncertainty distribution defined in Table 51.

These low temperature cycles make the benchmark designs less sensitive to the uncertainty present in the allowed $T_{41,max}$ and the variation in LPT inlet temperature due to other sources of uncertainty assumed present. By remaining insensitive to these sources of uncertainty and corresponding variability in design performance, the resultant engine designs can remain relatively small by minimizing secondary flows required to operate within temperature limits while also being able to provide the thrust levels required.

In order to compare the engine core definitions of each of the benchmark engine designs selected to each other and ultimately to the resultant common engine core selection made for all three applications, the HPC exit corrected flow is also listed in Table 52. As is done throughout the gas turbine industry, the HPC exit corrected flow (W_{3R}) is a geometric similarity parameter used for comparison between different engine core sizes[61]. Maintaining

consistent flow areas and HPC exit corrected flow at consistent map operating conditions ensures consistent component blade heights. As explained in Chapter 3, this approach is used to enforce geometric commonality. Core flow areas are held to design levels by not allowing core components to be resized during the variant engine simulation. As was verified in the previous experiment, the design HPC exit corrected flow (W_{3R} at ADP) is maintained between the core defining design engine and all common core variant cycles. As is expected, as the required thrust levels increase from Application A to Application C, the design levels of HPC exit flow also increase. This highlights the relationship between thrust requirements and core power requirements. The engine core definition grows as more thrust is required, increasing the available core thermal energy by increasing core flow. When considering a common engine core, it will be shown where the common core design HPC exit corrected flow level lies with respect to the benchmark design levels. Implications of using a common core size for multiple applications will also be identified, comparing desirable variant cycles and corresponding performance to the benchmark design selections.

With the establishment of benchmark new centerline engines designs for each unique set of application requirements, the designer can now consider the feasibility and performance implications of a common engine core design for the set of applications considered. The following section will set the uncertainty distribution shape parameters to reflect the assumed uncertainty scenario, and exploration of the common core design space will be performed for the multiple application design problem.

4.4.3.2 Multiple Application Common Core Design Space Exploration

With benchmark engines cycles established for each of the three applications considered in this experiment, the designer can now explore the common core design space. Figure 78 shows the functional dependencies present and accounted for in the probabilistic surrogate models representing the common core variant design ($SSD_{Var,Prob}$), when extended to the current consideration of three variant applications. As shown in the figure in the upper left hand corner, the design engine is first simulated and its corresponding probabilistic surrogate models are evaluated based on the set of design engine control variable (CV) and noise

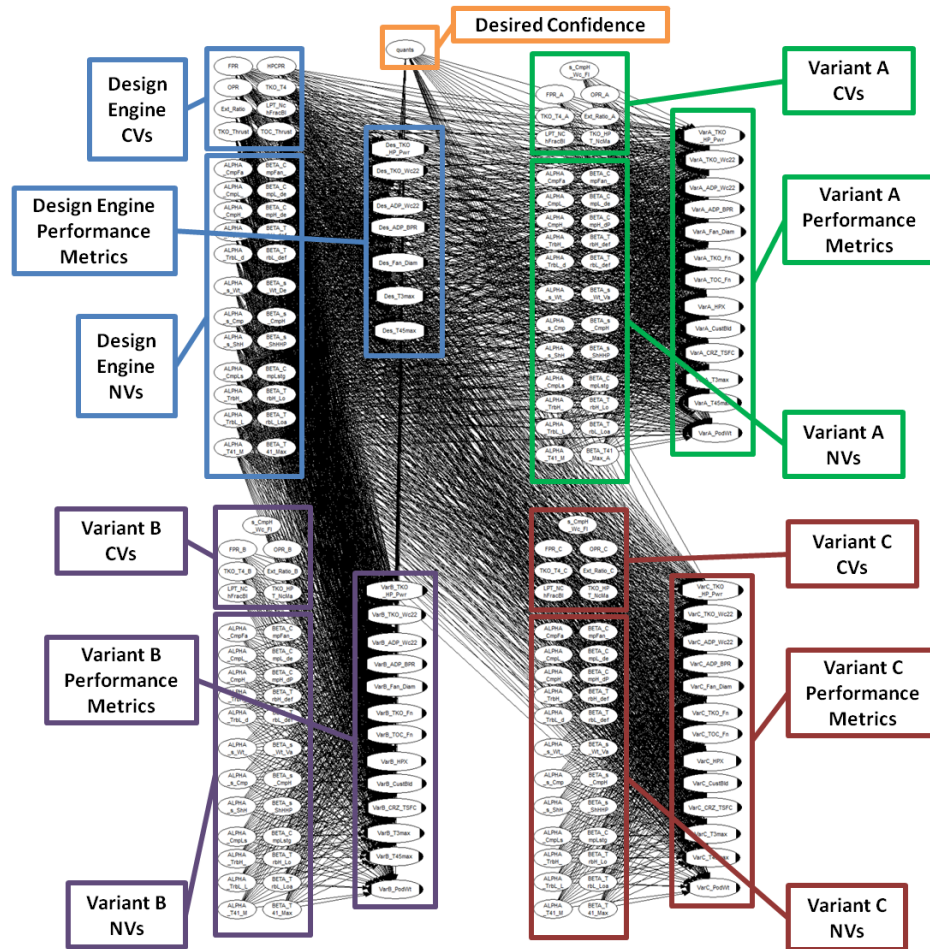


Figure 78: Representation of multiple application common core design problem considered in Experiment 3. Each engine variant performance estimate is dependent upon that variant’s input variable settings as well as the design engine input settings which establish the common engine core definition.

variable (NV) settings. The core definition is defined from the design engine simulation and is applied to each of the common core engine variants applications considered. This explains the functional relationships shown between the design engine input variables and engine variant performance metrics. This causes the probabilistic surrogate models predicting variant performance to be functions of the variant input variables as well as of the design engine input variables.

A simplified view of the functional relationships between the core defining design engine and each of the common core variant designs considered in this experiment is shown in

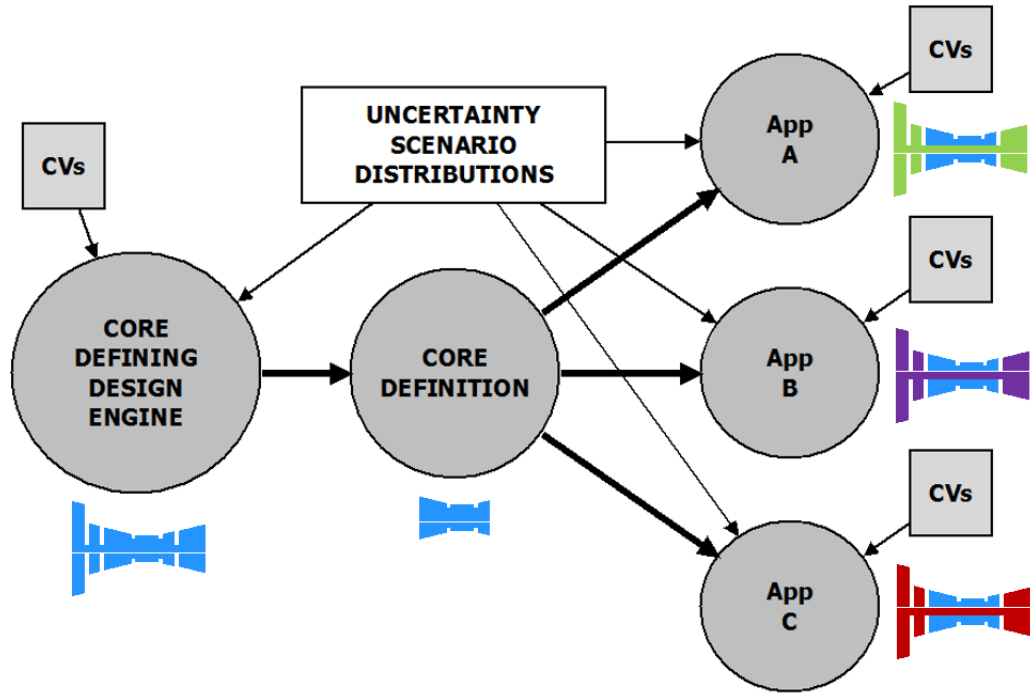


Figure 79: Simplified representation of multiple application common core design problem considered in Experiment 3.

Figure 79. For the core defining design engine and each of the three common core variants considered, the figure also contains cartoon representations of each engine’s flowpath, showing the blue common core being present in each of the variant applications. These cartoon representations will also be displayed above each application’s corresponding performance levels when all applications’ performance are simultaneously displayed. This highlights the technique developed to ensure commonality across all common core applications considered as one of the major contributions of this work.

For each of the three common core applications, engine core design restrictions are applied, tying the variant core definitions to the same design engine core. Scaling of the design HPC inlet corrected flow (sW_{25R} at ADP) and over-speeding the HPT at takeoff ($N_{C,HPT,Var}$ at TKO) are allowed in addition to any technology infusion of the engine components. This experiment assumes that the technology level of all three variants are equivalent and equal to the design engine technology level, so no additional technology

infusion is allowed.

The four-engine probabilistic design space (one design engine defining the core and three variants) can now be explored simultaneously. Ranges are applied to all of the control variables and samples of the design space are simulated. The control variable ranges sampled for the core defining design engine are identical to those sampled during the benchmark design explorations, which are listed in Table 18. The control variable ranges sampled for each of the three variant applications are listed in Table 19. A given sample contains a unique set of input variable settings for all four engine designs. These samples allow the designer to determine the feasibility of a common core definition offering feasible solutions for all of the applications considered.

It should be noted that as the number of engine applications considered increases, the probability of a sample (input variable settings for all designs considered) offering feasible common core designs for all applications considered decays exponentially. Therefore, a large number of samples are necessary in order to arrive at a significant number of feasible samples. Even more samples would be necessary to arrive at a multi-application design achieving competitive performance across all engine applications. For sample-based robust design techniques, a technique would need to be developed to overcome this challenge. Otherwise, the method would not be able to efficiently arrive at a good multiple application common core family design. This curse of dimensions would be compounded if repeated simulations of candidate designs were still necessary to account for the uncertainty present in the designs. The developments made by the author in the formulation of probabilistic surrogate models produced by the ERDS method overcome this curse of high dimensionality. By employing the continuous probabilistic surrogate models previously trained ($SSC_{Des,Prob}$ and $SSD_{Var,Prob}$) to provide confidence interval performance estimates with a single function call, the computational budget can be better used to more richly explore the common core design space.

After sampling the multiple application common core design space, limits are imposed, eliminating samples that do not meet the requirements or constraints present in the design problem. Figure 80 displays where feasible samples lie within the core design space. The

Table 19 Revisited: Control variables and corresponding ranges explored for the present common core variant engine design explorations.

Variable	Units	Min	Max
$\pi_{Fan,Var}$ @ ADP	-	1.3	1.7
$\pi_{Overall,Var}$ @ ADP	-	30.0	55.0
$T_{4,Var}$ @ TKO	$^{\circ}R$	3,300	3,750
ER_{Var} @ ADP	-	0.90	1.25
$NChg_{LPT,Var}$	% W_{25}	0.00	0.06
sW_{25R} @ ADP	-	1.0	1.2
$N_{C,HPT,Var}$ @ TKO	%	100.	105.

feasible samples, represented by the black lines in the parallel coordinate chart, meet all temperature limits imposed on all three applications, and meet the thrust levels required of each of the three applications. By analyzing the feasible design placement within the core defining design space in the left block of the chart, the designer is able to quickly identify favorable core design regions. For example, the common core applications obviously favor high design HPC pressure ratio core designs. Additionally, low design OPR cores are also favored. A high HPCPR, low OPR core definitions have high flow capacity offering significant growth potential while also having the ability to generate high core power. This is similar to what was shown during the benchmark engine design explorations. Low pressure and corresponding low temperature designs are favored, minimizing the required secondary cooling flows required to keep the engine operating within the imposed temperature limits, particularly when significant uncertainty is assumed present. The high HPCPR designs load up the design core, increasing the power generating capabilities when the core is applied to later variant applications. Also of note, the design engine thrust levels that provide feasible variants are similar to the thrust levels required of the common core applications. Design engines that offer feasible common core applications have design takeoff thrust levels above 26,000 *lbf* and design top-of-climb thrust levels above 6,700 *lbf*.

In addition to looking purely at the thrust capabilities of the common core engine variants, Figure 81 shows the cruise TSFC levels achievable by each of the feasible common core design samples. It is shown in the plot that competitive cruise TSFC levels are achievable,

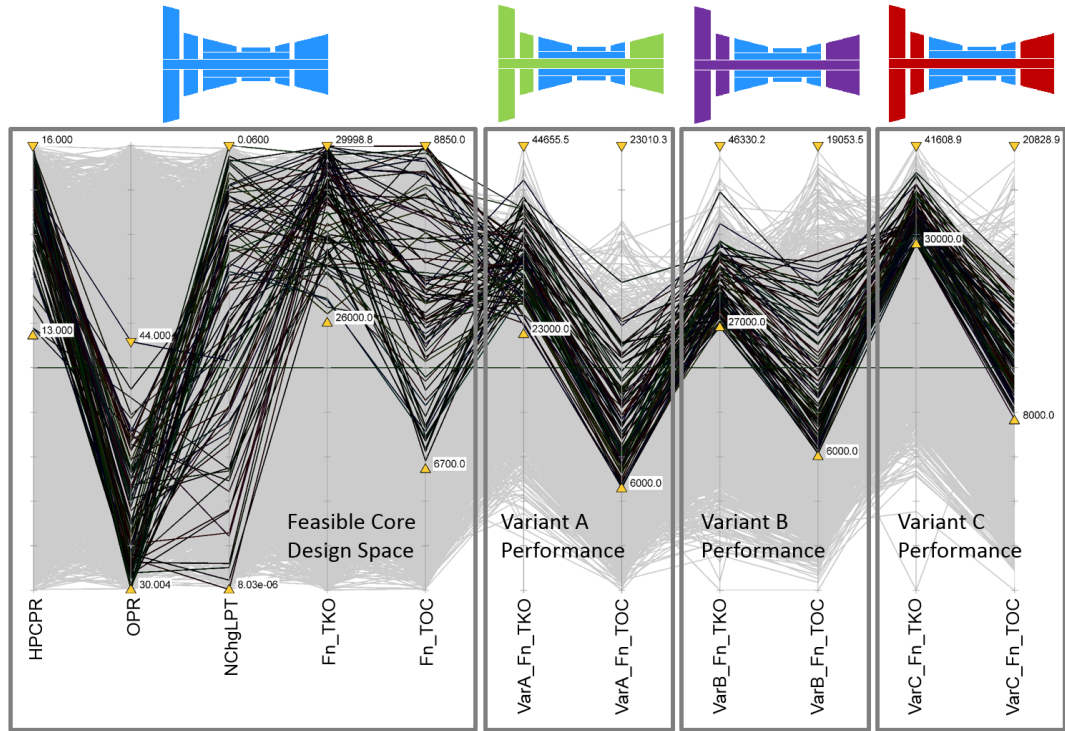


Figure 80: Experiment 3 - Common core design candidates that meet thrust requirements of all three applications while also satisfying all temperature constraints.

lying between 0.5 and 0.7 for a majority of the feasible designs for all three applications. Also, Figure 82 shows the predicted engine pod weights for all applications of each of the feasible candidate designs. The spread in the weight is quite significant, and only samples that provide variant applications that have minimum engine weights remain in the group of feasible candidate designs considered. The coordinate axis present on the far right hand side of both Figure 81 and Figure 82 is the design HPC exit corrected flow of the engine core. The benchmark engine selection made earlier had design HPC exit corrected flow levels of 5.91, 7.09, and 8.74 for Applications A, B, and C, respectively. The parallel coordinate charts of the feasible common core applications show design HPC exit corrected flow levels offering feasible solutions lie between 5.7 and 8.3. The range of corrected flows being in the interior of the benchmark levels is a product of allowing variant applications to have elevated HPT speeds to achieve high takeoff thrust. This effectively represents a HPT vane reset intended to squeeze more working fluid through the core to increase the maximum

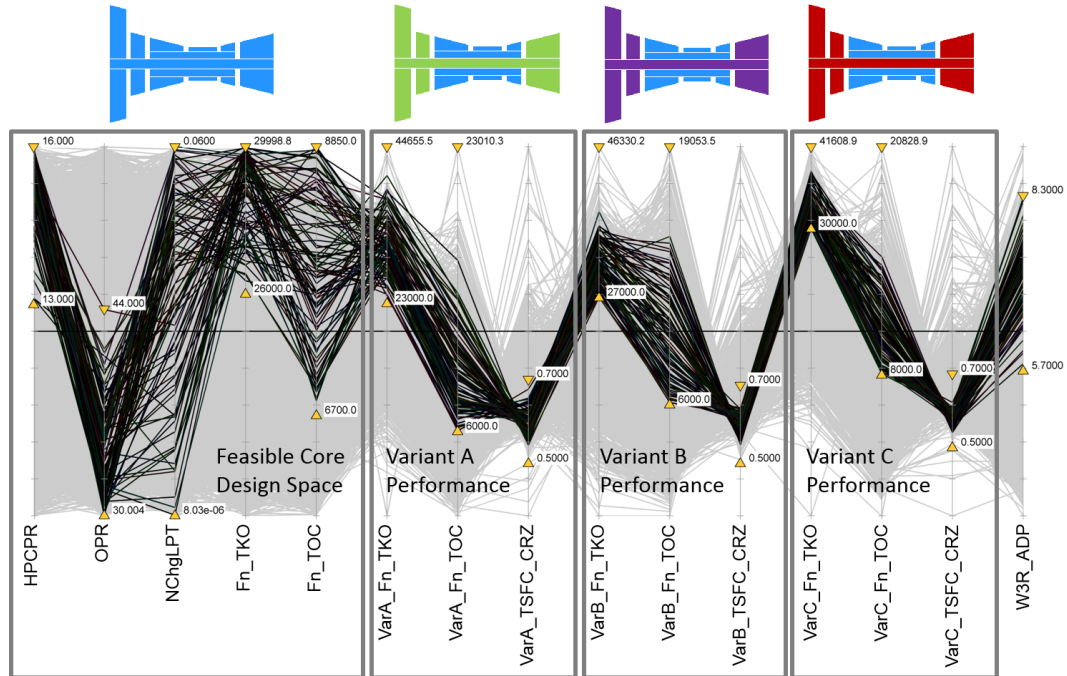


Figure 81: Experiment 3 - Feasible common core design candidates and corresponding cruise TSFC levels for each engine variant application..

achievable thrust at takeoff, while allowing the design flow levels at ADP and 100% speed to remain lower than would otherwise be required if the maximum HPT corrected speed was held at 100% during takeoff.

After considering the regions of engine core feasibility, secondary considerations can be made to further improve common core variant performance levels. Figure 83 contains a parallel coordinate plot of the variant control variables for Application A along with the performance metrics of interest for the application. Again, all infeasible samples that did not meet either the temperature limits or the thrust requirements of the application have been filtered, leaving the black lines representing feasible common core samples. This filtering leaves a very small population of feasible samples to analyze the feasible variant cycle space of Application A. This small population prevents the cycle designer from having a clear view of the definite feasible regions of the variant design space. There are obvious variable values, such as low design FPR and very high design OPR variant designs that cannot satisfy all requirements. The small population of feasible samples prevents the designer to

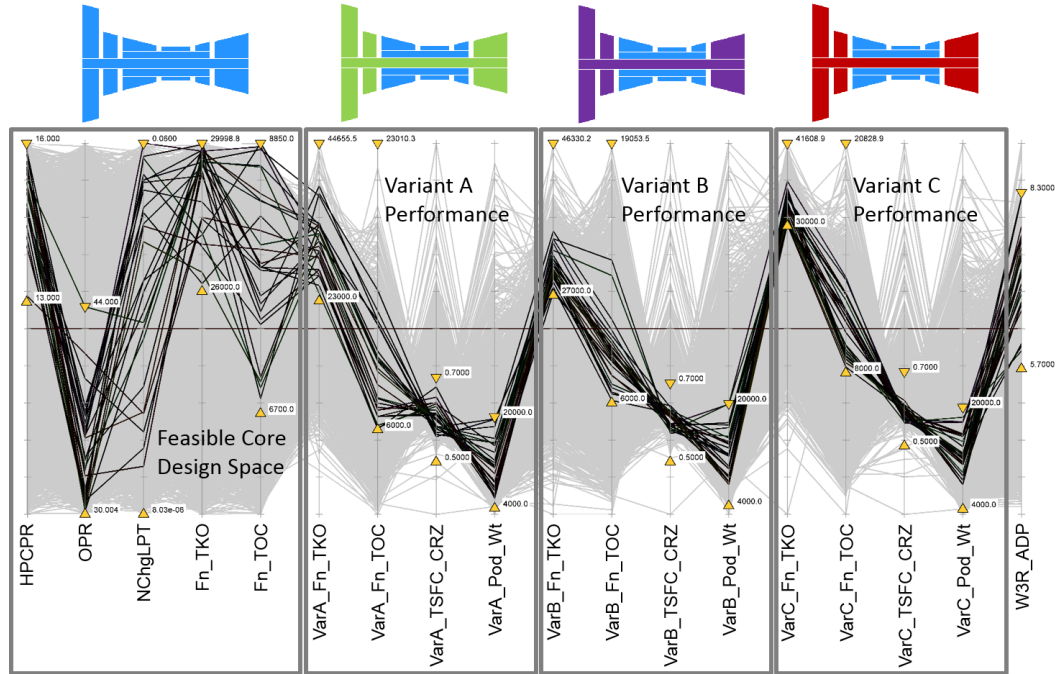


Figure 82: Experiment 3 - Feasible common core design candidates which achieve high performance among the samples collected within the common core application design spaces.

identifying a clear variant design region of interest.

Figure 84 displays the feasible common core design candidates for Application B, along with their corresponding variant cycle settings and 95% likely performance levels. Again, no clear correlations exist to help identify region(s) of interest in improving variant engine performance for Application B. Figure 85 has clearer regions that should be avoided for the variant cycle design for the highest thrust Application C. This high thrust application is a key driver in the overall common core feasibility since it demands the most engine power, both at takeoff and top-of-climb conditions. It is quite clear that low variant FPR, low variant OPR designs do not offer common core applications that meet the required amount of thrust for Application C while satisfying all imposed temperature limits. But again, there is no clear set of input variable settings that guarantees a high performance common core variant design for Application C.

The sparsity of the feasible design space when considering multiple engine applications simultaneously can be problematic when using a traditional sample-based robust design

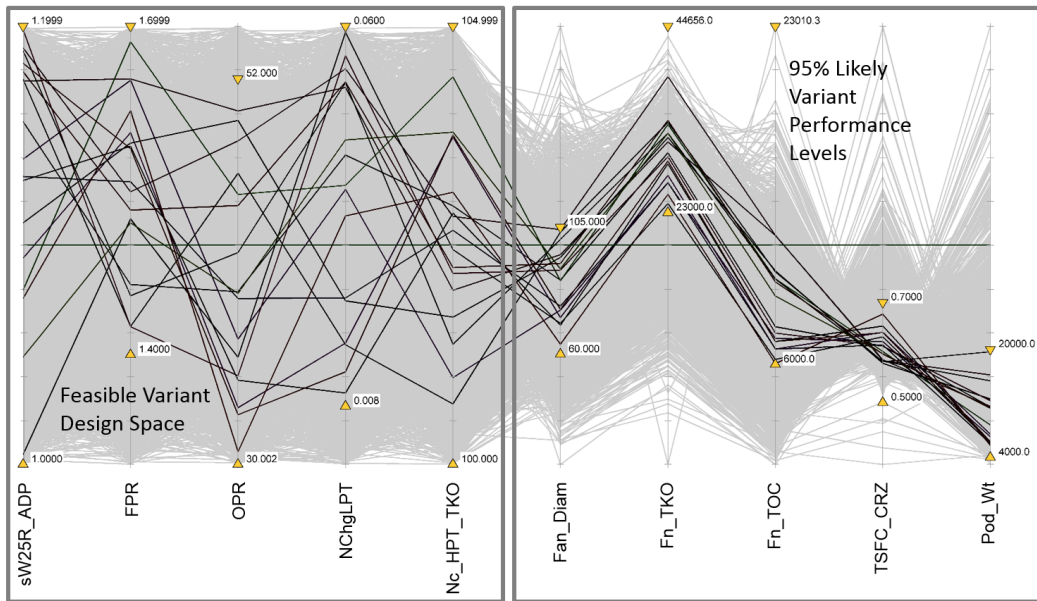


Figure 83: Experiment 3, Application A - Feasible common core variant engine cycle candidates among the sample population.

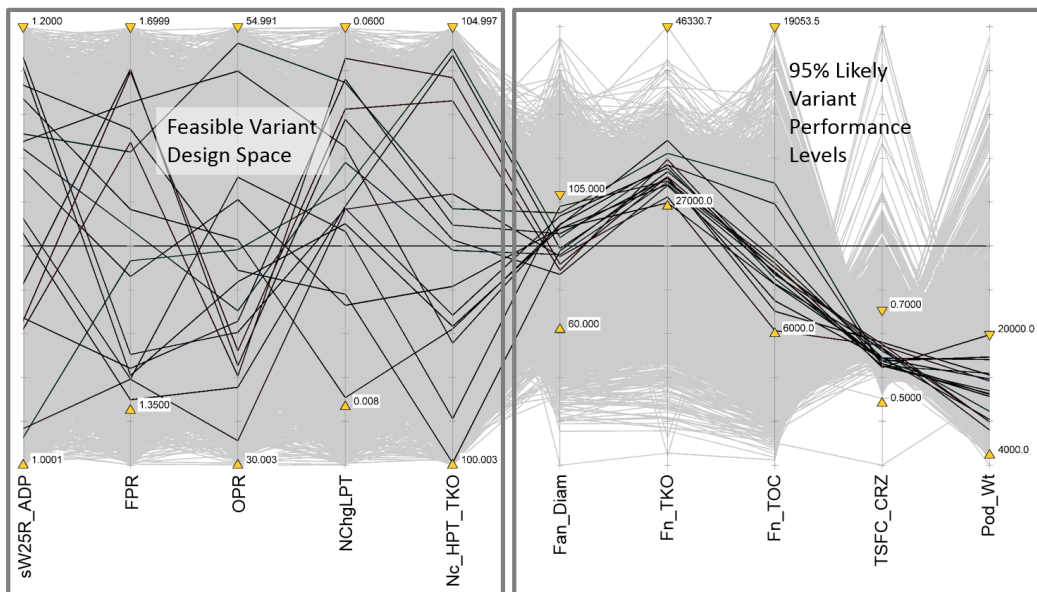


Figure 84: Experiment 3, Application B - Feasible common core variant engine cycle candidates among the sample population.

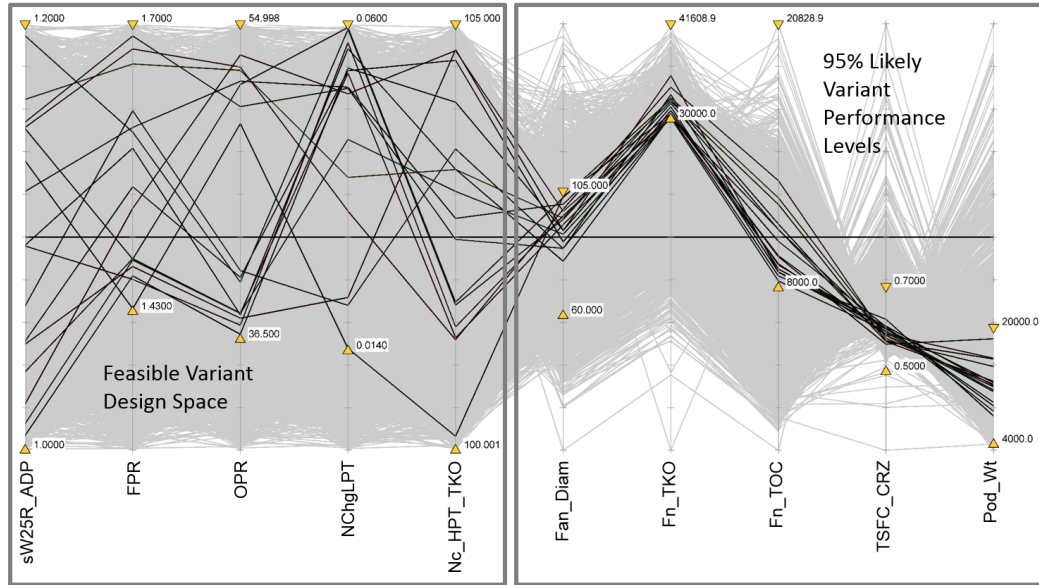


Figure 85: Experiment 3, Application C - Feasible common core variant engine cycle candidates among the sample population.

approach. Since a given computational budget must be reserved for repetitions of design simulations to account for the uncertainty assumed present, even fewer exploratory samples can be used to search for a feasible region that simultaneously meets all requirements and constraints of all applications considered. Traditional methods that depend on sampling to arrive at a high performing design candidate have an extremely low probability of arriving at a multiple application design that meets all requirements, let alone achieving competitive performance levels for all applications considered. Fortunately, the COMMENCE method offers the designer the ability to take advantage of continuous surrogate models that implicitly account for the uncertainty present in the designs while also instantaneously estimating the effects of various core design decisions on all variant applications considered. Constraint analyses can be performed using the continuous surrogate response space. Contours of constant constraint values can be displayed directly on the common core design space of interest. This greatly assists the designer in identifying any feasible regions in a high dimensional design space such as this, and also assists the designer in adjusting the design in order to move in the direction of performance improvement within the feasible region.

Figure 86 contains contours of all the thrust and temperature limits of the three engine applications plotted on two core design dimensions, the takeoff thrust and the design HPCPR of the design engine which defines the common engine core utilized across applications. The contour locations for each variant application are dependent upon all of the core design input variable settings as well as the variant cycle input variable settings for the applications considered. Extensive exploration within the various core design input space was necessary to identify a core design region offering feasible common core variant designs for all three applications considered. All contour plots shown for this experiment are representative of the multiple application common core design selection made and later tabulated. If any input variable settings were to change, the placement and shapes of the contours displayed would likely change.

The feasible region displayed in Figure 86 shows that the lower level of feasible design engine takeoff thrust is not surprisingly limited by the takeoff thrust requirements of each of the three applications. The upper limit of feasible design engine takeoff thrust levels is limited by the HPC exit temperature limits. The diagonal constraints present are the LPT inlet temperature limits that limit the lower takeoff thrust level for a given HPCPR level.

Similarly, Figure 87 shows the feasible core design regions of design engine top-of-climb thrust levels for various HPCPR designs. Again, the thrust requirements limit the lower feasible level of design engine top-of-climb thrust that offers feasible common core applications while the LPT inlet temperature constraints limit the high end of design engine top-of-climb thrust for various HPCPR designs. This is again expected due to the fact that increasing the desired top-of-climb thrust would require more thermal energy to exit the core and in turn do work on the LPT, increasing the temperature of the flow for a given core size.

The last contour chart shown for the feasible core design space is shown in Figure 88, showing the feasible range of design engine OPR at the high HPCPR designs. This feasible region is very small compared to the design space initially considered, showing a range of feasible OPR levels between 43.0 and 43.5 at ADP conditions for the core defining design engine. Keep in mind though that the placement and size of this feasible region is based on

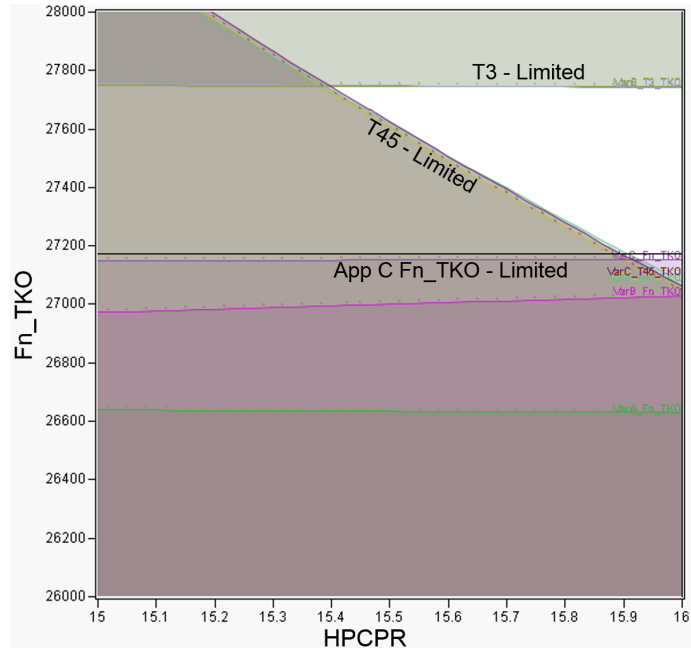


Figure 86: Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine takeoff thrust and HPC design pressure ratio after simultaneous exploration of core and variant engines.

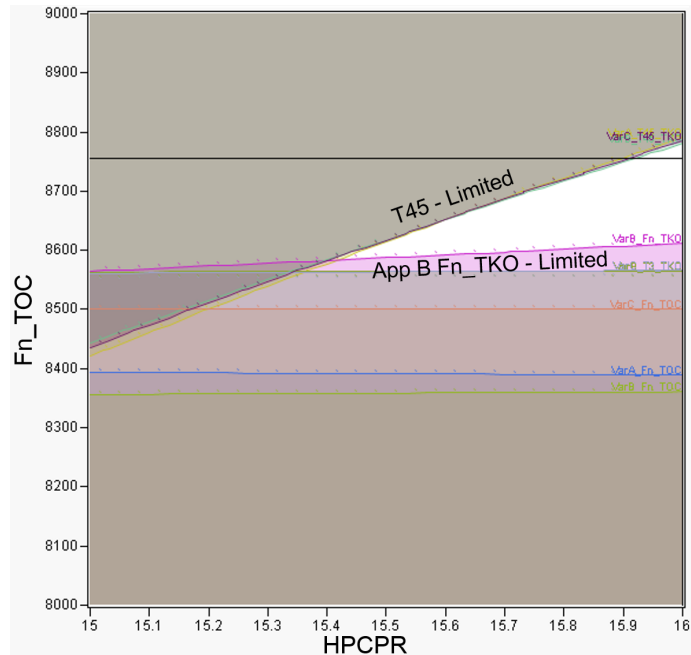


Figure 87: Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine top-of-climb thrust and HPC design pressure ratio after simultaneous exploration of core and variant engines.

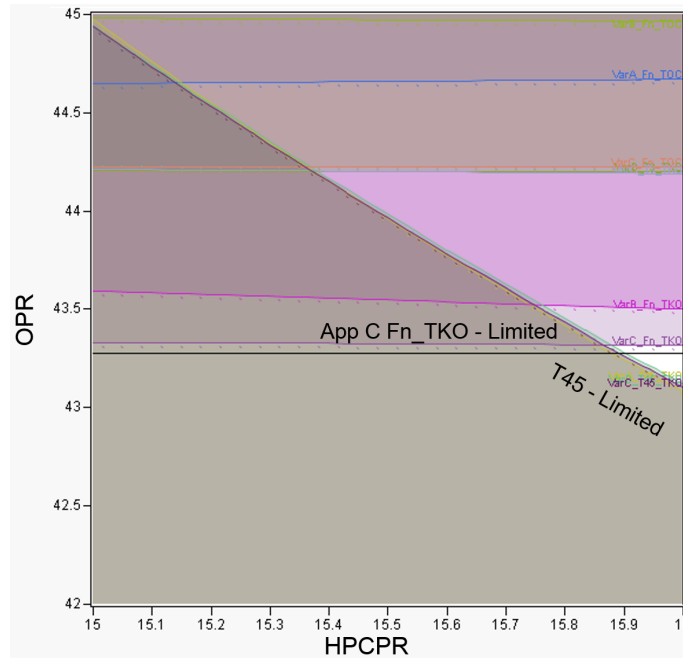


Figure 88: Experiment 3 - Common core design exploration for three variant applications: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio after simultaneous exploration of core and variant engines.

all the other input variable settings not shown in the chart. By adjusting the input variable settings, the size of the feasible region can be increased. However, the settings have been adjusted to be in the region of performance improvement. By improving on performance, more compact cores are considered and more efficient compact engine variants are preferred as well. This shrinks the feasible design region in turn for improved performance. Normally when one considers robust design selections, one would want as large a feasible design region as possible in order to remain robust to changing requirements and constraints. However, since all the performance estimates, requirements and constraint evaluations and all corresponding contours are 95% confidence interval values (meaning that the design is 95% likely in meeting or improving upon the contour value), a design selection within a small feasible region can still be considered robust since the region is essentially representing where there is more than a 95% likelihood of meeting all requirements and constraints considered. In order to visualize all three core design planes simultaneously, Figure 89 displays all three contour profilers side-by-side.

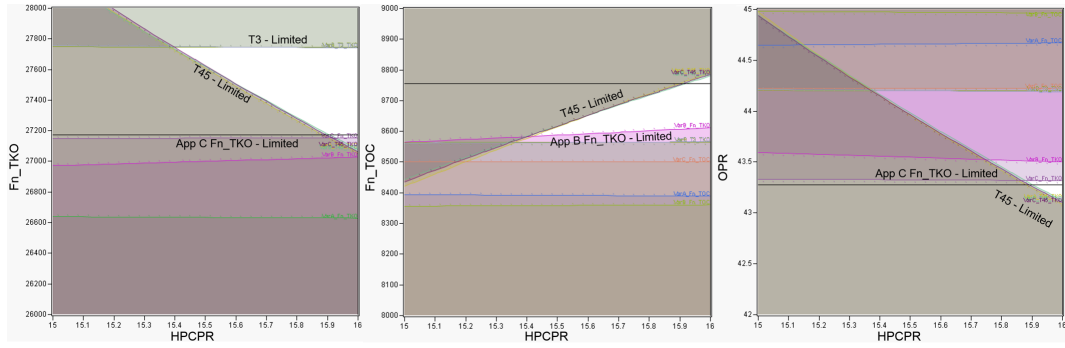


Figure 89: Experiment 3 - Side-by-side comparison of contour profilers representing several core design planes.

A major advantage of using contour profilers when searching for feasible and favorable design regions that cannot be shown on paper is their dynamic nature. The designer has the ability to adjust design settings on-the-fly and see the resultant effects on the constraints and performance instantaneously. What-if games can be played, making adjustments to assumptions, constraint values, or technology scenarios and the dynamic contour profiler updates to reflect those adjustments. The dynamic nature of these visualizations are very beneficial to decision makers when weighing the various options for the design of interest.

Once exploring and setting the core design region of interest has been performed, each design space corresponding to the engine variant applications can be explored using the probabilistic surrogate models predicting common core engine variant performance. Required thrust levels and any constraints imposed on each of the engine applications can be displayed in contour profile charts, helping aid the designer in determining the feasible design region for the core design selection.

Figure 90 displays the feasible common core variant design space for Application A in the design planes of variant HPT takeoff corrected speed, the core corrected flow scale factor, and the variant FPR. For the current set of input variable settings, the LPT inlet temperature constraint and the required thrust at top-of-climb are the limits constraining the feasible design space. As the amount of core flow scaling is reduced and the amount of HPT design overspeed at takeoff is reduced, the amount of work performed by the engine core is reduced, causing the amount of excess thermal energy exiting the core to be

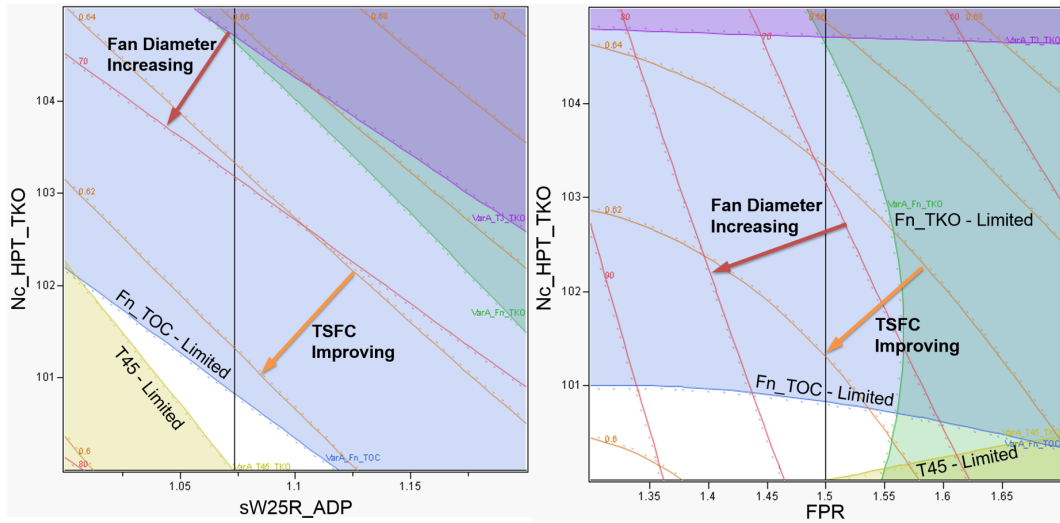


Figure 90: Experiment 3 - Common core variant exploration for Application A: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant A feasibility.

increased, corresponding to an increase in the LPT inlet temperature for a given amount of LPT nonchargeable cooling. This causes the LPT inlet temperature constraint to become active at lower level combinations of sW_{25R} and $N_{C,HPT,Var}$. Conversely, at these two variant cycle parameter values increase, the fraction of flow traveling through the engine core with respect to the amount of flow bypassing the core is increased for a given design FPR and extraction ratio and engine. This causes the achievable thrust at top-of-climb to be reduced due to less propulsor flow traveling through the bypass stream, reducing the net thrust. Also present in the contour chart are contours of how cruise TSFC and fan diameter change as the design settings vary within the plane of design values. As the flow scaling and takeoff high spool speed are reduced, the cruise TSFC is improved and the fan diameter grows due to a higher variant design bypass ratio. The core flow scaling and takeoff HPT corrected speed settings selected for Application A provide the best cruise TSFC while remaining in the feasible variant design region.

Considering the other variant cycle design plane displayed in Figure 90, the LPT inlet temperature limit and top-of-climb thrust requirements are constraining most of the space. Additionally, the requirement of Application A limits the high levels of design FPR due to

a lack of core power available to drive the higher FPR fan with the common core definition. The TSFC contours expectedly show that as design FPR and takeoff corrected speed are reduced, the cruise TSFC is reduced due to a bypass ratio increase and accompanying increase in propulsive efficiency. However, these designs have larger fan diameters and accompanying extensive weight increases. Therefore, the design FPR and takeoff high spool speed is selected that compromises between the cruise TSFC achievable and fan diameter reduction to minimize weight.

Similar to the contour profile charts shown for the common core variant design space addressing Application A requirements, Figure 91 displays the feasible common core variant design region able to satisfy the Application B requirements. For the core design selection considered, the feasible variant design space is much smaller for this application, requiring almost 20% more takeoff thrust than Application A. As shown in the figure, the LPT inlet temperature limit ($T_{45,max}$) and the HPC exit temperature limit ($T_{3,max}$) define the feasible variant design space along with the increased takeoff thrust requirement. The takeoff thrust requirement holds the design high spool corrected speed to higher levels for the common core application, showing that increased takeoff core power is necessary to produce the thrust required than what the core defining design engine produced at takeoff. The LPT inlet temperature limit causes the amount of core flow scaling to be increased from the design level in order to provide more core working fluid to perform propulsive work while remaining within the temperature limit imposed.

Figure 91 shows the active constraints limiting the high spool design speed and variant design FPR levels for the selected common core design. The trends are similar to those constraining the Application A common core variant design. An intermediate variant design FPR level was selected that minimized the fan diameter necessary to provide the takeoff thrust required of the variant designed for Application B.

Moving onto the highest thrust common core engine variant application considered, Application C, it is shown in Figure 92 that there is a very small region of feasibility when using the common core definition being considered. Significant core flow scaling is necessary along with high takeoff over-speeding in order to meet all the application requirements and

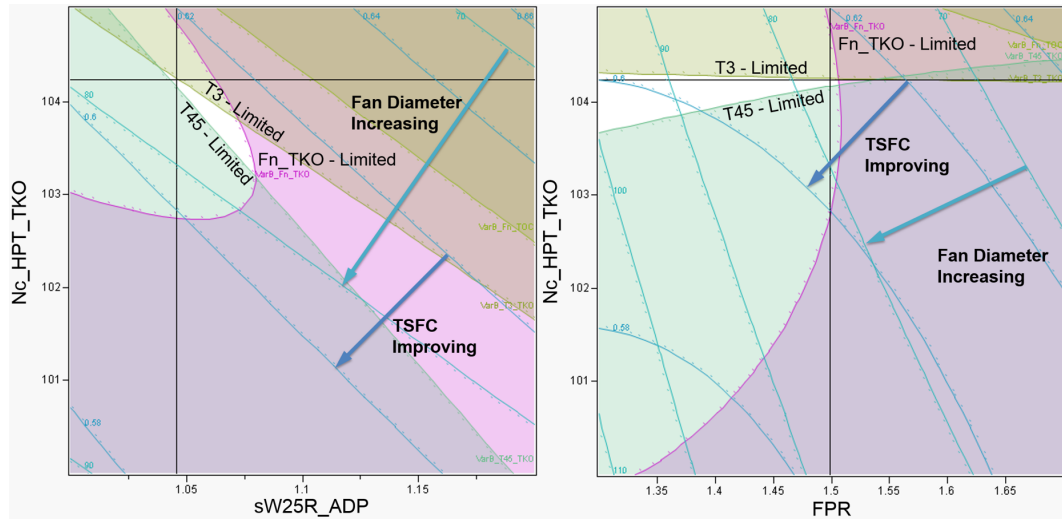


Figure 91: Experiment 3 - Common core variant exploration for Application B: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant B feasibility.

constraints. The feasible region is constrained by the LPT inlet and HPC exit temperature limits as well as the takeoff thrust requirement just as with the previous common core application. Looking at second chart in the figure, the feasible design region could be increased by reducing the variant design FPR, but at the cost of greatly increasing the fan diameter, which would cause the engine weight to increase as well. Although the feasible region is small, a candidate variant design contained within this region, just as with the other common core applications, carry with it a 95% likelihood of achieving or surpassing all the requirements and constraints imposed on the common core application.

4.4.3.3 Multiple Application Common Core Design Selection

By simultaneously exploring the common core design space, analyzing the core defining engine design space and each of the common core variant design spaces, a design region was identified that offered a common core design and corresponding variant applications that met all requirements and constraints imposed on the multiple application engine design problem. Table 53 contains the design selections of the core defining engine as well as the settings selected for each of the common core variant designs. Note that the variable

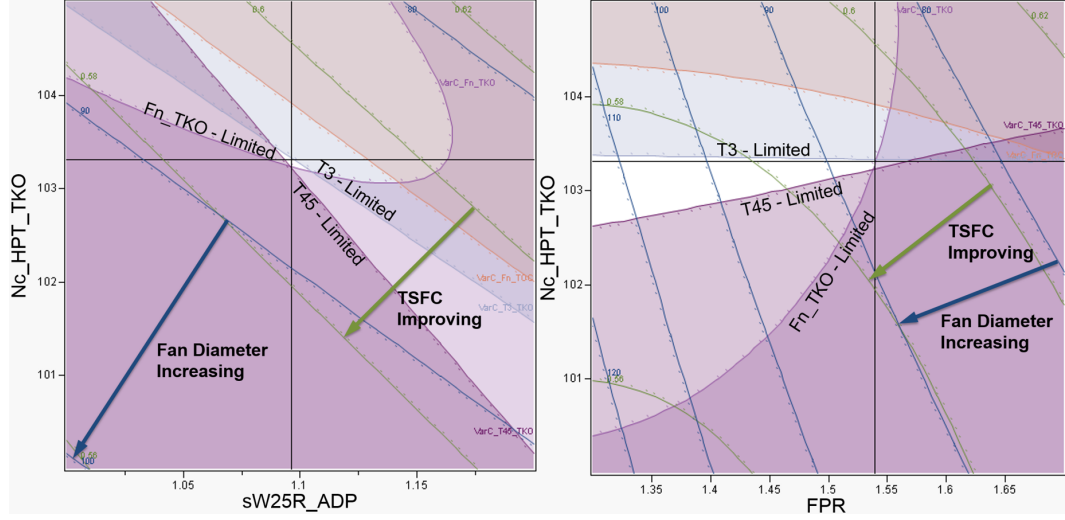


Figure 92: Experiment 3 - Common core variant exploration for Application C: Constraints determining the amount of TKO over-speed, core flow scaling, and FPR necessary for Variant C feasibility.

selections were made through visualization and manual exploration of the multi-application common core design space. Therefore, further improvement of the common core family performance may be achievable with the use of an optimizer.

In order to determine the performance compromises made by applying a common core across multiple applications, Table 54 contains the common core evaluation metrics introduced in Chapter 3. Each common core variant is evaluated based on the performance deviation from its corresponding benchmark design, with equal preference in cruise TSFC and pod weight performance metrics. Referring back to Chapter 3, Equation 40 is used to evaluate a common core engine family as a whole is then evaluated based the individual variant performance levels, as shown in Equation 41, with equal preference given to each of the three engine applications.

$$y_i = \left(\frac{TSFC_{i,CC} - TSFC_{i^*}}{TSFC_{i^*}} \right) w_{TSFC} + \left(\frac{Wt_{i,CC} - Wt_{i^*}}{Wt_{i^*}} \right) w_{Wt} \quad (40 \text{ Revisited})$$

$$Y_{CC} = \sum_{i=1}^{N_{apps}} w_i y_i \quad (41 \text{ Revisited})$$

The core defining engine cycle selection provides a core definition with enough flow

Table 53: Experiment 3 - Common Core Variant Engine Cycle Selections and corresponding 95% confidence interval performance levels for each of the three applications considered.

Design Engine Cycle Parameter	Core Defining Cycle Value		
π_{Fan} at ADP	1.70		
π_{HPC} at ADP	16.00		
$\pi_{Overall}$ at ADP	43.27		
T_4 at TKO, $^{\circ}R$	3,309.		
ER at ADP	0.90		
$NChg_{LPT}$, $\%W_{25}$	0.00%		
F_N at TKO, lbf	27,175.		
F_N at TOC, lbf	8,755.		
Application Requirement	App 3.A	App 3.B	App 3.C
Takeoff Thrust, lbf	23,000	27,000	30,000
Top of Climb Thrust, lbf	6,000	6,000	8,000
Variant Design Cycle Parameter	Variant 3.A	Variant 3.B	Variant 3.C
π_{Fan} at ADP	1.50	1.50	1.54
$\pi_{Overall}$ at ADP	30.66	36.38	43.66
T_4 at TKO, $^{\circ}R$	3,301.	3,344.	3,329.
ER at ADP	1.16	1.18	1.21
$NChg_{LPT}$, $\%W_{25}$	5.84%	1.56%	1.61%
sW_{25R} at ADP	1.074	1.046	1.096
$N_{C,HPT,Var}$ at TKO	100.0	104.2	103.3
95% CI Metric Value	Variant 3.A	Variant 3.B	Variant 3.C
F_N at TKO, lbf	23,481	27,112	29,519
F_N at TOC, lbf	6,284	6,371	8,283
$T_{3,max}$, $^{\circ}R$	1,565.	1,700.	1,700.
$T_{45,max}$, $^{\circ}R$	2,300.	2,300.	2,300.
W_{3R} at ADP, lbm/s	6.34	6.34	6.34
TSFC at CRZ, $lbm/lbf/hr$	0.6101	0.6120	0.5893
Pod Weight, lbm	4,489	4,625	7,259
Fan Diameter, in	77.4	77.7	87.0

Table 54: Experiment 3 - Common core variant and overall program evaluation based on comparisons to new centerline benchmark cycles independently designed for each application considered.

95% CI Metric Estimate	Bench 3.A	Bench 3.B	Bench 3.C
TSFC at CRZ ($lbm/lbf/hr$)	0.6526	0.6440	0.6525
Pod Weight (lbm)	3,685	3,633	6,458
95% CI Metric Estimate	Variant 3.A	Variant 3.B	Variant 3.C
TSFC at CRZ ($lbm/lbf/hr$)	0.6101	0.6120	0.5893
Pod Weight (lbm)	4,489	4,625	7,259
Preference Levels	App A	App B	App C
TSFC Preference: w_{TSFC}	0.5	0.5	0.5
Weight Preference: w_{Wt}	0.5	0.5	0.5
App Preference: w_{App}	0.33	0.33	0.33
Performance Deviation, y_i	7.66%	11.17%	1.36%
Total Common Core Performance Deviation, Y_{CC}	6.72%		

capacity and available power to offer engine variants for each of the three sets of requirements considered, all at the same assumed technology level. Scaling of the core inlet corrected flow was allowed to provide increased core working fluid when needed in order to achieve significant power growth while remaining under the imposed temperature limits. The high pressure shaft was also allowed to operate at elevated corrected speeds at takeoff in order to attempt to reach the takeoff thrust requirements of each application. Although these core modifications have an accompanying reduction in the variant cycle thermal efficiency levels, the modifications offer increased engine core capabilities when utilized for multiple sets of application requirements.

The scaling of core inlet corrected flow and increase in takeoff high spool corrected speed have allowed for a compact engine core to be utilized for applications that would otherwise require increased core corrected flow. The corrected HPC exit flow of the common core definition is at a design level of $6.34 lbm/s$, as opposed to benchmark engine values of 5.91 , 7.09 , and $8.74 lbm/s$ for Applications A, B, and C, respectively. The common core design value is on the lower side of the spectrum desired by the benchmark engines. The more compact common core definition allows for each of the common core engine variants

to achieve lower levels of cruise TSFC. However, all common core variant designs weigh more than each of the benchmark engine selections. This is the key compromise made when employing common core engine applications. The common core size is large enough to provide power, but provides the necessary power at off-design operational levels away from the design values at which the core was originally designed. This is a necessary compromise if the thrust levels for which common core designs are selected are considered, requiring the designer to rationalize whether the added engine size of the common core application is worth any development costs saved by utilizing a common core.

4.4.4 Conclusions

This experiment demonstrates how the Common Engine Core Evaluation (COMMENCE) method can be used to consider multiple applications when exploring and selecting design within a common engine core design space. It shows how probabilistic surrogate models can be utilized to successfully account for assumed sources of uncertainty present without the need to repeat simulations of each candidate design being evaluated under uncertainty. Variability in engine performance is implicitly accounted for in the probabilistic surrogate models, allowing the designer to estimate any confidence interval performance level desired. Another advantage of the method is the fact that it offers the designer the ability to perform engine cycle design space explorations in a manner similar to how he or she traditionally carries out such an exploration. However, instead of only predicting deterministic performance levels for a single engine application when performing cycle explorations, the COMMENCE method allows for design implications to be made on multiple engine applications when making engine core design decisions. The Enhanced Robust Design Simulation (ERDS) method that is integrated into the COMMENCE process allows for robust common core designs to be selected that are likely to perform well and meet all requirements and constraints while being subject to various sources of uncertainty assumed present. Applications that constrain and compromise the common core design performance can be identified, offering the designer the ability to consider whether the constraining application is worth having in the common core program. The method provides quantitative measures of variant design

implications when exploring and making core design decisions. This offers decision makers the ability of considering more engine program trade-offs across a portfolio of products earlier in the design process. Robust core design decisions can now be made that will offer more of an ability to select and employ an engine core across multiple sets of requirements for which it is being designed, while also having a high level of accompanying confidence in the core design successfully providing feasible variant engines.

This experiment also highlights the dynamic capabilities that the COMMENCE method enables. Various exploration, visualization, and optimization techniques can be used during the consideration and selection of common core designs. The probabilistic surrogate models in $SSC_{Des,Prob}$ and $SSD_{Var,Prob}$ are continuous, closed form equations allowing for instantaneous evaluation of a candidate design. Additionally, the models enable the use of optimization techniques to be employed to search for the most desirable design settings across all control variable dimensions.

While this experiment has extended the common core design rules presented in Experiment 2 to multiple applications, it also assumed that all engine applications considered were at the same technology level and under the same set of uncertainty distributions. The next experiment considers more engine applications at different technology levels, representing designs that would be projected to be introduced at different times throughout the life of the engine program. More applications will be considered and the common core design space of the more complex multiple application problem will be explored with the goal of reaching a feasible engine core definition to allow for variant designs to be offered for all applications considered throughout the projected program of engine applications.

4.5 *Experiment 4: Common Core Design for Multiple Applications Introduced Over Time*

- **Primary Research Question:** *How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?*

Hypothesis: *Simultaneous simulation and evaluation of current and future common core applications should be performed when exploring the common core design space in order to quantitatively estimate the feasibility and likely performance levels of program applications due to changes in the common core definition. If this is possible and implemented, the likelihood of achieving feasible, competitive common core variant designs will be increased while minimizing the amount of mitigation actions required later in the program.*

- **Research Question 1:** *How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?*

Hypothesis 1: *Probabilistic performance levels of candidate cycle designs should be estimated with the use of surrogate models that predict likely performance under various inputted uncertainty scenarios for any desired confidence interval.*

This experiment is a demonstration of the full set of capabilities enabled by the fully integrated Common Engine Core Evaluation (COMMENCE) method. The previous experiment tested the method’s ability to consider multiple common core applications, all at the same assumed technology level. This experiment explores the common core design space for a larger set of applications, each under a unique uncertainty scenario. The probabilistic surrogate models of *Surrogate Set C* ($SSC_{Des,Prob}$) will be used to evaluate the core defining design engine. The probabilistic surrogate models of *Surrogate Set D* ($SSD_{Var,Prob}$) will be used to evaluate all common core variant designs considered. This highlights the enabled flexibility of a single set of surrogate models produced by the ERDS method being able to

accurately provide any desired confidence interval performance level for a design under a wide range of uncertainty scenarios. This allows the designer to account for the advancement of technology of an engine program when making common core design decisions early in the development time-line. By simulating later term engine variants with increased technology level impacts as well as increased uncertainty in those impacts due to being longer term projections, designers can account for the program later taking advantage of advanced technology in later term common core engine applications. This also is advantageous in allowing the designer to select which technology impact categories to pursue in order to achieve the best common core variant or overall engine program performance improvement by giving him freedom in controlling the uncertainty distributions when exploring the robust common core design space.

Just as in Experiment 3, feasible regions of the common core design space will be identified and explored. Seven sets of requirements will be considered for common core engine variant applications. Robust benchmark engines will be selected for each application and its corresponding technology and requirements uncertainty assumed, establishing the achievable engine performance of a new centerline engine. Then the common core design space will be explored to see if and where any feasible common core design regions exist. Candidate common core designs will be considered, and corresponding common core variant design spaces will be further explored and robust candidates selected.

This experiment also considers options for further advancement of the technology capabilities of certain engine variant applications past initially assumed levels in order to improve the overall program performance. Limiting factors are identified that when mitigated enable a more compact common core definition to be used. Additionally, consideration of relaxing a constraining application's takeoff thrust requirement is made in order to achieve the same goal, to improve the overall engine program performance by shifting the core design selection to a more favorable region that would otherwise be infeasible.

This final experiment demonstrates the full suite of capabilities enabled by the fully integrated COMMENCE method. The key contributions of this work will be utilized to perform a multiple application common core design study that would otherwise require

Table 55: Experiment 4 - Engine Applications considered for the present multiple application common core design problem.

App	Tech Level	Takeoff Thrust	Top of Climb Thrust	Most Likely HPX	Most Likely Air Bleed	Max Fan Diameter
4.A	Low	22,000 <i>lbf</i>	5,400 <i>lbf</i>	200 <i>shp</i>	2.00 <i>lbm/s</i>	70 <i>in</i>
4.B	Low	24,000 <i>lbf</i>	5,800 <i>lbf</i>	200 <i>shp</i>	2.00 <i>lbm/s</i>	70 <i>in</i>
4.C	Mid	27,000 <i>lbf</i>	6,400 <i>lbf</i>	250 <i>shp</i>	2.35 <i>lbm/s</i>	64 <i>in</i>
4.D	Mid	30,000 <i>lbf</i>	6,400 <i>lbf</i>	250 <i>shp</i>	2.50 <i>lbm/s</i>	70 <i>in</i>
4.E	High	27,500 <i>lbf</i>	6,000 <i>lbf</i>	250 <i>shp</i>	2.35 <i>lbm/s</i>	64 <i>in</i>
4.F	High	24,000 <i>lbf</i>	6,000 <i>lbf</i>	300 <i>shp</i>	3.00 <i>lbm/s</i>	61 <i>in</i>
4.G	High	30,000 <i>lbf</i>	8,000 <i>lbf</i>	300 <i>shp</i>	3.00 <i>lbm/s</i>	70 <i>in</i>

an immensely greater computational budget and much more data processing. Multiple applications under multiple uncertainty scenarios will be simultaneously considered when making common core design considerations. Additional activities enabled by the method will be performed, demonstrating the great flexibility, dynamic capabilities, and ease of use of the COMMENCE method.

4.5.1 Sets of Requirements for Multiple Applications

Seven applications are considered in the present experiment. The requirements of each application are listed in Table 55, which contains the thrust requirements, most likely horsepower and customer bleed requirements, and the maximum fan diameter allowed for each engine. Also shown is the assumed technology level of each application. Applications A and B are assumed to be the engine program's near-term applications, with the lowest level of assumed technology. Applications C and D are assumed to be mid-term engine program applications and have an accompanying mid-level assumed technology level. Applications E, F, and G are long-term engine program applications, having the highest technology level considered. These technology levels will be described in further detail in the following section.

Figure 93 contains graphical representations of the engine program applications considered. Each image represents a common core variant engine. The purple components show the imposed geometric core commonality between applications. Each representation's fan

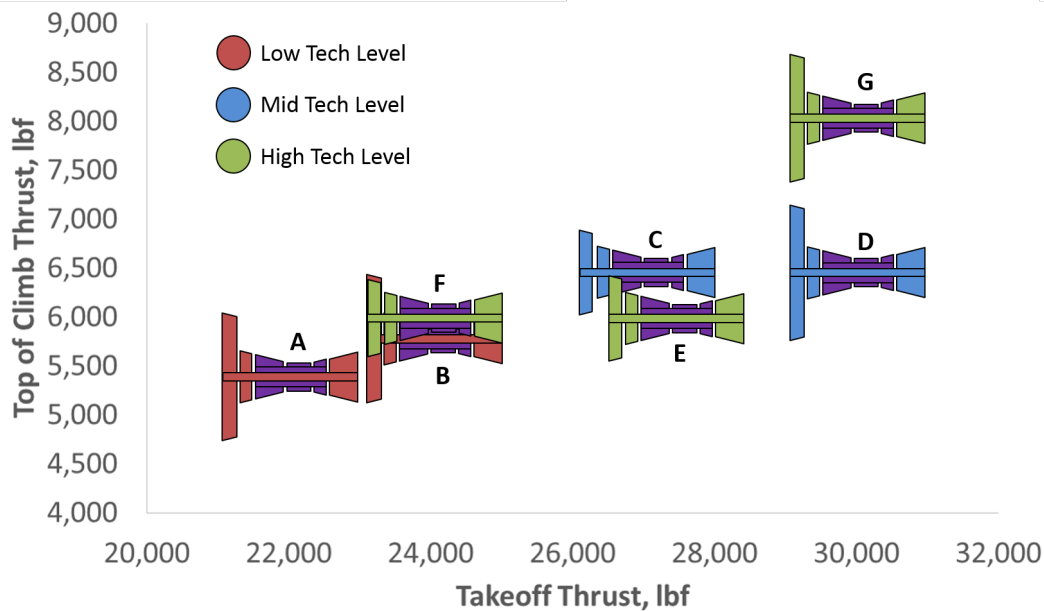


Figure 93: Experiment 4 - Common Core Variant Representations, Thrust Requirements and Maximum Fan Diameter Limits, each application colored by its corresponding technology level assumed.

size depicts the maximum fan diameter allowed. Each application's color is based on the technology level assumed for the application. The engine representations are placed on the chart based on their top-of-climb and takeoff thrust requirements. This shows that the initial low technology Applications A and B are low thrust engines with large fan diameters allowed. The mid-level technology Applications C and D have increased thrust requirements while Application C also has a reduction in its allowed fan diameter. The high technology level Applications E, F, and G aim to replace earlier applications of similar thrust levels, with Application G aiming to achieve significant top of climb thrust growth. Application F has a further restriction in the allowed fan diameter, requiring the smallest engine capture area of the applications considered.

Looking at the overall engine program considered, later term applications are aimed to achieve more than 36% growth in takeoff thrust capabilities with respect to the near term Application A while also aiming to provide more than 48% growth in top-of-climb thrust capabilities. A significant range of propulsive power is desired from the common engine

core definition which is to be applied across the engine program.

Note that the horsepower and bleed extraction requirements in Table 55 are listed as *most likely* requirements. Uncertainty distributions have been constructed for each application's installation requirements, each of which is displayed in Table 56. The peak probability of each distribution of installation requirements is located at the most likely installation value. The assumed confidence in the most likely value being realized is reduced for later engine applications due to the installation projections being for longer-term engine applications that will enter the market later in the life of the engine program. Therefore, Applications A and B have the highest associated confidence in the realized installation estimates, Applications C and D have less associated confidence, and the later-term Applications E, F, and G have the least amount of confidence in the most likely installation values being realized. When simulating common core designs, both the core defining engine and all variant designs, the variability of these requirements will be accounted for when estimating candidate design performance levels. The installation requirements, both of which are core extractions, play a role in determining the required size of the common core definition in order to provide these uncertain requirements while also meeting the thrust requirements with a high level of associated confidence, all while meeting any constraints imposed on the engine applications.

4.5.2 Technology Scenarios for Multiple Time Periods

As mentioned above, there are three assumed levels of technology that are considered in the present multiple application common core design problem. The four categories of technology impacts and their varying uncertainty distributions for the three technology levels are listed in Table 57. The low level technology impact uncertainty distributions have the lowest modal value of improvement, but also have the lowest amount of variability around the modal value. As the technology level increases, the modal improvement value is improved, but with accompanying increases in the variability around the modal value due to increased uncertainty of farther out impact projections. Changes in the technology uncertainty distributions can be made by simply changing the beta distribution shape parameter

Table 56: Experiment 4 - Application Installation Requirements Uncertainty Distributions assumed for each of the seven Common Core Variant Applications considered in the present experiment.

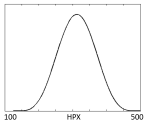
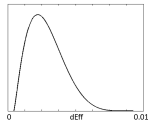
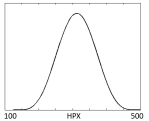
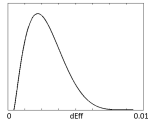
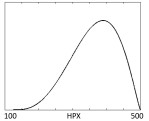
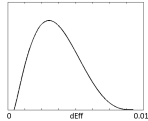
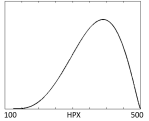
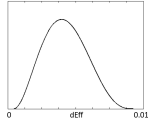
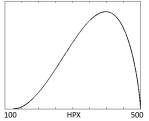
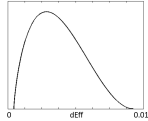
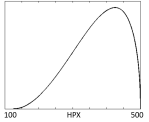
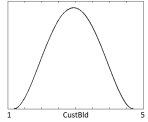
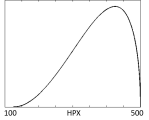
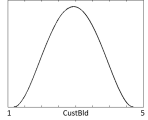
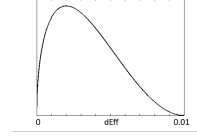
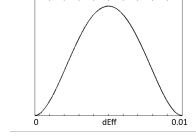
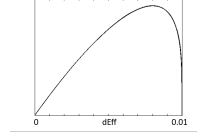
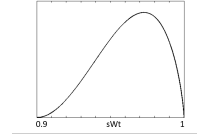
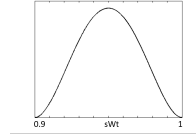
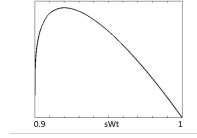
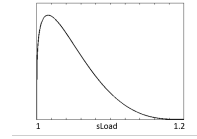
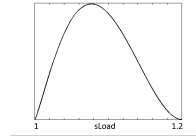
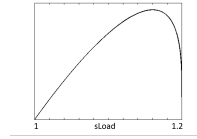
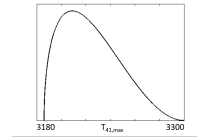
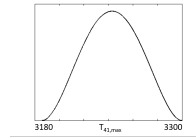
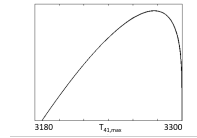
Application	HPX Requirement Uncertainty	Customer Bleed Requirement Uncertainty
App 4.A		
App 4.B		
App 4.C		
App 4.D		
App 4.E		
App 4.F		
App 4.G		

Table 57: Experiment 4 - Technology Impact Uncertainty Distributions assumed for the three technology levels considered in the present multiple application experiment.

Tech Category	Low Level	Mid Level	High Level
Component Efficiency Improvement			
Component Weight Reduction			
Component Loading Improvement			
HPT Inlet Temperature Allowed			

values that the probabilistic surrogate models of likely engine performance were trained as functions of.

In addition to the four technology impact improvement categories considered, the temperature limits imposed on the HPC exit temperature ($T_{3,max}$) and the LPT inlet temperature ($T_{45,max}$) are assumed to increase as the engine program advances over time. The assumed temperature limits are listed in Table 58 for each common core application considered.

Just as with the uncertainty in the installation requirements, these sources of technology uncertainty are accounted for and implicit in the probabilistic surrogate models employed during the following design space explorations performed. The shape parameter values corresponding to each uncertainty distribution listed above are used when simulating engine performance of each common core application, and various confidence interval performance estimates can be estimated based on all the relevant design variable settings as well as

Table 58: Experiment 4 - HPC exit temperature and LPT inlet temperature limits imposed on each of the seven engine applications considered.

Application	HPC exit temperature limit ($T_{3,max}$)	LPT inlet temperature limit ($T_{45,max}$)
4.A	1,700 °R	2,300 °R
4.B	1,700 °R	2,300 °R
4.C	1,750 °R	2,350 °R
4.D	1,750 °R	2,350 °R
4.E	1,800 °R	2,400 °R
4.F	1,800 °R	2,400 °R
4.G	1,800 °R	2,400 °R

the uncertainty distribution shape parameter settings, allowing the designer to make selections that provide a high level of confidence in satisfying all requirements and constraints considered.

4.5.3 Common Core Engine Design for Multiple Applications at Various Time Periods

4.5.3.1 Benchmark Design Explorations and Selections

Referring back to the COMMENCE method shown in Figure 21, after establishing the engine applications for which common core variants are to be designed, benchmark engines are selected for each set of customer requirements. For each application, the ERDS method is used to explore and select a benchmark engine design that has a high likelihood of meeting all requirements and satisfying all constraints imposed. Exploration of the design space contained within the control variable ranges listed in Table 18 is performed for each set of requirements. For each application’s set of requirements and corresponding uncertainty scenario, the exact process used in Experiment 1b is used to explore and ultimately select a candidate benchmark design that meets all requirements and achieves the highest scoring performance among the feasible designs considered. For each benchmark design problem, 10,000 random designs are uniformly sampled throughout the single application design space. The probabilistic surrogates of *Surrogate Set C* ($SSC_{Des,Prob}$) were used to evaluate all 70,000 unique designs sampled to explore the seven applications. The designs that are

Table 18 Revisited: Clean sheet engine control variables and corresponding ranges explored during benchmark design space explorations. For consistency, identical ranges are used to explore the core defining engine design space.

Variable	Units	Min	Max
π_{Fan} @ ADP	-	1.3	1.7
π_{HPC} @ ADP	-	9.0	16.0
$\pi_{Overall}$ @ ADP	-	30.0	55.0
T_4 @ TKO	$^{\circ}R$	3,300	3,750
ER @ ADP	-	0.90	1.25
$NChg_{LPT}$	% W_{25}	0.00	0.06
F_N @ TKO	lbf	20,000	30,000
F_N @ TOC	lbf	5,900	8,850

unable to satisfy all requirements and constraints are filtered out of the population. Of the remaining candidate designs, a robust design selection is made in the presence of the application-specific assumed uncertainty distributions.

Tables 59, 60, and 61 list the design cycle settings and corresponding performance characteristics of the top scoring feasible benchmark designs for the low, mid, and high technology level single engine applications, respectively.

Before moving onto the common core design exploration for these seven applications, there are some interesting observations that can be drawn from the benchmark engine cycle selections. The first is the amount of LPT nonchargeable cooling flow selected for each benchmark application. Most benchmark engines use the maximum amount of LPT cooling flow, 6% of the core inlet flow, in order to bring the LPT inlet temperature below each application's imposed $T_{45,max}$ limit. This allows for the maximum amount of thermal energy to enter the LPT, providing maximum power to the propulsor to produce thrust. This indicates that there is ample core power available to both drive the LP system as well as pressurize the additional cooling flow used to bring the LPT inlet flow within its temperature limit. Additionally, the maximum allowed design extraction ratio (ER at ADP, defined in Equation 50) was selected for Applications A and B. This shows that an engine with a 70 *in* fan diameter can easily provide the relatively low thrust requirements of those applications without the need for significant thrust to be produced by the core nozzle. However, for the

Table 59: Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the low technology level designs: Applications A and B.

Design Cycle Parameter	Bench 4.A	Bench 4.B
π_{Fan} at ADP	1.70	1.70
π_{HPC} at ADP	14.72	14.69
$\pi_{Overall}$ at ADP	31.08	31.16
T_4 at TKO, $^{\circ}R$	3,485.	3,485.
ER at ADP	1.25	1.24
$NChg_{LPT}$, $\%W_{25}$	6.00%	6.00%
95% CI Metric Value	Bench 4.A	Bench 4.B
F_N at TKO, lbf	23,864	23,905
F_N at TOC, lbf	6,680	6,683
HP Shaft Power at TKO, shp	40,600	40,735
W_{3R} at ADP, lbm/s	7.14	7.13
BPR at ADP	6.66	6.63
Fan Diameter, in	70.1	70.1
$T_{3,max}$, $^{\circ}R$	1,600.	1,602.
$T_{45,max}$, $^{\circ}R$	2,300.	2,300.
TSFC at CRZ, $lbm/lbf/hr$	0.6331	0.6332
Pod Weight, lbm	7,101	7,086

Table 60: Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the mid technology level designs: Applications C and D.

Design Cycle Parameter	Bench 4.C	Bench 4.D
π_{Fan} at ADP	1.70	1.70
π_{HPC} at ADP	16.00	16.00
$\pi_{Overall}$ at ADP	32.86	33.42
T_4 at TKO, $^{\circ}R$	3,739.	3,342.
ER at ADP	0.90	1.06
$NChg_{LPT}$, $\%W_{25}$	6.00%	0.00%
95% CI Metric Value	Bench 4.C	Bench 4.D
F_N at TKO, lbf	28,901	29,797
F_N at TOC, lbf	7,164	7,047
HP Shaft Power at TKO, shp	69,211	63,966
W_{3R} at ADP, lbm/s	8.63	8.32
BPR at ADP	3.97	5.06
Fan Diameter, in	64.2	68.1
$T_{3,max}$, $^{\circ}R$	1,750.	1,750.
$T_{45,max}$, $^{\circ}R$	2,232.	2,321.
TSFC at CRZ, $lbm/lbf/hr$	0.6887	0.6602
Pod Weight, lbm	6,558	5,998

Table 61: Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for the high technology level designs: Applications E, F, and G.

Design Cycle Parameter	Bench 4.E	Bench 4.F	Bench 4.G
π_{Fan} at ADP	1.70	1.70	1.63
π_{HPC} at ADP	16.00	16.00	15.10
$\pi_{Overall}$ at ADP	35.64	30.01	30.37
T_4 at TKO, $^{\circ}R$	3,455.	3,653.	3,318.
ER at ADP	0.90	0.90	0.90
$NChg_{LPT}$, $\%W_{25}$	6.00%	0.00%	0.00%
95% CI Metric Value	Bench 4.E	Bench 4.F	Bench 4.G
F_N at TKO, lbf	28,528	29,953	30,327
F_N at TOC, lbf	7,214	7,623	8,083
HP Shaft Power at TKO, shp	73,716	75,992	61,066
W_{3R} at ADP, lbm/s	8.18	9.41	9.09
BPR at ADP	3.69	3.52	5.42
Fan Diameter, in	64.2	61.3	70.0
$T_{3,max}$, $^{\circ}R$	1,800.	1,797.	1,692.
$T_{45,max}$, $^{\circ}R$	2,213.	2,292.	2,367.
TSFC at CRZ, $lbm/lbf/hr$	0.6869	0.7211	0.6687
Pod Weight, lbm	5,883	5,686	4,052

remainder of applications, the fan diameter constraints necessitate significant core nozzle thrust in addition to propulsor thrust in order to meet the requirements. This is shown by the low design Extraction Ratio selections that reserve core thermal energy for core nozzle thrust production. The relationship between the available amount of engine airflow set by the fan diameter constraint and the required amount of thrust significantly limited the bypass ratios attainable for Applications C - G, increasing the accompanying cruise TSFC levels. Significantly more core power was required of these applications, which required significantly more core flow, indicated by the HPC exit corrected flow levels (W_{3R}) listed in Table 60 and Table 61. In fact, comparing Applications B and F, whose thrust requirements are almost identical, Application F required 32% more core corrected flow in order to meet its required thrust levels with a 61 inch fan where Application B was able to use a 70 inch fan diameter. Even though Application F had the highest technology level projected for its later entry into the market, its fan diameter constraint caused it to have the largest core

size of all the benchmark engines.

$$ER = \frac{P_{t,19}}{P_{t,9}} = \frac{P_{t,BypNozzExit}}{P_{t,CoreNozzExit}} \quad (50)$$

Strict fan diameter constraints imposed on engines that require large amounts of thrust have heavy implications on the engine core power and corresponding flow requirements. This can cause significant concern when considering multiple applications of a single engine core definition. Granted, the present experiment allows for core inlet flow scaling and HPT overspeed at takeoff in order to increase the available power when needed for a particular application. However, the overall common core size, indicated by the design HPC exit corrected flow (W_{3R}), which is held constant across common core applications, will be driven by the engine application with the largest core power required to meet thrust requirements within the constraints imposed.

Identification of the limiting common core variant applications that cause the core definition to be necessarily oversized to meet its requirements is an essential capability when making common core design decisions. Otherwise, design margins such as core corrected flow margins may blindly be built into the core design in order to increase the likelihood of achieving later application power demands. Without accounting for and simulating projected applications when making the initial core design decisions, size margins may either not be enough or may be too large, causing unnecessary performance degradations throughout the entire engine program. Early identification of limiting variant applications also allows the designer to pose mitigation strategies to limit the amount of performance compromise made due to the limiting common core applications. For example, additional technology may be identified for advancement, aiming to reduce the uncertainty in its impacts and/or to improve its most likely impact levels. This would potentially allow for a more compact core design with less built-in design margins necessary to account for variability in the application's likely performance. Requirements may be selected to be relaxed in order to reduce the maximum power demands of the limiting applications. Identification of program-limiting applications may also convince decision makers to drop a proposed

common core application altogether in order to improve the overall performance of the remaining program applications, making them more competitive and possibly increasing their market demand. This type of mitigation exercise will be performed later in this experiment, demonstrating the enabled capabilities of the COMMENCE method and highlighting the features that designers can take advantage of when making initial engine program design trade-offs, down-selections, and core design decisions.

4.5.3.2 Utilization of Existing Surrogate Models

Just as with all of the previous experiments in this chapter, the surrogate models trained at the beginning of this chapter are again utilized for this final experiment. *Surrogate Set C* ($SSC_{Des,Prob}$) is again used to provide probabilistic estimates for the core defining designs engine under the low technology level uncertainty scenario. Each common core variant engine application is evaluated using *Surrogate Set D* ($SSD_{Var,Prob}$). These probabilistic surrogate models are functions of the design variables corresponding to both the core defining design engine and the common core variant. Also, the probabilistic models are functions of the beta distribution shape parameters for both the design and variant engines.

For each common core application, the uncertainty distribution shape parameter settings corresponding to the application's assumed uncertainty scenario are inputted and remain fixed throughout the initial design exploration of the common core application. For each common core variant application, *Surrogate Set D* is evaluated under the various uncertainty scenarios, exploring various design-variant design variable settings throughout the multiple application design space.

4.5.3.3 Core Defining Design Cycle Exploration

With the benchmark new centerline engine cycles established for each of the seven applications, exploration of the common core design space can now be performed, utilizing the *Surrogate Set D* probabilistic surrogate models that provide the desired confidence interval performance under each application's corresponding uncertainty scenario. Figure 94 contains a representation of the multiple application, multiple technology level common core design problem. As shown in the figure, dependencies exist between the core defining design

engine which establishes the common core definition. The resultant baseline core definition is applied to each of the seven variant engine applications. Also shown are the various technology level distributions being applied to different common core engine variant applications. This is enabled by the probabilistic surrogate models of variant performance being functions of the uncertainty distributions assumed present in the design problem. The dependencies allow variant design performance implications to be drawn relating design engine control variable settings to variant performance levels in terms of whether the common core applications meet all constraints imposed and whether the performance estimates compare well to each corresponding benchmark engine's likely performance levels.

Just as in the previous experiment, a search can be performed to find a feasible common core design space offering feasible common core design options for all applications. When considering a large set of applications or a wide range of capabilities, common core feasibility may not be attainable. If this is the case, identification of limiting requirements can be identified and mitigated. This mitigation process will be demonstrated after the initial core design is selected. Assuming that a feasible common core design region does exist, core defining engine design parameter settings can be explored. Whether through manual exploration of the continuous design and metric space as was done for the present experiment, rich sampling, or through optimization, a common core design region sought and identified that offers feasible solutions. Figure 95 contains a similar contour profile chart as was shown in the previous common core design problem, this time considering the seven engine applications presently considered. The design plane shown is that of the core defining design engine. Enabled by *Surrogate Set D*, all constraints imposed on all seven engine applications can be displayed on the core defining design plane of the design takeoff thrust and design HPC pressure ratio of the core defining design engine cycle. The first observation that can be made is that a feasible common core design space for all seven applications is present, however the region is very small. There is very little variation allowed in the core defining design engine cycle's control variable settings that allow for seven feasible engine variants. However, just as in the previous experiment all the constraints imposed are applied to 95% confidence interval performance estimates. This means that

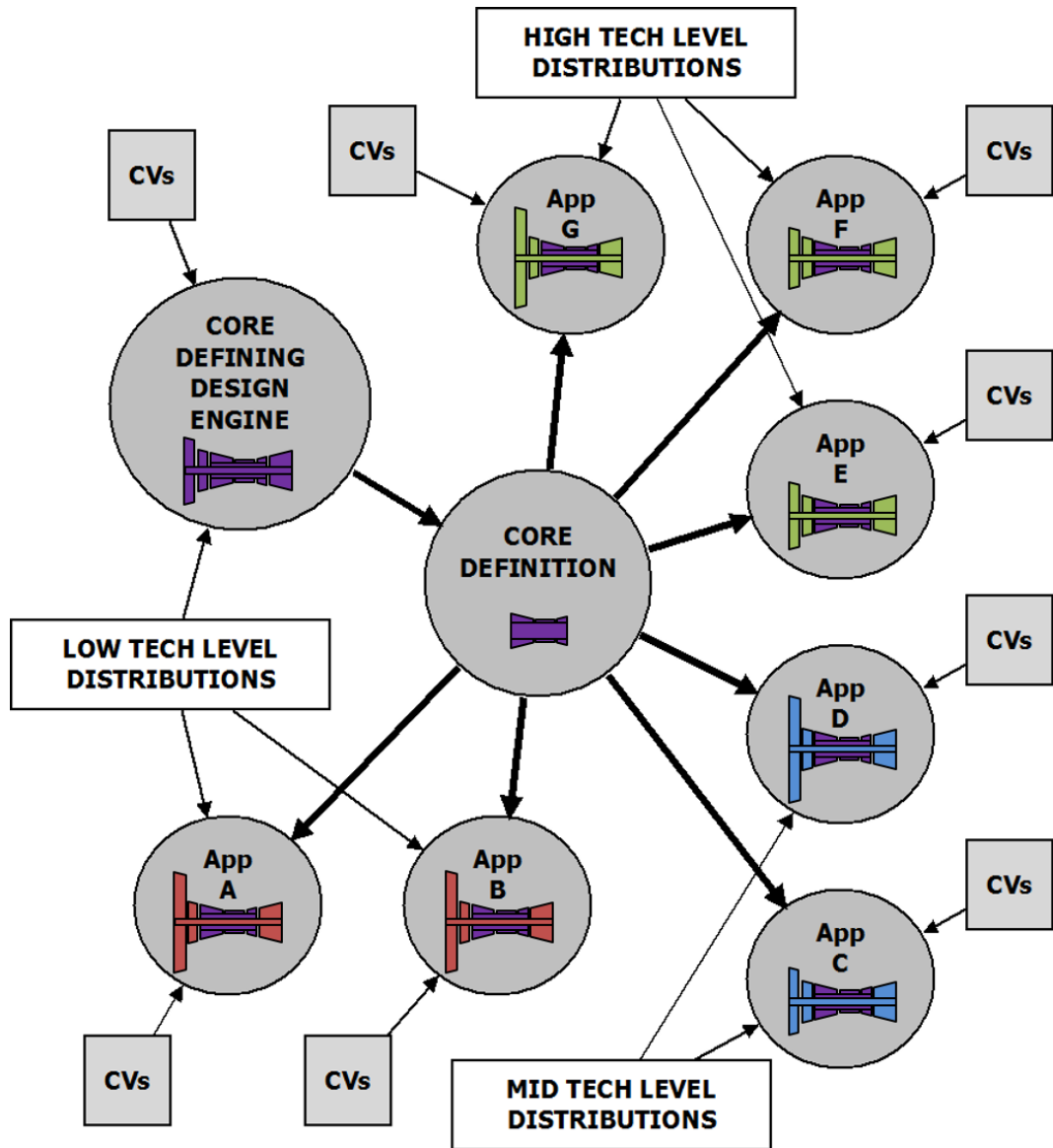


Figure 94: Experiment 4 - Representation of the multiple application, multiple technology level common core design problem, and the dependencies present in the problem.

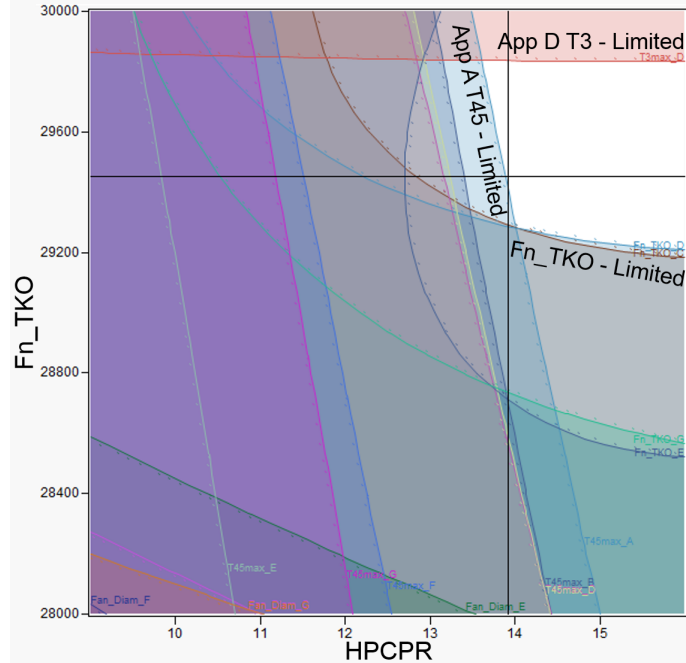


Figure 95: Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine takeoff thrust and HPC design pressure ratio during simultaneous exploration of core and variant engines.

the designer is still able to have 95% confidence in meeting all the constraints considered, even though the feasible design region for designs that have that accompanying high level of likelihood is very small.

The contour chart displayed in Figure 95 shows that for the particular set of fixed design parameter settings not shown in the chart, the limiting application constraints are the LPT inlet temperature constraint of Application A, and the takeoff thrust requirements of Applications C and D. The takeoff thrust requirements limit the core defining engine's takeoff thrust to remain near the maximum value considered, ensuring that ample core power is available to achieve the relatively high takeoff thrust required of the mid level technology applications. The LPT inlet temperature constraint of Application A is shown by the diagonal contour limiting the core defining cycle to a high HPC pressure ratio. Otherwise, a lower pressure ratio HPC would reduce the amount of extraction in the HPT of Application A, causing the LPT inlet pressure ratio constraint to be surpassed for the particular LPT cooling fraction of the application.

Another key core sizing parameter of the core defining design engine is the top of climb thrust for which the core defining engine is sized. The contour chart in Figure 96 displays the variant application constraints of the design plane of top of climb thrust and the HPC pressure ratio of the core defining engine cycle design. Again, a very small region of feasibility exists that has an accompanying 95% level of confidence in achieving feasibility across all seven common core variant applications. The small band of feasible top of climb thrust levels of the design engine is, at the current set of control variable settings across applications, restricted on the lower end by the takeoff thrust requirements of several applications, particularly Applications C, D, and E. The upper limit of feasible design engine top of climb thrust is limited by several application fan diameter constraints. Top of climb thrust is a key driver in the airflow available to the engine to produce thrust. Larger levels of top of climb thrust, apparent in the contour plot by the fan constraints becoming active, require a larger fan diameter for a given cycle. Conversely, decreased top of climb thrust for a given cycle decreases the sized airflow and the thrust achievable at other operational points while being able to hold the cycle at the inputted design settings. As shown in the previous chart, the LPT inlet temperature limit is holding the core design HPC pressure ratio to the higher end of the range considered.

One last constraint plot shown for the feasible common core design problem displays the core design OPR and core design HPC pressure ratio plane, and the various application constraints are again displayed on the design plane in Figure 97. An extremely small range of OPR designs permit a 95% likelihood of achieving feasible variant designs across the range of applications considered. Higher OPR designs would require more work to be performed on the core working fluid, decreasing the bypass ratio of the variant engines and having takeoff thrust constraints become active for several variant applications. Lower core design OPR levels would not maintain the core power necessary to aid in the production of thrust by the core nozzle, which would require variant engines to have larger fan diameters in order to meet thrust. These larger fans, however, would surpass the limits imposed for several applications.

In order to simultaneously view the three core defining engine design planes that most

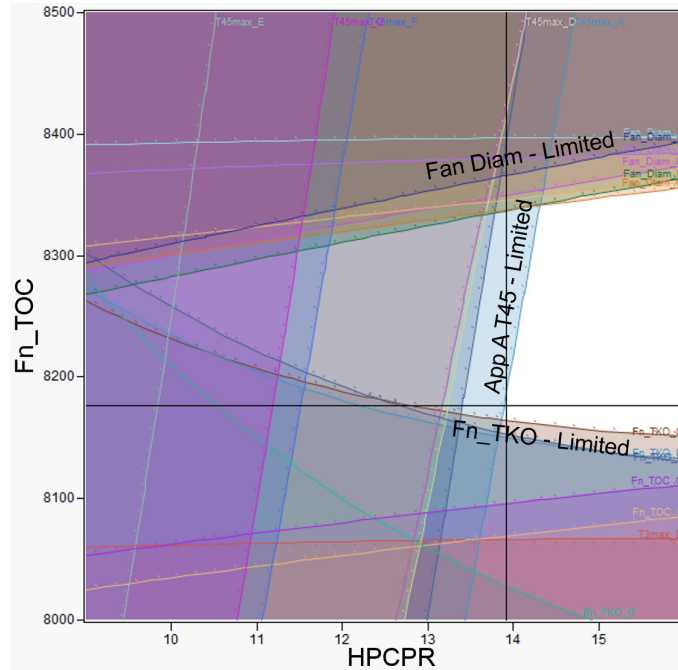


Figure 96: Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine top-of-climb thrust and HPC design pressure ratio during simultaneous exploration of core and variant engines.

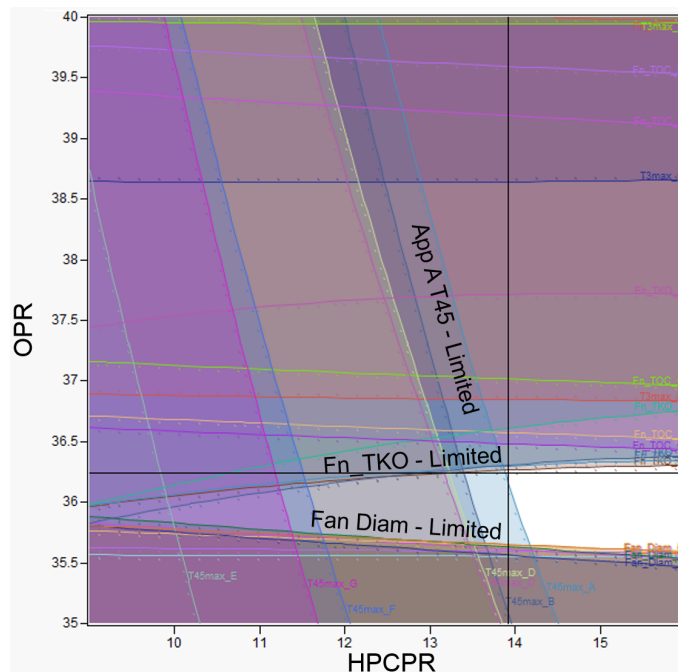


Figure 97: Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio during simultaneous exploration of core and variant engines.

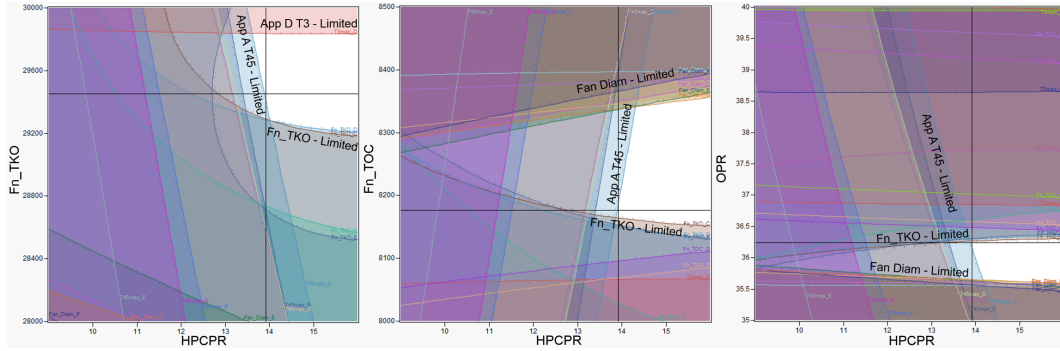


Figure 98: Experiment 4 - Common core design exploration: Side-by-Side contour profilers identifying the feasible common core design space across the key contributing design planes of the core defining design engine.

impact common core variant feasibility across applications, Figure 98 displays all three previously displayed contour profilers side-by-side.

For the seven common core applications considered, each having a unique set of requirements and constraints, a feasible common core design was identified. The core definition established with the core defining design engine, when applied to each of the variant engine applications, had at least a 95% likelihood of meeting all of the requirements and constraints for all applications. The selected cycle parameter values of the core defining engine cycle are listed in Table 62 along with some engine core characteristics of interest. As was shown in the constraint diagrams above, in order to ensure feasibility, the core has a HPC with a design pressure ratio on the high end of the range considered, ensuring that the common core applications are able to generate enough power for all applications. Keep in mind that the variant designs will be able to further increase the HPC design pressure ratios when scaling the core working fluid by flaring or zero-staging the HPC. As the design flow increased for a variant application, the design pressure ratio is increased in order to maintain consistent HPC exit corrected flow across all applications. Again, maintaining HPC exit corrected flow at the engine's aerodynamic design point (ADP) ensures a geometrically common core design when simulating variant designs, even when allowing for core flow increases when needed.

Another observation of the core defining engine cycle selection is that the nonchargeable

Table 62: Experiment 4 - Core Defining Engine Cycle Selection.

Core Defining Cycle Parameter	Selected Value
π_{Fan} at ADP	1.70
π_{HPC} at ADP	13.92
$\pi_{Overall}$ at ADP	36.24
T_4 at TKO, °R	3,342.
ER at ADP	1.05
$NChg_{LPT}$, % W_{25}	6.00%
F_N at TKO, <i>lbf</i>	29,450.
F_N at TOC, <i>lbf</i>	8,177.
HP Shaft Power at ADP, <i>shp</i>	16,143
HP Shaft Power at TKO, <i>shp</i>	48,092
W_{3R} at ADP, <i>lbm/s</i>	7.49

LPT cooling fraction ($NChg_{LPT}$ in the table) is at its maximum value allowed, bleeding 6% of the core flow at the HPC interstage bleed port. This allows for the maximum amount of thermal energy to exit the HPT and have the high energy flow cooled as much as possible before entering the LPT. The relatively low overall pressure ratio selected for the design core not only remains insensitive to any HPC exit temperature limits by remaining far away from the constraints imposed, but this also indicates that a relatively high core size is desired. The higher the OPR, the more compact the core and the lower the HPC exit corrected flow. The lower the OPR, the less amount of total compression is performed, requiring larger flow areas throughout the core components. This combined with a high pressure ratio HPC enables common core applications to have significant amounts of power growth. The core definition established by the current design selection produces over 48,000 horsepower at 100% corrected speed during takeoff, and over 16,000 horsepower at its aerodynamic design point when at full speed. The core size parameter, the HPC exit corrected flow (W_{3R} at ADP) has a design value of 7.49 *lbm/s*. Compared to the benchmark engine cycle selections for the seven applications considered, this value as well as the HP shaft power (or core power) is on the lower end of values found for the seven benchmark engine cycles selected. This suggests that significant core power growth is desired for the common core variant selections, whether through scaling of the core working fluid by flaring the HPC

or through overspeeding of the HP shaft during takeoff. This will be confirmed during the discussion of the variant design space explorations and selections made for the common core applications.

At this point, a feasible core design region has been identified and a promising candidate core definition has been selected. The designer could move on to applying the core definition to the common core applications and exploring the common core variant design spaces. However, in order to demonstrate the capabilities of the COMMENCE method in providing the designer with the ability to easily change the assumptions implicit in the probabilistic estimates of common core performance, additional considerations will now be made during further exploration of the feasible common core design. Consideration of additional technology as well as relaxation of a particular limiting thrust requirement will be made in order to improve the overall common core engine program by allowing the core definition to be improved while still having a high confidence in achieving feasible variant solutions for all applications considered.

4.5.3.4 Technology Push and Requirement Relaxation to Improve Feasible Core Definition

When performing the initial common core design space exploration, it was shown that the feasible region of the design space was constrained by several applications. The constraint diagram shown in Figure 99 displays the core definition's OPR / HPCPR design plane. The cross-hairs are located at the feasible core defining cycle selection settings previously listed in Table 62. Also present in the plot are contours of constant HPC exit corrected flow, indicating that increases in design OPR and reduction in the HPC pressure ratio cause the core corrected flow to decrease. As mentioned before, a smaller value of HPC exit corrected flow corresponds to a smaller, more compact engine core. Therefore, the desired region within the context of the core size is in this region of decreased core flow. However, there are several common core applications whose constraints limit how far the core definition can move in this desirable direction.

The COMMENCE method, by enabling simultaneous consideration of initial and future engine applications of interest and their corresponding scenario assumptions, has allowed the

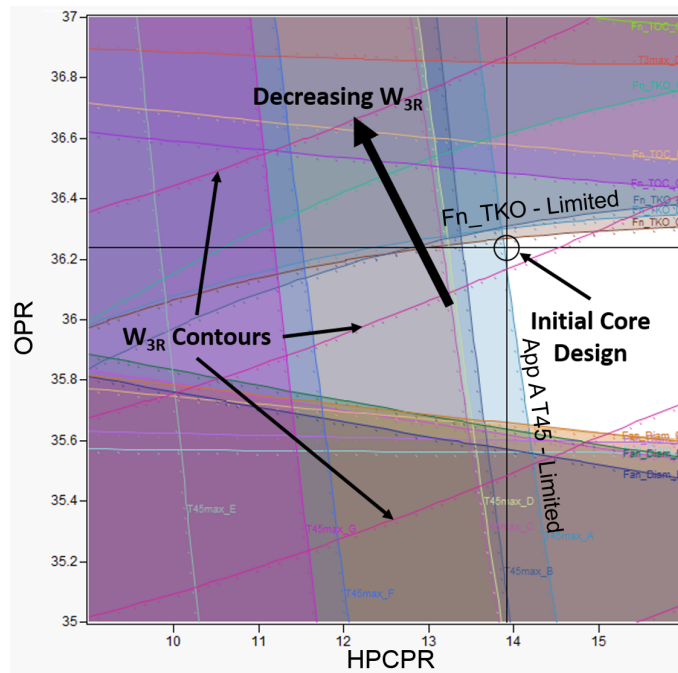


Figure 99: Experiment 4 - Common core design exploration: Constraints defining the feasible region of design engine OPR and HPC design pressure ratio during simultaneous exploration of core and variant engines, labeled to identify contours of constant HPC exit corrected flow and the favorable design region that would allow for a more compact common core definition.

Table 63: Experiment 4 - Core Performance Mitigation - Adjustments made to imposed limits of core design limiting applications in order to improve overall engine program performance.

Engine App	Limit	Description of Limit	Old Limit Value	New Limit Value
4.A	$T_{45,max}$	Maximum LPT inlet temperature	2,300 °R	2,350 °R
4.B	$T_{45,max}$	Maximum LPT inlet temperature	2,300 °R	2,350 °R
4.E	F_N at TKO	Takeoff thrust requirement	27,500 <i>lbf</i>	27,000 <i>lbf</i>

designer to identify the application constraints which are holding the common core definition from further program performance improvement. In an attempt to allow for a smaller core definition, two temperature limits and one thrust requirement were modified. The LPT inlet temperature constraints of Applications A and B were increased by 50 degrees Fahrenheit, increasing the limiting value to 2,350 °R which is equivalent to the mid-technology level value applied to Applications C and D. This represents a technology push on the LPT, which may be accomplished by either a material improvement or a film cooling effectiveness improvement. Although this technology push would require additional development, the capability was assumed to be achieved for the mid-technology level engine applications nevertheless. This tech push for the initial applications would require the development schedule to be expedited. Another limit adjustment that has been made to enable a more compact common core definition is the relaxation of the takeoff thrust requirement of variant Application E, reducing its required level by 500 *lbf*. These modifications have been listed in Table 63.

Before performing subsequent common core design space explorations with the modified limits imposed, benchmark engines needed to be re-established for Applications A and B with the new LPT inlet temperature limits. Again, the procedure of Experiment 1b was exactly following, and *Surrogate Set C* was again used to evaluate candidate benchmark designs. The new benchmark selections can be found in Table 64. Compared to the original

Table 64: Experiment 4 - New Centerline Benchmark Engine Cycle Selections and corresponding 95% confidence interval performance levels for Applications A and B after Core Performance Mitigation.

Design Cycle Parameter	Benchmark 4.Aadv	Benchmark 4.Badv
π_{Fan} at ADP	1.70	1.69
π_{HPC} at ADP	11.46	10.64
$\pi_{Overall}$ at ADP	31.77	31.77
T_4 at TKO, °R	3,439.	3,326.
ER at ADP	1.25	1.25
$NChg_{LPT}$, % W_{25}	6.00%	6.00%
95% CI Metric Value	Benchmark 4.Aadv	Benchmark 4.Badv
F_N at TKO, lbf	23,389	25,029
F_N at TOC, lbf	6,666	6,633
HP Shaft Power at TKO, shp	36,145	41,145
W_{3R} at ADP, lbm/s	6.87	7.36
BPR at ADP	6.75	6.31
Fan Diameter, in	70.0	70.0
$T_{3,max}$, °R	1,594.	1,632.
$T_{45,max}$, °R	2,350.	2,349.
TSFC at CRZ, $lbm/lbf/hr$	0.6296	0.6332
Pod Weight, lbm	7,083	6,994

benchmark cycles for these applications, the increase in the LPT inlet temperature limit allowed for the cruise TSFC and pod weight of both applications to be slightly improved. This was accomplished by unloading the core and slightly increasing the design OPR of the engines. The compression work to achieve the overall pressure ratio was allowed to be performed more by the LP components compared to the previous benchmark designs. The increase in the allowed LPT inlet temperature permitted less expansion to be done in the HPT, allowing for more compact, higher thermal energy (higher temperature) flow to enter the LPT. This warranted the more efficient LP compression components, namely the booster, to have more of the burden of pressurizing the core flow, which both improved the TSFC of the engines and also had the advantage of reducing the engine weight by allowing for a higher OPR design.

Now that the benchmark engines have been updated to reflect the technology limit push for the two initial engine program applications, exploration of the common core design space

can be performed again. In reality, the modifications made did not require any additional simulations. The limits imposed on the continuous design space were dynamically updated to reflect the modifications made. Since this exploration is done by taking advantage of the continuous probabilistic surrogate models, the updates can be instantaneously made on the charts previously shown, and the impacts of the changes on the feasible design space can again be identified. Figure 100 shows the core OPR / HPCPR design plane and the feasible region before and after making the limit modifications. Note that the single constraint in the *AFTER* contour chart that is being violated is the original takeoff thrust requirement of Application E before being relaxed. This shows how the thrust requirement was preventing the common core definition from having a lower core size with corresponding design levels of HPC exit corrected flow. After making the imposed limit modifications and finding the feasible common core design region, the comparison of contour charts shows the shift in the feasible region to lower core design OPR levels. Although this seems to mean that the core size would be larger since the direction of core flow reduction is in the direction of higher OPR levels, the new core definition is actually slightly smaller than the definition previously selected.

Table 65 contains the core defining cycles selected before and after making the imposed limit modifications. Both design cycle selections are very similar. The notable differences in the design cycle selections are in the HPC pressure ratio, the design OPR, and the extraction ratio of the core defining engine cycle. The updated core defining engine cycle was able to have slightly lower values of its design OPR and HPC pressure ratio while still being able to offer feasible engine variant applications for all seven common core variants considered. The major cycle difference is present in the higher extraction ratio that the core defining cycle was able to have after modifying the imposed limits. As mentioned earlier, the higher extraction ratio is an indicator that the design engine and its common core variants are less constrained by the fan diameter limits while still being able to achieve its takeoff thrust requirements. This shows that the common core variant designs will be able to reach their takeoff thrust levels primarily with their propulsors and bypass nozzles, without the need for significant core nozzle thrust, making the variant cycles more efficient. This is a good

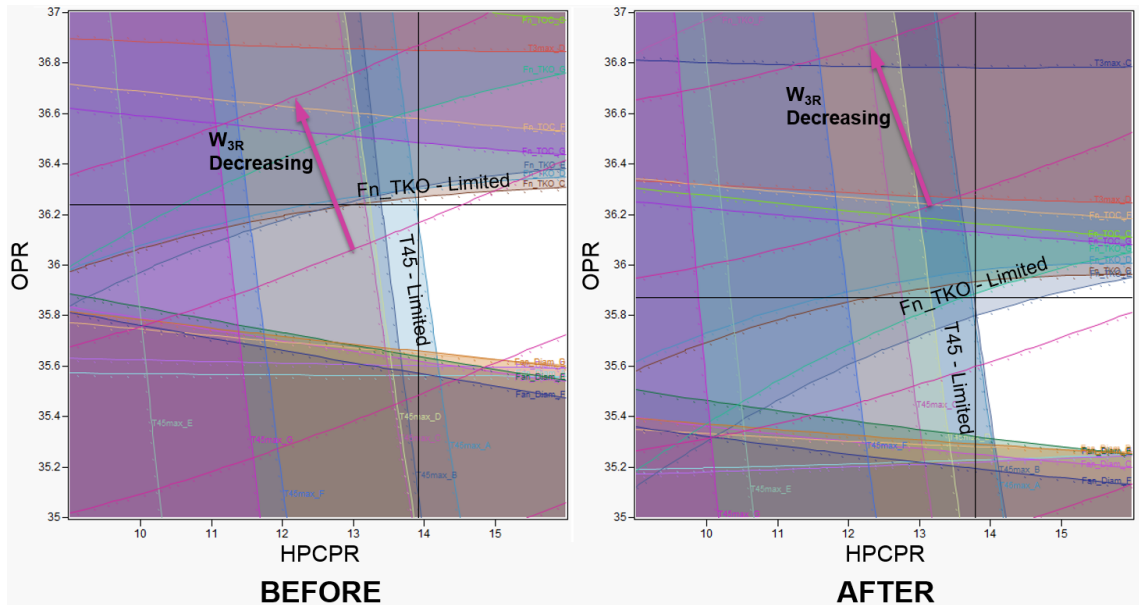


Figure 100: Experiment 4 - Common core design exploration: Constraint contours and design core selection settings before and after Core Performance Mitigation.

indication that the new core definition will be less constrained when being utilized for the various engine applications considered.

Comparing the engine core characteristics before and after modifying the core limiting constraints, Table 65 shows that the new core definition has a slightly smaller design core corrected flow (W_{3R}), a reduction of about 0.5%. A more significant difference between the core definition selections is the reduction in maximum required core takeoff power. By making the early application LPT technology push and relaxing the takeoff thrust requirement of Application E, the core definition was able to reduce its design takeoff power by 3.5%, which is a significant reduction in the maximum power margin necessary to provide viable common core variant engines for all applications considered. The core design power at ADP was also reduced by 2.1%, allowing for a smaller core to be employed across the engine program. A quantitative measure can now be attributed to the LPT technology push and relaxation of takeoff thrust for Application E. The designer can state with a high level of confidence: *By advancing our early LPT technology level so that the maximum LPT inlet*

Table 65: Experiment 4 - Common Core Design Cycle Selections before and after Core Performance Mitigation.

Core Defining Cycle Parameter	Before	After
π_{Fan} at ADP	1.70	1.70
π_{HPC} at ADP	13.92	13.78
$\pi_{Overall}$ at ADP	36.24	35.87
T_4 at TKO, $^{\circ}R$	3,342.	3,342.
ER at ADP	1.05	1.20
$NChg_{LPT}$, $\%W_{25}$	6.00%	6.00%
F_N at TKO, lbf	29,450.	29,450.
F_N at TOC, lbf	8,177.	8,177.
HP Shaft Power at ADP, shp	16,143	15,810
HP Shaft Power at TKO, shp	48,092	46,409
W_{3R} at ADP, lbm/s	7.49	7.46

temperature can be increased by 50 °F and also by relaxing the thrust requirement of Application E, we are able to reduce the maximum takeoff power of the common core definition by 3.5% while still having a 95% likelihood of meeting all requirements and constraints imposed on the seven applications considered in our engine program.

This exercise of identifying regions for technology development and also considering the relaxation of particular assumed requirements shows a key advantage of the COMMENCE method. It's ability to simultaneously simulate any number of applications, each having their own requirements sets and unique uncertainty assumptions, allows the designer to determine quantitatively the implications of initial core design decisions that are made. It allows the decision makers to make down selections while having a high level of confidence that the selection will be able to provide viable design solutions to all engine applications considered. As shown in this exercise, it also enables the designer to consider alternative means of improving the core selection, whether it be through additional technology infusion that initially was not identified as being needed, or by considering whether a certain assumed requirement is really necessary to be exactly met.

A key advantage of the method is that the explorations performed for the core design have been done in a manner very similar to what a cycle design engineer would be familiar

with, while providing the designer with vastly more information when doing so. More upfront work is of course necessary in order to account for the assumed sources of uncertainty, and the probabilistic surrogate models are needed to implicitly account for the uncertainty distributions. Once this minimal amount upfront work has been done, the designer is able to apply any number of scenarios to any number of proposed engine applications. Highly efficient explorations of the robust design space of a possible common core engine program can be easily performed. More informed decisions can then be made, having a high level of associated confidence that the decisions will result in designs with the intended capabilities.

4.5.3.5 Design Explorations of Common Core Variant Applications

For each common core engine application, exploration of the variant design space is performed in order to search for a feasible region and make a selection that meets all of the application requirements while also performing well compared to the associated benchmark cycle design. In reality, these variant design space explorations are performed simultaneously while exploring the core defining cycle design. In order to determine whether a candidate core definition is able to offer feasible variant applications, the variant designs must be concurrently explored. This requirement of the multi-application design problem is due to the fact that all engine applications utilize the same core definition, whether modifications are made to it or not.

This again highlights a challenge when considering a large number of variant applications. For the present seven application common core design problem, the odds of a random sample of the multi-application design space having feasible solutions for all applications is nearly zero. Particularly for such a design problem where the core definition is applied to a new engine application with new requirements, the likelihood is a single common core application being feasible is very low, as previously shown in the constraint plots with very small feasible regions. Figure 101 shows the exponential decay of the probability of a multiple application data sample being feasible as the number of design applications increases. The axis labeled P_{App} indicates the probability of a single application data sample being feasible. Needless to say, with the current seven application common core design problem,

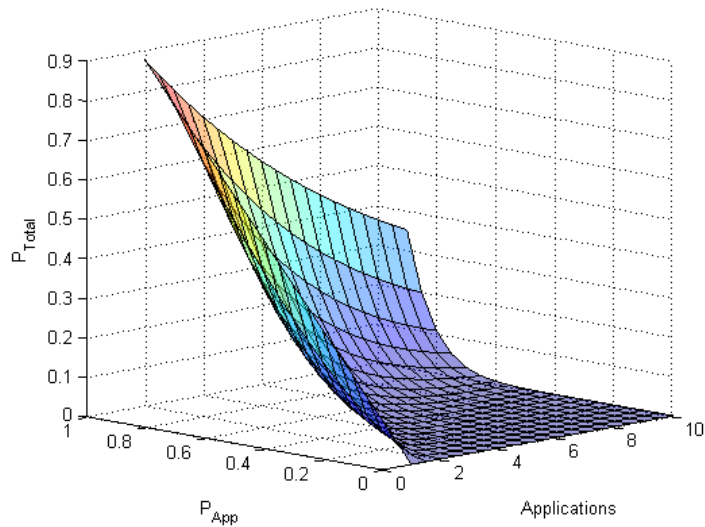


Figure 101: Surface plot showing the exponential decay of the likelihood of a multiple application sample being feasible as the number of applications considered increases for various levels of the probability of a single application being feasible.

which has very small regions of design feasibility, the likelihood of a random sample being feasible for all seven applications is essentially zero.

Traditional robust design methods that depend on discrete sampling of a continuous design space would not be able to perform such a study without an extremely high computational budget. The COMMENCE method does not depend on discrete samples for candidate design identification. The continuous probabilistic surrogate models of likely variant performance offer the designer the ability to search throughout the continuous multi-application design space for a feasible design region. This allows any computational budget to be allocated to the construction of the probabilistic surrogate models of likely common core variant performance levels in addition to the deterministic metamodels representing the physics based engine model.

For each engine variant design space exploration, the control variables and corresponding ranges considered are found in Figure 19. Constraints are imposed on each common core application's design space, eliminating regions where the constraints are violated or the requirements cannot be met. As described in the previous experiment, the power setting

Table 19 Revisited: Control variables and corresponding ranges explored for the present common core variant engine design explorations.

Variable	Units	Min	Max
$\pi_{Fan,Var}$ @ ADP	-	1.3	1.7
$\pi_{Overall,Var}$ @ ADP	-	30.0	55.0
$T_{4,Var}$ @ TKO	$^{\circ}R$	3,300	3,750
ER_{Var} @ ADP	-	0.90	1.25
$NChg_{LPT,Var}$	% W_{25}	0.00	0.06
sW_{25R} @ ADP	-	1.0	1.2
$N_{C,HPT,Var}$ @ TKO	%	100.	105.

control variable for the engine variant design was selected to increase the likelihood of NPSS model convergence. For example, instead of inputting a desired takeoff thrust level and having the thermodynamic engine model run to a thrust that may not be achievable, the amount of HPT over-speed ($N_{C,HPT,Var}$) is used as a control variable. The cycle model can most likely run to a HPT corrected speed, requiring a possible vane reset in order to do so if the flow becomes choked at a lower speed. Additionally, to provide the otherwise fixed engine core the ability to have an increase in core working fluid, the HPC inlet corrected flow is allowed to scale by an inputted amount (sW_{25R}). For each application, the feasible common core variant design space is identified, and the limiting constraints are shown that limit the application from improving its performance beyond the levels achieved by the cycle currently selected.

For each common core application considered, four constraint plots are shown. Two plots are shown representing the design space before the technology push of Applications A and B LPT inlet temperature and relaxation of the takeoff thrust requirement of Application E, and the same two design plane constraint plots are shown after making this modification. All variant constraint plots to be shown will display contours of constant cruise TSFC, allowing the reader to see which regions of the design space allow for TSFC improvement. The first variant design variable plane shown for each application displays the HPT corrected speed at takeoff and the variant design fan pressure ratio. The second variant design plane shows the impacts of the HPT corrected speed at takeoff as the inputted amount of HPC inlet

flow changes.

Figure 102 displays the feasible variant design region of Application A before and after the LPT technology push and relaxation of the Application E takeoff thrust requirement. The contours show that the most desirable TSFC levels are achieved with designs having low variant design fan pressure ratios ($\pi_{Fan,Var}$), minimal amounts of HPT over-speeding ($N_{C,HPT,Var}$), and minimal HPC inlet flow scaling (sW_{25R}). For Application A, the LPT inlet temperature limit requires an amount of HPT over-speed. By over-speeding the core at takeoff, which could be achieved with a vane reset of the HPT, more core work is performed. Remember that when analyzing constraint diagrams that the control variables not shown are all fixed, only the two control variables shown on the axes of the diagram can be varied. Therefore, when more core work is performed when over-speeding the HPT at takeoff, the combustor exit temperature (T_4) is fixed. This increase in core work for a fixed T_4 reduces the LPT inlet temperature (T_{45}), showing why the constraint becomes active at lower levels of HPT corrected speed at takeoff. The other constraint limiting Application A common core variant design is the fan diameter constraint, which limits the variant to higher design FPR levels. The constraint diagrams on the bottom row of plots in Figure 102 show that the Application A common core variant is able to meet all requirements and constraints without the need to scale the ADP design core flow, meaning that minimal core design modifications would be required for this variant application.

The feasible variant design space for common core Application B, shown in Figure 103, is very similar to the feasible space shown for Application A. Again, the fan diameter constraint and LPT inlet temperature limit restricts the variant design to high FPR levels and requires the takeoff HPT corrected speed to be greater than 100%. No HPC flow scaling is necessary either. Comparing the constraint diagrams labeled *Before* and *After*, both Applications A and B have notable feasible design improvements through the LPT technology push and Application E thrust requirement relaxation. By pushing the allowed LPT inlet temperature by 50 °F, less over-speeding of the HPT at takeoff is necessary, allowing the sizing of the LP system and HPT vane reset to be more favorable in terms of the cycle efficiency when operating at cruise conditions.

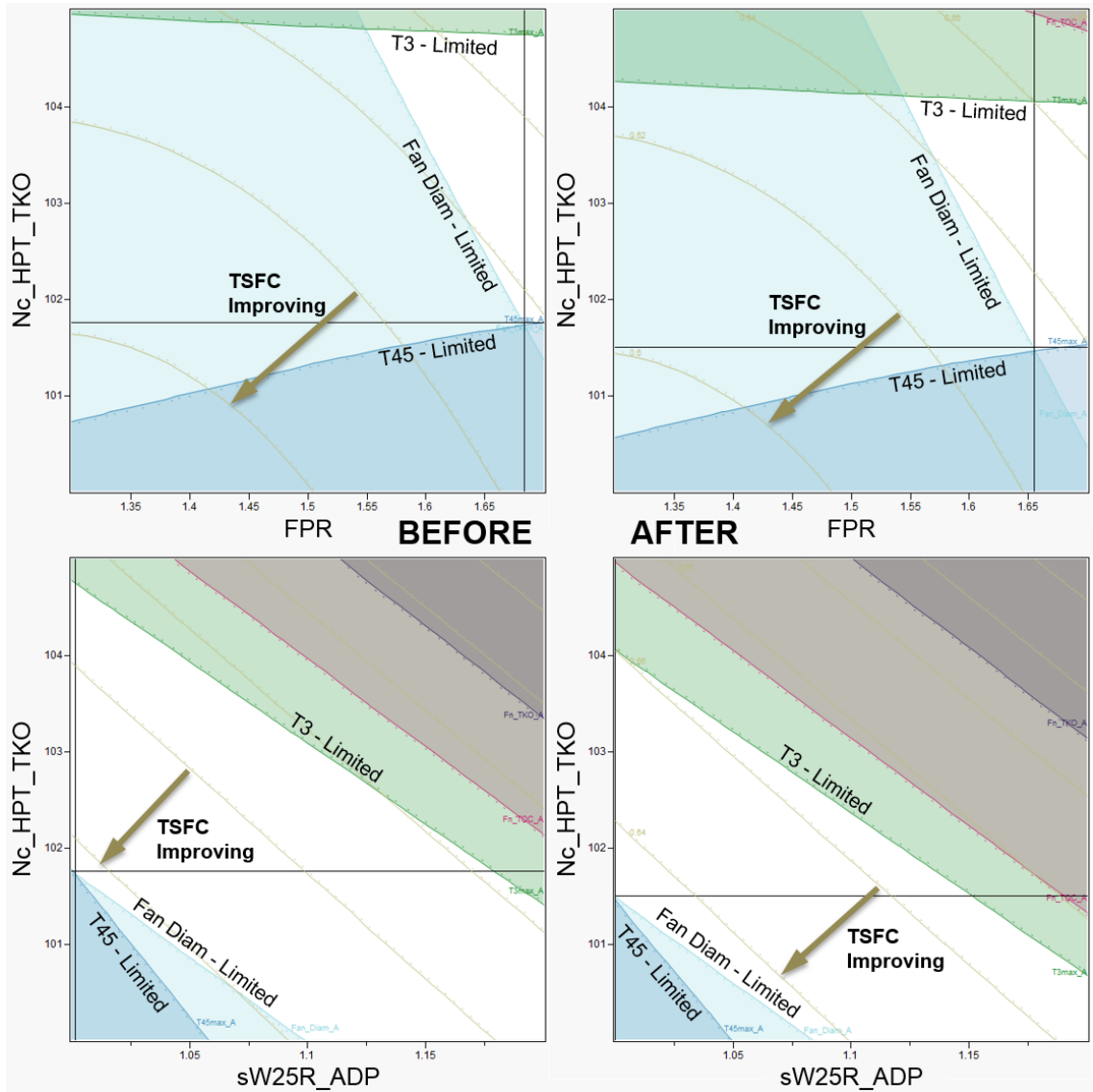


Figure 102: Experiment 4 - Common Core, Variant A Design Exploration: Constraint contours and Variant A design selection settings before and after Core Performance Mitigation.

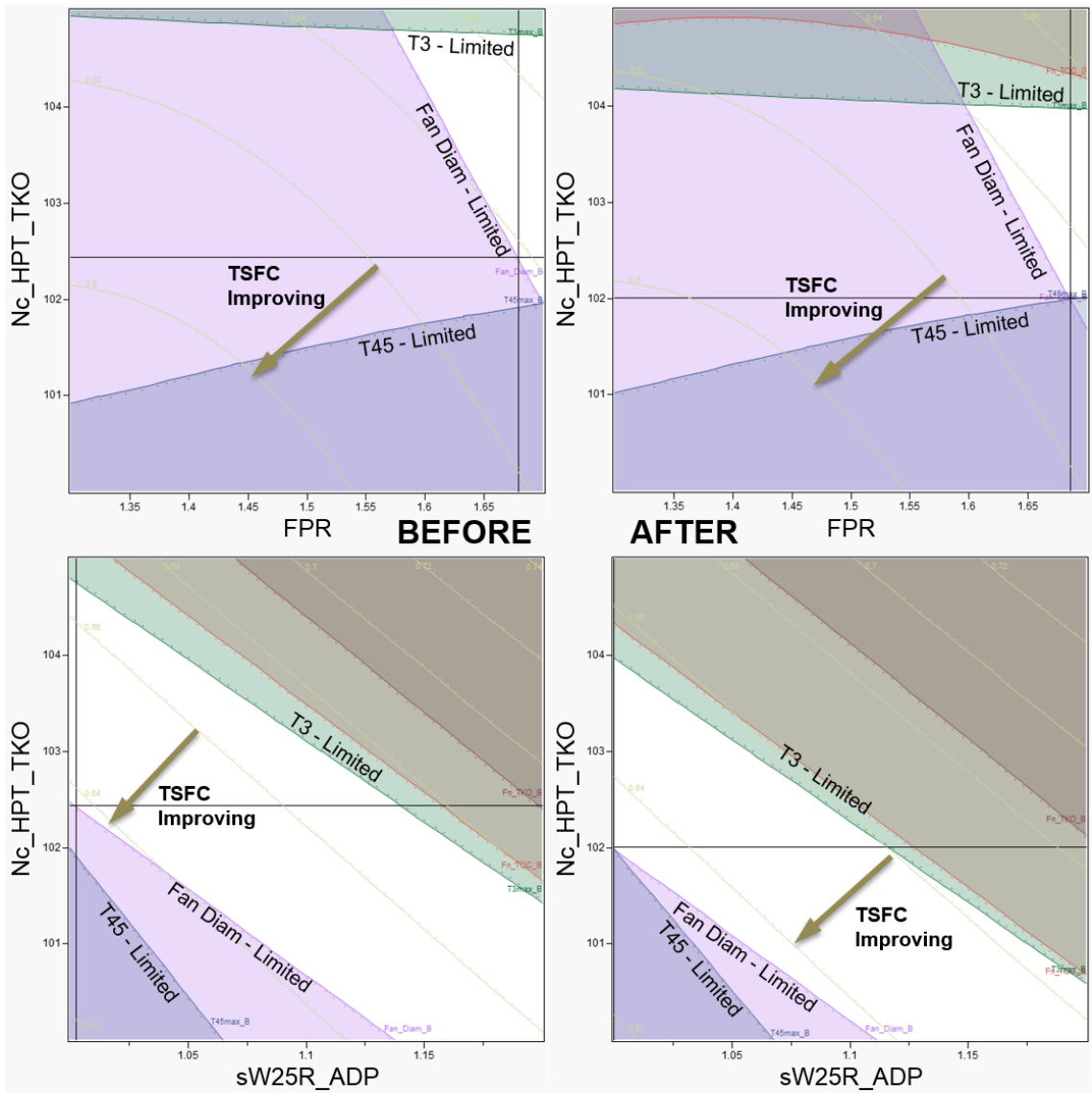


Figure 103: Experiment 4 - Common Core, Variant B Design Exploration: Constraint contours and Variant B design selection settings before and after Core Performance Mitigation.

The common core variant design cycle selections for the initial program applications, both with the low technology level assumptions applied, are listed in Table 66 along with their 95% likely performance estimates. These selections correspond to the core design selection made before modifying the LPT inlet temperature limits and takeoff thrust relaxation, whose selected settings accompany the selected variant application input values in the table. Comparing the variant cycle input selections to their corresponding benchmark engine cycles found in Table 59, the cycles are very comparable to each other. However, the core corrected flow at ADP (W_{3R} is referred to as core corrected flow) and the HP shaft power at takeoff are significantly greater for the common core variant engines. This shows that the common core definition is over-sized for these initial program applications, reserving flow and power capabilities for later engine variants. This extra built-in flow and power margin of the core causes degradations in the cruise TSFC levels of the variant designs, degrading them 0.57% and 0.77% for Applications A and B, respectively. These variant engines do however have improvements in the pod weight compared to the benchmark designs.

It must be noted that optimization of the benchmark engines and the multiple application common core designs were not performed. Therefore, it is likely that the achievable performance levels of feasible benchmark and common core designs can be improved. During the explorations performed in the present work, feasibility was the first goal, with additional input setting adjustment made manually to improve the likely performance as much as possible while also remaining in the feasible regions of the spaces explored. Having the common core variant designs in the regime of performance levels that the benchmark designs lie is a good indicator that the common core applications would be viable candidates that would likely have unit sales. Particularly since the development of the core definition can be distributed across the remainder of the common core applications, the slight performance degradations when going from a new centerline engine to a common core application would be likely be outweighed by the per unit cost savings.

Moving on to the mid term common core program applications, the constraint diagrams for variant Application C are displayed in Figure 104. From the design space exploration for this application, and as will be shown to be similar to the remainder of applications, the

Table 66: Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications A and B.

Design Engine Cycle Parameter	Core Defining Cycle Value	
π_{Fan} at ADP	1.70	
π_{HPC} at ADP	13.92	
$\pi_{Overall}$ at ADP	36.24	
T_4 at TKO, $^{\circ}R$	3,342.	
ER at ADP	1.05	
$NChg_{LPT}$, $\%W_{25}$	6.00%	
F_N at TKO, lbf	29,450.	
F_N at TOC, lbf	8,177.	
Application Requirement	App A	App B
Takeoff Thrust, lbf	22,000	24,000
Top of Climb Thrust, lbf	5,400	5,800
Max Fan Diam, in	70	70
Variant Design Cycle Parameter	Variant 4.A	Variant 4.B
π_{Fan} at ADP	1.68	1.68
$\pi_{Overall}$ at ADP	30.78	31.76
T_4 at TKO, $^{\circ}R$	3,749.	3,326.
ER at ADP	1.19	1.17
$NChg_{LPT}$, $\%W_{25}$	6.00%	6.00%
sW_{25R} at ADP	1.002	1.003
$N_{C,HPT,Var}$ at TKO	101.8	102.4
95% CI Metric Value	Variant 4.A	Variant 4.B
F_N at TKO, lbf	26,419	26,955
F_N at TOC, lbf	6,696	6,779
HP Shaft Power at TKO, shp	44,620	46,853
W_{3R} at ADP, lbm/s	7.49	7.49
BPR at ADP	6.01	5.85
Fan Diameter, in	70.0	70.0
$T_{3,max}$, $^{\circ}R$	1,614.	1,633.
$T_{45,max}$, $^{\circ}R$	2,300.	2,293.
TSFC at CRZ, $lbm/lbf/hr$	0.6367	0.6381
Pod Weight, lbm	6,690	6,253

feasible design space is very small. A small feasible region for Application C, with its 64 *inch* fan diameter requirement, is found with a maximum FPR fan design with significant takeoff corrected speed increases necessary to meet the takeoff thrust requirement while staying below the fan diameter constraint. Also keep in mind that installation requirements uncertainty are also being accounted for and designed for. Both the horsepower extraction and customer air bleed requirements have been increased for these mid level entry applications, and the variability in the requirements have been increased due to the larger uncertainty in the requirements of the mid term common core applications. Therefore, in order to ensure a 95% likelihood of meeting its thrust requirements and the constraints imposed, built-in margins account for the chance of the installations varying from the expected amount. Also worth noting for Application C is that the core selection made in response to the technology push and requirement relaxation made and reflected in the constraint plots on the right hand side of the figure show a trade in core flow scaling for more over-speeding of the HPT at takeoff.

The other mid-term common core program application, Application D, requires the maximum amount of takeoff thrust required of the program along with its later family member, Application G. The constraint diagrams found in Figure 105 shows a minute feasible space almost identical to what was shown for Application C. Again, the feasible space is highly restricted by the fan diameter constraint and takeoff thrust requirement, requiring a maximum FPR fan and HPT over-speeding at takeoff in order to achieve the required thrust with a 70 *inch* fan. Again, the core is pushed to its speed limit allowed and fan to its maximum compression in order to ensure a 95% likelihood of meeting its requirements and constraints while under the assumed technology uncertainty and more significantly the uncertainty of the increased power and bleed extraction requirements.

Table 67 contains the common core variant design selections and corresponding likely performance estimates for the mid term program releases, Applications C and D. Comparing the common core variant designs to their corresponding benchmark engines found in Table 60, it is easy to tell that the core definition is hard pressed to meet the requirements of these mid term applications. Significant core nozzle thrust was necessary for these

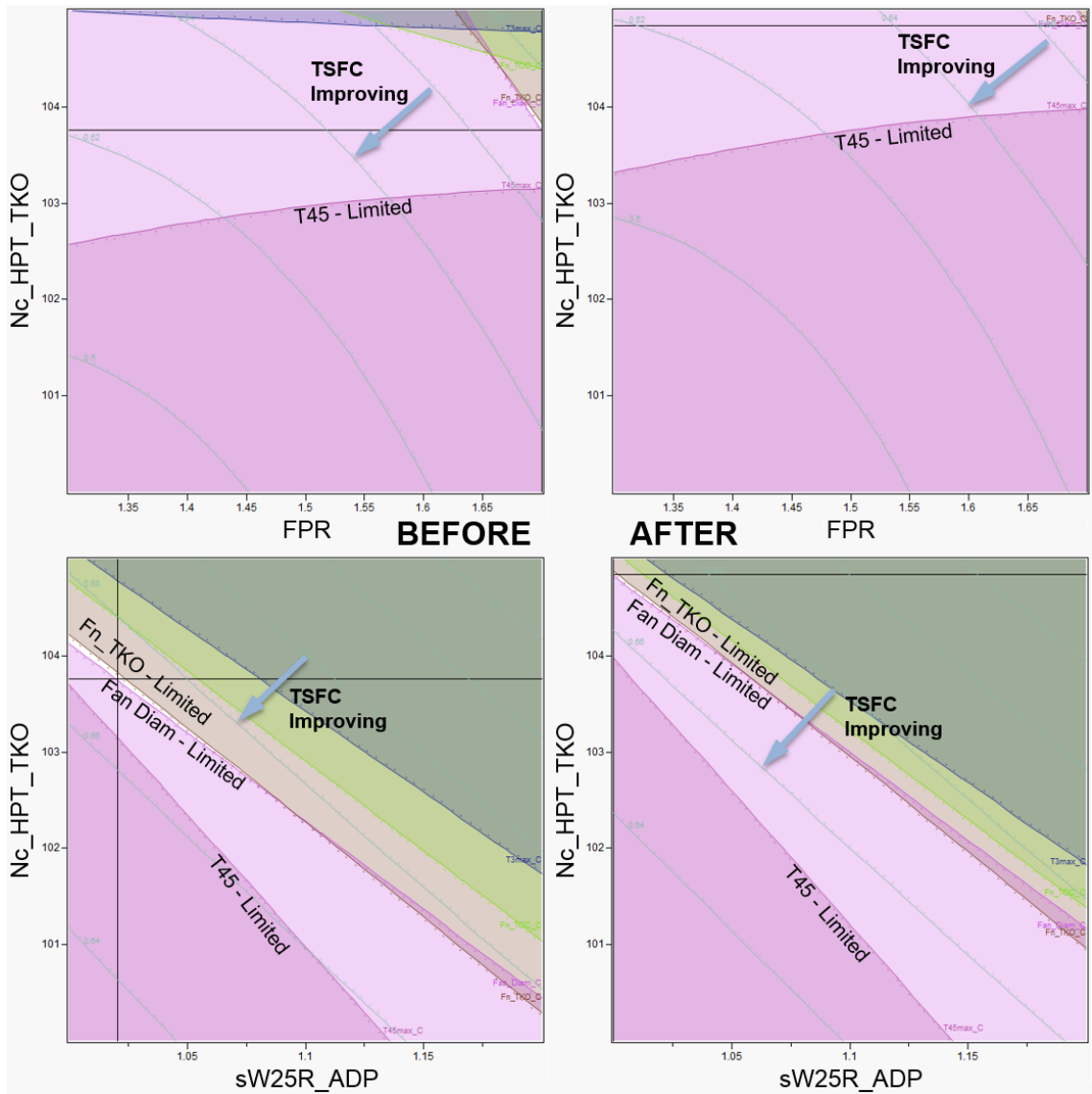


Figure 104: Experiment 4 - Common Core, Variant C Design Exploration: Constraint contours and Variant C design selection settings before and after Core Performance Mitigation.

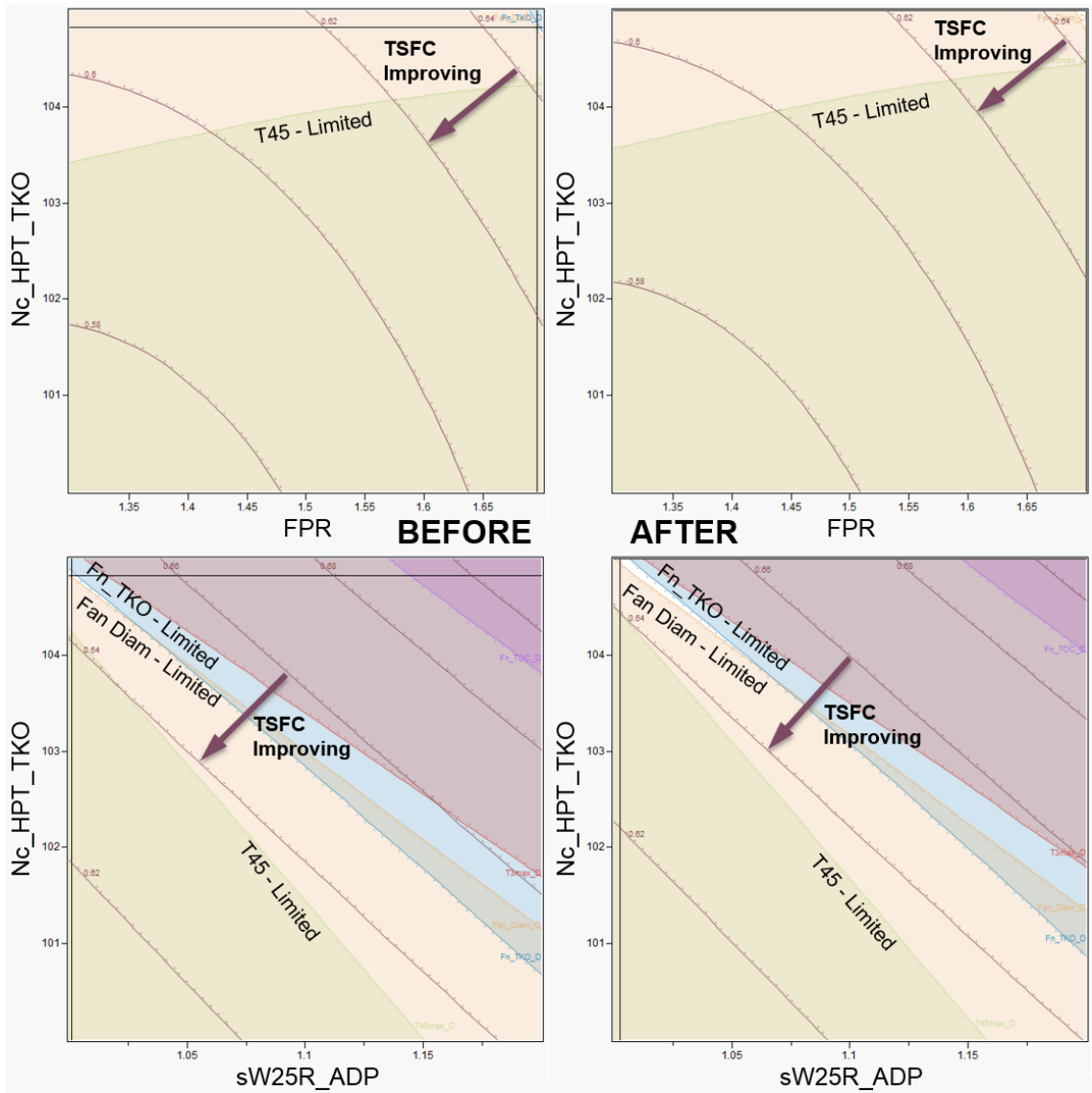


Figure 105: Experiment 4 - Common Core, Variant D Design Exploration: Constraint contours and Variant D design selection settings before and after Core Performance Mitigation.

applications, indicated by the reduced extraction ratio selections. Also, the variant OPR levels are significantly greater than the benchmark designs while no bleed flow is allotted for LPT cooling. The variant designs selections are aiming at producing as much core power as possible with minimum amounts of core flow scaling. Scaling of the core flow would take bypass flow away from the propulsor, in turn reducing the bypass nozzle thrust which is at a premium due to the relatively large takeoff thrust demands under the fan diameter constraints. These common core variants are still able to perform comparably to their benchmark designs, but these mid term, mid technology level common core applications are definitely some of the most limited of the applications in terms of the small regions of the design space that offer highly likely feasible solutions.

Now the high technology, far out engine program application designs will be explored. The remaining three common core applications offer a wide range of thrust capabilities possibly aiming to replace their earlier term engine family members. They also have a wide range of fan diameter constraints imposed. Application E, which has thrust requirements in the middle of the range of the program with a 64 *inch* fan diameter limit, has feasible design regions displayed in Figure 106. The feasible design space resembles those of the mid term Applications C and D, very limited in terms of feasibility and restricted to max FPR designs with substantial over-speeding necessary while having minimal core flow scaling. This application was the one identified for relaxation of the takeoff thrust requirement in order to improve the overall program performance by allowing for a lower power core definition to be used across the applications. These constraint plots show that story, requiring the maximum allowed HPT corrected speed in order to produce the thrust required with the 64 *inch* fan. The constraint diagrams on the right-hand side show how the LPT technology push for Applications A and B and the relaxing of the takeoff thrust requirement of the present application slightly opens up the feasible design region.

Figure 107 contains the constraint diagrams for the common core variant design space of Application F, the engine with lower thrust requirements but with the most restrictive fan diameter constraint of the program. Although the thrust requirements are relatively low, the application accompanies Application G as the engine with the greatest shaft power

Table 67: Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications C and D.

Design Engine Cycle Parameter	Core Defining Cycle Value	
π_{Fan} at ADP	1.70	
π_{HPC} at ADP	13.92	
$\pi_{Overall}$ at ADP	36.24	
T_4 at TKO, $^{\circ}R$	3,342.	
ER at ADP	1.05	
$NChg_{LPT}$, $\%W_{25}$	6.00%	
F_N at TKO, lbf	29,450.	
F_N at TOC, lbf	8,177.	
Application Requirement	App C	App D
Takeoff Thrust, lbf	27,000	30,000
Top of Climb Thrust, lbf	6,400	6,400
Max Fan Diam, in	64	70
Variant Design Cycle Parameter	Variant 4.C	Variant 4.D
π_{Fan} at ADP	1.70	1.70
$\pi_{Overall}$ at ADP	34.83	38.54
T_4 at TKO, $^{\circ}R$	3,350.	3,319.
ER at ADP	0.95	1.02
$NChg_{LPT}$, $\%W_{25}$	0.00%	0.00%
sW_{25R} at ADP	1.021	1.001
$N_{C,HPT,Var}$ at TKO	103.8	104.8
95% CI Metric Value	Variant 4.C	Variant 4.D
F_N at TKO, lbf	26,930	29,957
F_N at TOC, lbf	6,639	7,444
HP Shaft Power at TKO, shp	55,167	58,793
W_{3R} at ADP, lbm/s	7.49	7.49
BPR at ADP	4.58	5.13
Fan Diameter, in	64.0	70.0
$T_{3,max}$, $^{\circ}R$	1,720.	1,742.
$T_{45,max}$, $^{\circ}R$	2,338.	2,339.
TSFC at CRZ, $lbm/lbf/hr$	0.6711	0.6475
Pod Weight, lbm	5,717	6,308

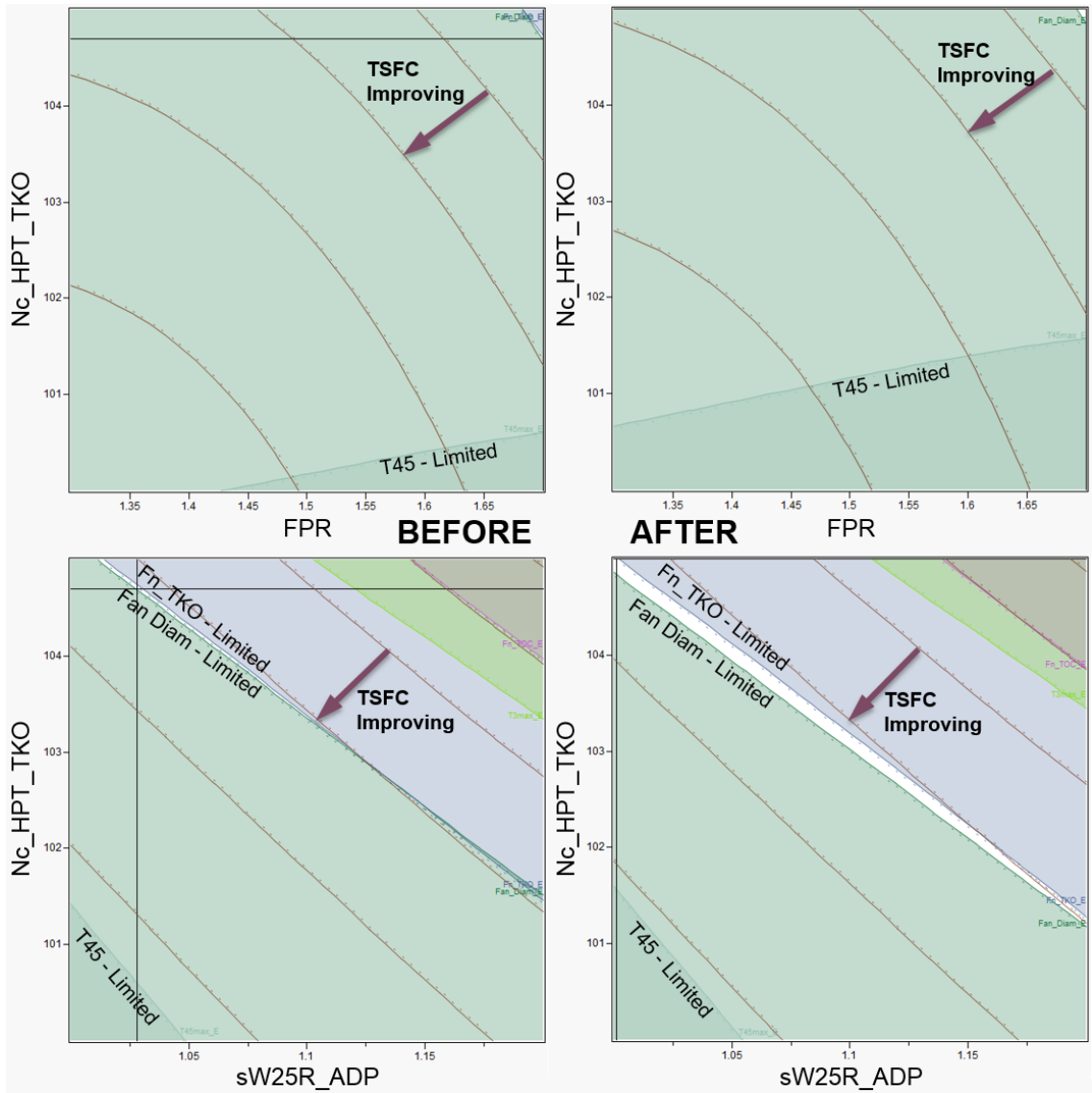


Figure 106: Experiment 4 - Common Core, Variant E Design Exploration: Constraint contours and Variant E design selection settings before and after Core Performance Mitigation.

and bleed extraction requirements of the program. The long term engine projections have greater assumed extraction requirements in order to provide the increases in subsystem power demands that are likely in the more electric aircraft of the future. These increases in installation extractions and the large variability due to uncertainty of the far out projection require significant margins to be present in the variant design selection to maintain the high desired likelihood of achieving requirements and meeting constraints. The constraint diagram expectedly shows that the stringent fan diameter constraint greatly limits the feasible design space, restricting the application to high FPR designs. However, the takeoff thrust requirement is not active for this application. It is the top of climb thrust requirement that further limits the feasible design region, putting a ceiling on the allowed over-speed levels of the HPT at takeoff as well as imposing an upper limit on the amount of core flow scaling. This shows that meeting the required thrust at top of climb while also providing the constant power and bleed extractions is a challenge for this common core application.

The last common core engine variant design considered, Application G whose constraint diagrams can be found in Figure 108, is the high technology variant that requires the maximum levels of takeoff thrust, top of climb thrust, and bleed and power extraction requirements of the entire engine program. It is allowed a larger 70 *inch* fan to provide the thrust levels required. The feasible design region resembles those of Applications C, D, and E, the other relatively high thrust applications that pushed the common core definition to its power limits. Again this is the case, with the engine requiring the maximum FPR along with HPT over-speeding at takeoff near the maximum amount allowed. Before the technology push exercise, the thrust requirement that was most restrictive was the top of climb thrust requirement. However, after the technology push exercise that reduced the common core design power, the takeoff thrust requirement became active along with the fan diameter constraint, and the HPT needed to have an even higher amount of over-speed at takeoff.

The candidate selections of the high technology, longer term common core variant designs can be found in Table 68, along with their 95% likely performance estimates. Comparing the cycle selections to their corresponding benchmark engines in Table 61, the design OPR levels of the variant designs are higher than the benchmark engines, with the most

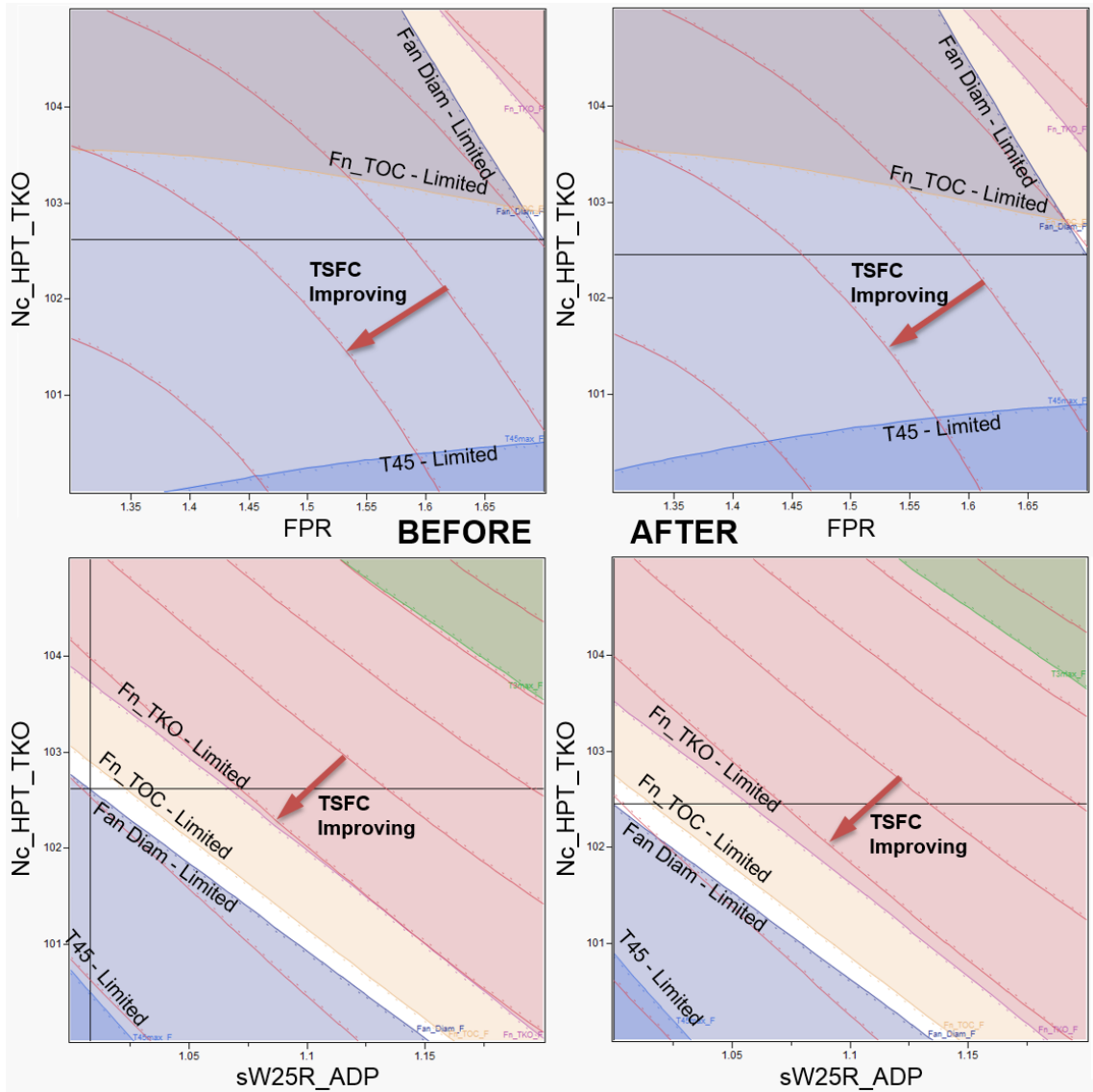


Figure 107: Experiment 4 - Common Core, Variant F Design Exploration: Constraint contours and Variant F design selection settings before and after Core Performance Mitigation.

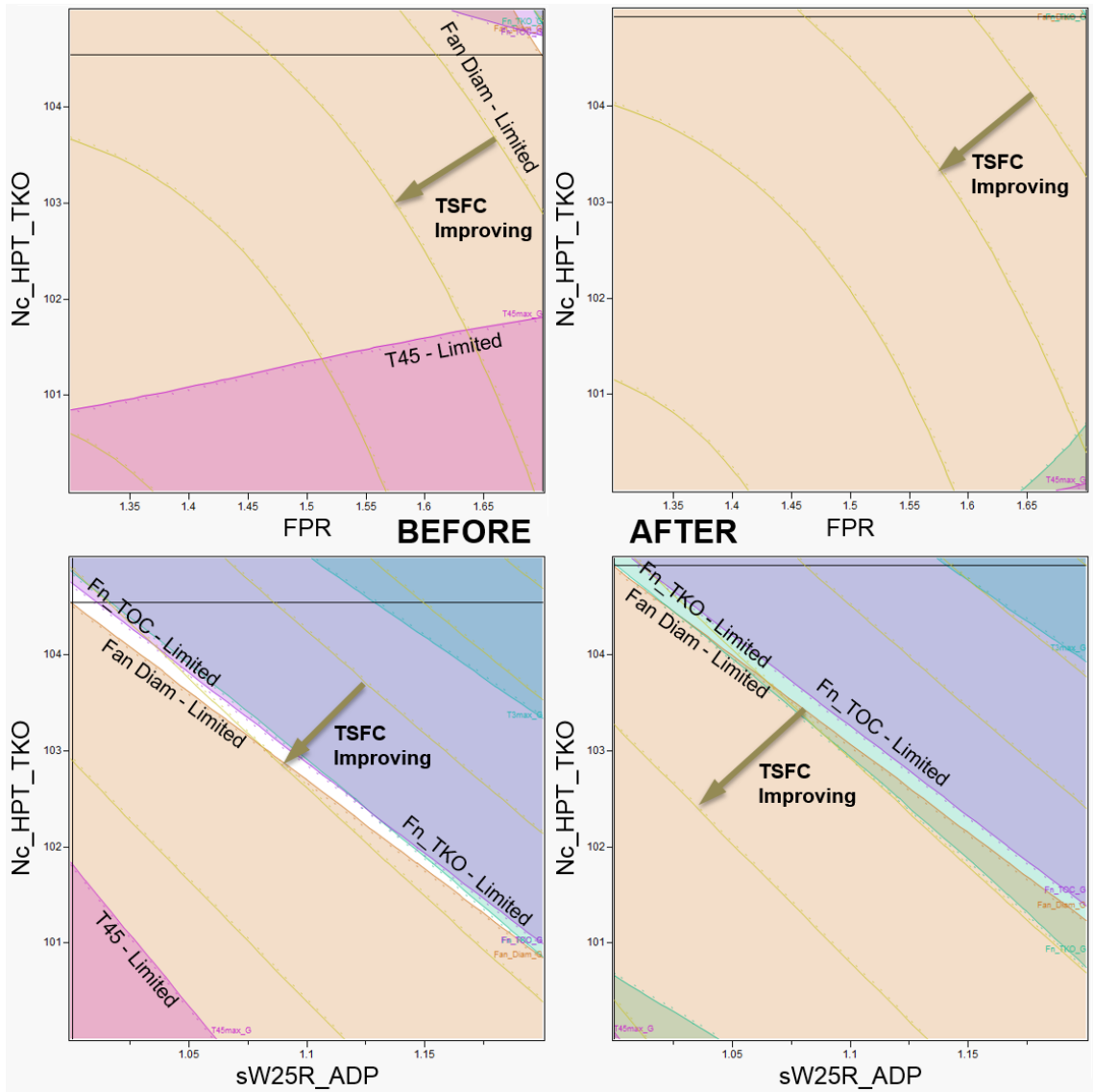


Figure 108: Experiment 4 - Common Core, Variant G Design Exploration: Constraint contours and Variant G design selection settings before and after Core Performance Mitigation.

significant OPR increase present for variant Application G. All three common core variant selections have zero HPC interstage bleed flow for LPT cooling, where the Application E benchmark design had all 6% of the allotted bleed used. This again shows that the power limitation of common core selection requires all the available power to be utilized for thrust production, loading up the core and having significant core nozzle thrust in addition to the bypass nozzle thrust production for the engine with relatively small fan. The most significant differences in the core characteristics can be found in these common core variants when compared to their benchmark new centerline engine designs. The differences in HP shaft power at takeoff as well as the design core corrected flow, comparing the common core engines and their corresponding benchmarks, are very significant. Common core variant Application F had the greatest reduction in core power and corrected flow when compared to its benchmark design, with a core shaft power having 35% less power than the benchmark with 20% less design core corrected flow.

Now that candidate cycles have been explored and selected for all seven common core variant applications, a final comparison is performed. The goal of this examination is to draw observations about how the common core definition selection has impacted the program applications. For each engine application, a table has been constructed that places characteristics of the common core definition, the common core engine variant, and the corresponding benchmark engine cycle next to each other. These charts aim to show that based on the size of the common core definition with respect to the benchmark engine, how the resultant variant design characteristics generally compare to the benchmark designed purely for one particular set of requirements.

Keep in mind that the method of design exploration and feasible design selection employed for this experiment consisted of visualization and manual search throughout the multiple application design space. Therefore, there is not guarantee that global optimal designs were selected for any of the designs considered. This is true for the benchmark engine selections made as well as the common core engine family design selections made. The COMMENCE method enables a wide variety of techniques to be used in the exploration and selection of candidate designs. Since the probabilistic surrogate models are continuous

Table 68: Experiment 4 - Common Core Variant Cycle Selections and corresponding 95% confidence interval performance levels for Applications E, F, and G.

Design Engine Cycle Parameter	Core Defining Cycle Value		
π_{Fan} at ADP	1.70		
π_{HPC} at ADP	13.92		
$\pi_{Overall}$ at ADP	36.24		
T_4 at TKO, $^{\circ}R$	3,342.		
ER at ADP	1.05		
$NChg_{LPT}$, $\%W_{25}$	6.00%		
F_N at TKO, lbf	29,450.		
F_N at TOC, lbf	8,177.		
Application Requirement	App E	App F	App G
Takeoff Thrust, lbf	27,500	24,000	30,000
Top of Climb Thrust, lbf	6,000	6,000	8,000
Max Fan Diam, in	64	61	70
Variant Design Cycle Parameter	Variant 4.E	Variant 4.F	Variant 4.G
π_{Fan} at ADP	1.70	1.70	1.70
$\pi_{Overall}$ at ADP	36.81	30.35	39.72
T_4 at TKO, $^{\circ}R$	3,385.	3,354.	3,330.
ER at ADP	0.93	1.00	0.90
$NChg_{LPT}$, $\%W_{25}$	0.00%	0.00%	0.00%
sW_{25R} at ADP	1.028	1.008	1.001
$N_{C,HPT,Var}$ at TKO	104.7	102.6	104.5
95% CI Metric Value	Variant 4.E	Variant 4.F	Variant 4.G
F_N at TKO, lbf	27,468	25,182	30,298
F_N at TOC, lbf	6,888	5,993	7,993
HP Shaft Power at TKO, shp	58,850	49,383	58,420
W_{3R} at ADP, lbm/s	7.49	7.49	7.49
BPR at ADP	4.35	4.66	4.97
Fan Diameter, in	64.0	61.0	70.0
$T_{3,max}$, $^{\circ}R$	1,754.	1,680.	1,736.
$T_{45,max}$, $^{\circ}R$	2,324.	2,360.	2,353.
TSFC at CRZ, $lbm/lbf/hr$	0.6755	0.6809	0.6559
Pod Weight, lbm	5,543	5,110	6,188

and closed-form, an optimizer could easily be integrated into the design exploration in order to make benchmark and common core design selections. This would likely improve performance levels of both sets of designs. The fact that an optimizer was not used explains some unexpected performance trends where the common core variant design selection is able to perform better in terms of both cruise TSFC and pod weight. This unexpected observation is the result of comparing a non-optimal common core variant design to a non-optimal benchmark design.

Table 69 contains characteristic comparisons of the engine program's initial products, Applications A and B. The common core selection made for the program lies the closest to these two applications' benchmark cycles, in terms of the design HPC exit flow (W_{3R}) and takeoff core power. The common core definition is over-sized for these applications in order to provide the power required of the program's later applications. Both the takeoff core power and the design LPC inlet corrected flow (W_{23R}) of the common core variant cycle is reduced from the core's original design levels in response to the less stringent application requirements than the core is sized for. The over-sized core causes the variant design bypass ratio (the BPR at ADP) to be lower than the benchmark design, causing a compromise in the cruise TSFC achievable.

Table 70 contains characteristics of the mid-term common core program releases, Applications C and D, comparing them to their respective benchmark attributes. For these applications, the design flow and takeoff power of the common core definition is lower than the benchmark cycle values, requiring the over-speeding of the high spool at takeoff in order to produce the thrust required of each application. Having under-sized cores compared to their benchmark engines, the common core variants with HPT vane resets to increase takeoff power have higher corresponding design bypass ratios, improving the TSFC levels at cruise, while having comparable pod weights. These common core variants are able to achieve competitive performance compared to new centerline engines designed particularly for each single set of requirements. However, the core design modifications would require additional development capital along with possible maintenance cost increases due to operating the core at elevated levels during the most demanding conditions experienced by the

Table 69: Experiment 4 - Initial-term common core variant design responses compared to each corresponding benchmark engine cycle selection.

App	Characteristic	Common Core Def	Common Core Var	Benchmark Engine	Var - Ben (% Diff)
4.A	W_{23R} at ADP, lbm/s	113.2	101.0	96.1	+5.10%
	W_{3R} at ADP, lbm/s	7.49	7.49	7.14	+4.91%
	W_{23R} at TKO, lbm/s	105.8	106.5	99.5	+7.04%
	Core Pwr at TKO, shp	48,092	44,620	46,798	-4.65%
	BPR at ADP	N/A	6.01	6.66	-9.66%
	TSFC at CRZ, $lbm/(lbf \cdot hr)$	N/A	0.6367	0.6331	+0.57%
	Pod Weight, lbm	N/A	6,690	7,101	-5.80%
4.B	W_{23R} at ADP, lbm/s	113.2	103.7	96.2	+7.81%
	W_{3R} at ADP, lbm/s	7.49	7.49	7.13	+4.98%
	W_{23R} at TKO, lbm/s	105.8	109.6	99.7	+9.89%
	Core Pwr at TKO, shp	48,092	46,853	46,928	-0.16%
	BPR at ADP	N/A	5.85	6.63	-11.69%
	TSFC at CRZ, $lbm/(lbf \cdot hr)$	N/A	0.6381	0.6332	+0.78%
	Pod Weight, lbm	N/A	6,253	7,086	-11.75%

Table 70: Experiment 4 - Mid-term common core variant design responses compared to each corresponding benchmark engine cycle selection.

App	Characteristic	Common Core Def	Common Core Var	Benchmark Engine	Var - Ben (% Diff)
4.C	W_{23R} at ADP, lbm/s	113.2	104.6	133.7	-21.72%
	W_{3R} at ADP, lbm/s	7.49	7.49	8.63	-13.16%
	W_{23R} at TKO, lbm/s	105.8	110.6	145.3	-23.89%
	Core Pwr at TKO, shp	48,092	55,167	75,993	-27.41%
	BPR at ADP	N/A	4.58	3.97	+15.29%
	TSFC at CRZ, $lbm/(lb_f \cdot hr)$	N/A	0.6711	0.6887	-2.55%
	Pod Weight, lbm	N/A	5,717	6,558	-12.83%
4.D	W_{23R} at ADP, lbm/s	113.2	114.1	118.6	-3.77%
	W_{3R} at ADP, lbm/s	7.49	7.49	8.32	-9.96%
	W_{23R} at TKO, lbm/s	105.8	120.1	130.0	-7.65%
	Core Pwr at TKO, shp	48,092	58,793	69,771	-15.73%
	BPR at ADP	N/A	5.13	5.06	+1.30%
	TSFC at CRZ, $lbm/(lb_f \cdot hr)$	N/A	0.6475	0.6602	-1.93%
	Pod Weight, lbm	N/A	6,308	5,998	+5.17%

engine, at the full power takeoff conditions.

Just as with the initial and mid-term program applications, characteristics of the common core program's later-term engine variants are compared to their corresponding benchmark designs in Table 71. Having a range of performance requirements, there is a spread in the trends found for the three late-term designs. All three applications' benchmark cycles are sized larger than the common core selection. The benchmark cycle for Application G being the closest to the common core definition in terms of max takeoff power, while the benchmark cycle for Application E lies closest to the common core definition in terms of design HPC exit corrected flow. Applications E and F require significant power growth

from the core in order to meet their takeoff thrust requirements with their stringent fan diameter constraints. However, for these constrained designs, again having the smaller core compared to the new centerline designs and allowing it to operate at elevated speeds at takeoff provides an advantage when at cruise, again permitting higher design bypass ratios and correspondingly improve TSFC performance. These two applications were not limited by their top of climb thrust requirements, being able to operate well above the required top of climb thrust levels, and resultantly were able to have minimal design power growth while still meeting all requirements and operating within constraints.

Application G on the other hand, being the engine with the highest thrust and extraction requirement levels but with a less stringent fan diameter constraint needed significant power growth from the common core variant in order to meet its requirements. The absolute takeoff common core power growth was the second greatest of the common core applications, requiring the core power to come closest to the benchmark engine's takeoff core power, coming almost within 10% of its benchmark engine's max power. It also was the only late term common core application to have growth in design core inlet flow (W_{23R} at ADP) in order to produce the highest amount of top of climb thrust required of the common core program. Because of the high top of climb thrust requirement, the under-sized common core application actually has a degraded design bypass ratio compared to its benchmark engine, causing a compromise in cruise TSFC.

4.5.3.6 Common Core Evaluation

Although comparisons have already been done between common core variant designs and their corresponding benchmark engines, the established method for evaluation of the individual designs and overall family needs to be performed. Table 72 contains the evaluation metrics for the resultant common core engine family. Equal preference is given to each application, with cruise TSFC reduction being the primary goal. Table 72 shows that the common core engine family is able to achieve comparable performance levels with respect to the benchmark designs for each application considered. In some cases, better cruise TSFC and engine weights were achieved with the variant designs. This again highlights the fact

Table 71: Experiment 4 - Late-term common core variant design responses compared to each corresponding benchmark engine cycle selection.

App	Characteristic	Common Core Def	Common Core Var	Benchmark Engine	Var - Ben (% Diff)
4.E	W_{23R} at ADP, lbm/s	113.2	109.6	139.9	-21.70%
	W_{3R} at ADP, lbm/s	7.49	7.49	8.18	-8.48%
	W_{23R} at TKO, lbm/s	105.8	115.2	152.5	-24.46%
	Core Pwr at TKO, shp	48,092	58,850	79,968	-26.41%
	BPR at ADP	N/A	4.35	3.69	+17.98%
	TSFC at CRZ, $lbm/(lbf \cdot hr)$	N/A	0.6755	0.6869	-1.65%
	Pod Weight, lbm	N/A	5,543	5,883	-5.78%
4.F	W_{23R} at ADP, lbm/s	113.2	94.5	141.3	-33.07%
	W_{3R} at ADP, lbm/s	7.49	7.49	9.41	-20.36%
	W_{23R} at TKO, lbm/s	105.8	99.8	152.6	-34.60%
	Core Pwr at TKO, shp	48,092	49,383	82,045	-39.81%
	BPR at ADP	N/A	4.66	3.52	+32.28%
	TSFC at CRZ, $lbm/(lbf \cdot hr)$	N/A	0.6809	0.7211	-5.57%
	Pod Weight, lbm	N/A	5,110	5,686	-10.14%
4.G	W_{23R} at ADP, lbm/s	113.2	117.3	125.6	-6.59%
	W_{3R} at ADP, lbm/s	7.49	7.49	9.09	-17.59%
	W_{23R} at TKO, lbm/s	105.8	122.5	138.4	-11.49%
	Core Pwr at TKO, shp	48,092	58,420	65,941	-11.41%
	BPR at ADP	N/A	4.97	5.42	-8.41%
	TSFC at CRZ, $lbm/(lbf \cdot hr)$	N/A	0.6559	0.6687	-1.92%
	Pod Weight, lbm	N/A	6,188	4,052	+52.71%

that non-optimal benchmark designs are being compared to non-optimal common core variant designs. Use of an optimizer in the various design selections would likely improve the benchmark design performance, increasing the amount of performance compromises made by employing common core designs for the applications considered.

This exercise estimates the likely performance levels when employing a common core family across the engine applications considered. It also identifies common core design regions offering feasibility across all applications. Whether considering additional applications or possible modifications of the common core definition to improve the performance of certain program applications, having benchmark designs established offers valuable information to decision makers. Direct quantification of compromises made by using a common core definition can be made. This exercise has shown, based on the assumptions made and the common core variant design rules established in this work, that a common core definition should be sized on the lower end of the power and corrected flow spectrum of applications considered. The compact core design is likely to provide viable variant solutions for low power applications, while having power growth potential through various upgrade options when responding to more demanding engine applications.

4.5.4 Conclusions

This final experiment has demonstrated the full suite of capabilities enabled by the COMMENCE method. A large number of engine applications were considered simultaneously for a common core engine program, each having unique sets of technology and requirements uncertainty distributions. Common core design considerations were made for these applications and simultaneous exploration of the common core design space was performed. A feasible region of the design space was identified, and a core definition was established that offered a high likelihood of achieving feasible variant applications for all sets of requirements considered. It was also found that the likely performance levels achievable by the resultant common core engine applications were relatively close to the levels achieved with new centerline engine designs selected specifically for each set of requirements.

Table 72: Experiment 4 - Common core variant and overall program evaluation based on comparisons to new centerline benchmark cycles independently designed for each application considered.

95% CI Metric Estimate	Bench 3.A	Bench 3.B	-
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6331	0.6332	-
Pod Weight (<i>lbm</i>)	7,101	7,086	-
95% CI Metric Estimate	Variant 3.A	Variant 3.B	-
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6367	0.6381	-
Pod Weight (<i>lbm</i>)	6,690	6,253	-
Preference Levels	App A	App B	-
TSFC Preference: w_{TSFC}	0.9	0.9	-
Weight Preference: w_{Wt}	0.1	0.1	-
App Preference: w_{App}	0.143	0.143	-
Performance Deviation, y_i	-0.07%	-0.48%	-
95% CI Metric Estimate	Bench 3.C	Bench 3.D	-
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6887	0.6602	-
Pod Weight (<i>lbm</i>)	6,558	5,998	-
95% CI Metric Estimate	Variant 3.C	Variant 3.D	-
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6711	0.6475	-
Pod Weight (<i>lbm</i>)	5,717	6,308	-
Preference Levels	App C	App D	-
TSFC Preference: w_{TSFC}	0.9	0.9	-
Weight Preference: w_{Wt}	0.1	0.1	-
App Preference: w_{App}	0.143	0.143	-
Performance Deviation, y_i	-3.58%	-1.21%	-
95% CI Metric Estimate	Bench 3.E	Bench 3.F	Bench 3.G
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6869	0.7211	0.6687
Pod Weight (<i>lbm</i>)	5,883	5,686	4,052
95% CI Metric Estimate	Variant 3.E	Variant 3.F	Variant 3.G
TSFC at CRZ (<i>lbm/lbf/hr</i>)	0.6755	0.6809	0.6559
Pod Weight (<i>lbm</i>)	5,543	5,110	6,188
Preference Levels	App E	App F	App G
TSFC Preference: w_{TSFC}	0.9	0.9	0.9
Weight Preference: w_{Wt}	0.1	0.1	0.1
App Preference: w_{App}	0.143	0.143	0.143
Performance Deviation, y_i	-2.07%	-6.03%	+3.55%
Total Common Core Performance Deviation, Y_{CC}	-1.41%		

Using the ERDS method, surrogate models were generated and employed, offering confidence interval performance estimates for any confidence interval of interest for a design under a wide range of possible uncertainty scenarios. The method allows for the designer to change uncertainty assumptions on the fly during the design space explorations without the need for additional Monte Carlo analyses or repetitions of design simulations. This enables consideration of technology advancements to be made concurrently with design space explorations in order to arrive at the best path to achieving improved overall engine program performance through common core cycle and technology design. Constraints of various common core applications were mapped directly to the core design space, allowing for rapid identification of program limiting factors requiring the common core to have excess size. Mitigation actions were identified and simulated through technology advancement and relaxation requirements. The common core design space was instantaneously updated to reflect the updated assumptions, and the resultant likely program benefits were instantaneously estimated.

Integration of the ERDS process into the COMMENCE method enables common core design studies to account for and simulate any number of possible program applications. Another major development that enabled the efficient use of the COMMENCE method was the establishment of a design-variant multi-engine modeling and simulation environment. Common core variant design rules were implemented to accurately and implicitly enforce geometric commonality for all physics based simulations. The design-variant relationships were captured in the resultant surrogate models, enabling the designer to simulate any number of common core variant applications and having geometric core commonality automatically enforced across all applications. High combinatorial, high dimensional design problems can now be explored with reasonable accuracy, offering high confidence performance estimates for a large common core engine program, all while having a relatively small computational burden. The enabled capabilities of the COMMENCE method allow decision makers to consider many different strategies and scenarios, determine the resultant effects of competing options almost instantaneous, and rapidly draw quantitative conclusions from the valuable knowledge gained from each consideration.

CHAPTER V

CONCLUSIONS

The research objective of the current work is to *develop a gas turbine engine design and decision making process that aims to increase the useful competitive life and overall versatility of a common core engine family. The process should consider current and future competitive engine family performance, utilizing current and eventual technology improvements without the need for a core re-design.* Through model development, testing, and full integration of a series of enabling contributions, the objective was achieved. Multiple application common core engine program design studies can now be performed with minimal computational burden, offering unique observations to be drawn that previously had not been able to be observed.

The present work develops a method to simultaneously consider initial and future common core engine variant applications when exploring a geometrically common core design space and evaluating candidate designs. The Common Engine Core Evaluation (COMMENCE) method serves to allow core designers to consider any number of initial and/or future engine application, simulate a wide variety of unique uncertainty scenarios, and explore the common core design space with the primary goal being to identify a feasible multiple application common core engine family. The resultant design selections increase the likelihood of providing feasible common core engine variant designs for all applications considered while also being able to maintain competitive likely performance levels.

A key enabler in the COMMENCE method is the integration of the Enhanced Robust Design Simulation (ERDS) method, which accounts for and allows for the selection of robust candidate designs that achieve the best performance for any confidence interval performance level desired. The method enables the generation of probabilistic surrogate models that predict confidence interval performance levels of candidate designs over a wide variety of uncertainty scenarios considered. The form of these probabilistic performance estimation

models, when integrated into the COMMENCE method, enables the simultaneous robust design exploration of the common core defining engine design space and n resultant common core variant design applications. Requirements and constraints of particular applications are mapped directly onto the core design space, allowing for rapid identification of feasible core design regions and program limitations.

Another enabler of the common core design space explorations performed in the present work is the developments made in modeling and simulating geometrically common core engine variant designs. Common core variant design rules were established and integrated into a design-variant multiple engine model, enforcing design restrictions to properly simulate a geometrically common core variant design. The relationships established between the core design and corresponding common core variant characteristics allow for commonality to be implicitly enforced across all applications considered during a multiple application design exploration.

The integration of these two major developments into the COMMENCE method results in a highly flexible design environment with a wide range of resultant capabilities. The process requires a small set of initial steps to be taken. The minimal computational burden of these added steps enables robust design explorations of single or multiple common core engine applications to be performed for a wide range of uncertainty scenarios in a highly efficient manner.

5.1 Research Questions and Hypotheses

- ***Primary Research Question:*** *How should core design selections be made for multiple current and future common core applications, ensuring a high likelihood of achieving feasible, competitive common core engine variant designs?*

Primary Hypothesis: *Simultaneous simulation and evaluation of current and future common core applications should be performed when exploring the common core design space in order to quantitatively estimate the feasibility and likely performance levels of program applications due to changes in the common core definition. If this is possible and implemented, the likelihood of achieving feasible, competitive common core variant*

designs will be increased while minimizing the amount of mitigation actions required later in the program.

This hypothesis was confirmed. A seven application common core design space was considered, with various projected technology levels considered. The probability of randomly sampling the seven application common core design space and achieving a highly likely feasible design for all seven applications is almost zero, not even mentioning the chance of landing on a candidate seven application common core program design that performed well throughout the applications considered. Utilizing the COMMENCE method, a feasible seven application common core program was achieved with a high associated level of confidence in meeting all imposed requirements and constraints. The common core program selection also performed well in terms of cruise TSFC and engine weight levels compared to each application's benchmark engine selection. The integration of an optimizer into the process for making design decisions would allow for more optimal designs to be identified both for benchmark designs and for common core engine family designs.

- ***Research Question 1:*** *How should the gas turbine cycle design process be modified to easily evaluate designs under various uncertainty scenarios, in a manner similar to traditional approaches, without the need for added computational burden, repeated simulations, and post-processing of statistical data?*

Hypothesis 1: *Probabilistic performance levels of candidate cycle designs should be estimated with the use of surrogate models that predict likely performance under various inputted uncertainty scenarios for any desired confidence interval.*

Through the development of the ERDS method, this hypothesis was confirmed, offering the designer the ability to perform design space explorations in a similar manner to what cycle designers are familiar with, but with a large amount of added benefits. Candidate design selections were able to be selected that had a high level of confidence in meeting its requirements while satisfying all constraints under the assumed sources of uncertainty present.

The next few research questions were posed directly at the modeling and evaluation of common core engine variant designs.

- **Research Question 2:** *For a given gas turbine engine core, how should a common core engine variant design be simulated? What parameter(s) must be held to consistent values in order to maintain geometric and aerothermodynamic commonality between engine applications?*

Hypothesis 2: *In order to simulate a common core engine variant, design rules must be established and enforced that maintain the design level of corrected flow exiting the high pressure compressor. Maintaining HPC exit corrected flow at design conditions will ensure geometric similarity between common core applications. In order to provide significance to maintaining design HPC exit corrected flow, the design rules must also ensure that the compressor map design operating point is also fixed between common core applications.*

While exploring common core design considerations that have been made in the past during the literature review, it was observed that the HPC design exit corrected flow is the parameter that must remain fixed throughout a common core in order to impose a geometrically fixed common core design across multiple applications. By itself, maintaining the HPC exit corrected flow at an engine's design point does not fully ensure geometric core commonality. The additional rules established in the current work were also necessary to not only simulate a fixed common core, but also allowed for simulation of a geometrically similar common work with the ability to have its core working fluid increased if needed while still maintaining the HPC design exit corrected flow at a specific design map operating point.

- **Research Question 3:** *What design options should be considered for common core engine variant applications in order to distribute development capital across the engine program by taking advantage of commonality, while also offering more design freedom when needed for more demanding applications?*

Hypothesis 3: *In order to provide a wide range of capabilities with common core*

applications, a range of design options should be available with differing levels of upgrade cost and design freedom. Geometrically fixed core and modified common core options should exist to allow for significant core power growth if needed. The common core applications should have design freedom in the LP system in order to be sized for a new set of customer requirements, and technology infusion should be considered for the core and/or the LP engine components in order to provide feasible, competitive common core solutions while placing preference on less expensive upgrade options when at all possible.

This hypothesis was confirmed, allowing the common core engine variants to meet their application requirements with very little change required of the common core definition, while also offering significant core flow scaling with HPT over-speeding for those applications that stood apart from the rest of the program in terms of the core power required to meet their requirements.

- **Research Question 4:** *What range of capabilities can various common core design options achieve without significant compromises made in application performance?*

Hypothesis 4: *A common core variant engine is able to provide a specific range of capabilities while maintaining acceptable performance levels, based on the technology level of the variant design, the core size, the overall engine architecture, and the amount of design freedom permitted for the particular common core application considered.*

Experiment 2 aimed at specifically addressing this research question. This hypothesis was confirmed. As the amount of technology infusion into the common core variant design increases, the growth capabilities also increase. When allowing for core flow scaling, the amount of core power growth is significantly increased, but the resultant thrust growth is dependent upon the remaining limitations of the variant engine. If a fan diameter limit is imposed, increases in core flow reduces the engine bypass ratio, which may reduce the resultant thrust growth achievable.

- **Research Question 5:** *How should a common core engine program consisting of multiple variant design applications be evaluated?*

Hypothesis 5: *A common core variant engine should be evaluated based on the performance deviation from a benchmark, new centerline engine designed specifically for the application it is being designed for. In order to determine the amount of performance compromise made by utilizing a common core application, the benchmark engine should be designed for an identical set of requirements while under the same set of assumptions as were made for the common core variant design. A weighted sum of variant performance levels should be used in the overall evaluation of the engine program.*

Observations from the literature survey from various industries considering product family design and the performance compromises that accompany product lines with increased commonality identified that benchmark designs selected purely for the set of requirements of the single application should be used to quantify the amount of performance compromise made through the utilization of a common core application. The hypothesis confirmed through the literature review.

5.2 Summary of Contributions Made

The principal contributions of this thesis are:

- The development of a method to simultaneously consider current and future common core engine variant applications when exploring the common core design space and evaluating candidate designs. The Common Engine Core Evaluation (COMMENCE) method serves to allow core designers to consider any number of initial and projected engine applications, each having an assigned uncertainty scenario, enabling simultaneous exploration of the multiple application common core design space. The method yields an increased likelihood of the identification of a common core definition that offers feasible variant engines that also achieve competitive performance levels for all applications considered. The method is enabled by the following contributions listed.

- Developments were made making the Robust Design Simulation (RDS) method developed by Mavris et al.[71] much more flexible. The resultant process enables the COMMENCE method and is referred to as the Enhanced Robust Design Simulation (ERDS) method. Three primary enhancements distinguish the ERDS method from the original RDS method. The first is the parameterization of the uncertainty distributions present. Control of the distribution shape parameters is given to the designer, allowing for an infinite number of uncertainty scenarios to be simulated within the ranges of shape parameter settings sampled during the Monte Carlo analysis. The second modification made is in the tracking of probabilistic performance data during Monte Carlo analyses. An entire sweep of confidence interval performance estimates is provided for each DoE case due to the performance variability present due to the uncertainty sources present. The first two modifications allow for the third enhancement to be made, which enables the COMMENCE method to rapidly simulate a common core variant application with a unique scenario of uncertainty distributions. The form of the probabilistic surrogate models was modified, now having the desired confidence interval as well as the uncertainty distribution shape parameter settings as surrogate inputs in addition to the typical design variables. This enables accurate estimates of any confidence interval performance estimate to be provided for a wide variety of uncertainty scenarios.
- The development of a common core modeling and simulation environment that ensures geometric core commonality. A core defining design engine cycle is first simulated, defining the engine core. Then the core characteristics are saved, the core component geometry is fixed, and a geometrically common core variant cycle is then simulated. Variant design rules were established, ensuring commonality between the design and variant cycle simulations while allowing for technology infusion and core upgrades when desired.

5.2.1 Possible Use Cases

The flexibility and wide range of capabilities of the fully integrated COMMENCE method makes it applicable across a wide variety of applications.

- The unique functional form of the probabilistic surrogate models produced by the ERDS method as part of the larger COMMENCE method provides for easy integration of an optimizer into the design method. It was shown in the final experiment that visual design exploration and design selections can be made that achieve feasibility across a large number of engine applications. However, in order to make more optimal robust design selections, the minimal effort required to integrate an optimizer into the COMMENCE method would likely offer improved design selections.
- While optimizers are beneficial when searching for optimal solutions, option selections made through visual exploration offer designers the ability to be in-the-loop when making design considerations and selections. The COMMENCE method has been shown to be able to rapidly simulate various strategic or mitigation options and the design implications are rapidly updated. Rapid visualization of these responses using the COMMENCE method offers decision makers a powerful tool. When collaborating with other decision makers design selection through visualization may be the preferred method of design selection.
- The COMMENCE method aims to reduce the capital risk through the utilization of a common core definition across a wide range of engine applications. Integration of the COMMENCE method into a larger decision making environment for value based design would greatly enhance the environment's capabilities. Considerations made for cost savings can be made and the COMMENCE method would then be able to quantify the likely performance implications across the engine family. Informed with both cost and performance estimates for various options to be considered, the decision maker would be able to make more informed decisions with the added knowledge of likely implications decisions would have in terms of performance and cost.

- The integration of the COMMENCE method into a Bayesian Belief Network based decision making environment may allow for further enhancements of the fully integrated method. Performing common core design space explorations with prior distributions applied to input variable settings may result in the selection of more economically viable options. When carrying out the experiments of this work, no penalty was incurred when scaling the core flow of a variant design. A BBN decision making environment would complement the COMMENCE method by pushing the design selections towards engine families that are more preferable in terms of more economic factors.

5.2.2 Recommendation for Future Method Enhancement

The next logical addition to the COMMENCE method is the integration of an optimizer in the benchmark engine robust design selection, common core design selection, and simultaneous common core variant design selection steps of the COMMENCE method. Although minimal performance improvements are expected for the multiple application design problems addressed in the present work due to the minuscule feasible design regions, significant performance improvements will likely be attainable for the single application benchmark designs against which common core variant design performance levels are evaluated. Also, for common core design problems where either very few variant applications are considered or where there is very little variation in the variant engine capabilities, more significant candidate performance levels may be attainable with the use of an optimizer in the common core program design selections.

5.2.3 Regarding Various Overall Engine Architectures

The present work considers design explorations and selections of cycles corresponding to two shaft, separate flow turbofan engines, with the assumption that common core variant applications are required to be geometrically similar across the family of engines. The COMMENCE method, however, takes a generic form that is independent of the overall engine architecture, the range of capabilities considered, and the design rules imposed. The method can be easily adapted to the common core design exploration of any overall gas

turbine engine architecture with any set of design rules imposed on the designs considered. The results and trends shown in the present work correspond only to the overall engine architecture and single set of design rules imposed on the engine models simulated. Comparison of how the overall versatility of common core engine programs employing various overall engine architectures would allow for an added layer of relationships to be established. This would enable the designer to evaluate candidate engine architectures based on their flexibility and added range of capabilities under common core variant design restrictions.

APPENDIX A

MODELING AND SIMULATION OF A GEOMETRICALLY COMMON CORE VARIANT ENGINE - NPSS CODE

A.1 Overview of Physics-Based Model Used to Simulate a Common Core Engine Variant Design

This appendix contains the NPSS code used to simulate a geometrically common core engine variant design. The goal of the code shown is the enforcement of geometric commonality between a core defining design engine and a variant application of the baseline core definition. The first code displayed contains the setup of the model for the variant design simulation, and the subsequent execution of the common core variant engine simulation. The second code displayed contains the NPSS solver variables established to impose geometric commonality between the core defining engine and the common core variant engine.

NPSS solver variables are used to provide the NPSS solver with goals which the eventual converged solution must achieve. For example: if the designer desires a particular extraction ratio, then the desired value is inputted as a solver dependent variable's right hand side, or *rhs*. The left hand side of the dependent variable, or the *lhs* is the actual extraction ratio of the engine. Convergence occurs when all solver dependent variables' right hand sides and left hand sides contain equivalent values. Solver independent variables are used as knobs for the solver to reach convergence of the dependent variables. For example, in order for the NPSS solver to converge upon a desired extraction ratio, an independent variable is created telling the solver to use the design bypass ratio of the engine as an independent variable which to vary until convergence is achieved on the desired extraction ratio.

The code found in this appendix would be used after converging upon a core defining design engine cycle whose core will be applied to a variant design cycle. The variant design rules established in this work are contained within the code and are labeled with their corresponding variant design rules.

A.2 Procedure for Simulation of Common Core Variant Cycle

The following procedure was used in the simulation of a geometrically common core variant design within the NPSS physics based modeling and simulation environment. The NPSS code that will be shown follows the following steps of the procedure for enforcing commonality and simulating a common core variant engine.

1. Whether manually or using a design space sampling approach, input two sets of input variable values.
 - (a) One set of input variable settings for the core defining design engine
 - (b) One set of input variable settings for the geometrically common core variant engine
2. For the input variable settings that correspond to the candidate core defining design engine, allow the NPSS solver to converge upon a steady state solution and corresponding design definition.
3. Save the complete set of variable values that fully define the core defining design engine.
4. Populate output file with characteristics of the core defining design engine at various operational points of interest.
5. Switch all engine components of the NPSS model assembly to *OFFDESIGN* mode.
6. Switch the components with variant design freedom back to *DESIGN* mode. For the present work, all engine components outside of the core definition were assumed to have design freedom. The core definition was assumed to contain all engine components between the high pressure compressor inlet and the high pressure turbine exit.
7. Add variant engine solver variable pairs to the NPSS solver, including the solver pairs corresponding to variant design rules being enforced.

8. Execute the model, allowing the NPSS solver to converge upon a common core variant engine definition.
9. Populate output file with characteristics of the common core variant engine at various operational points of interest.

The following section contains the NPSS code that carries out the above procedure used in the setup and simulation of a geometrically common core engine variant design.

A.3 NPSS Code for Common Core Variant Design Run

A.3.1 Simulation of Variant Engine - NPSS Code

```
/* *****  
* File Name: NPSS_var设计.run *  
* Description: NPSS engine variant design and sizing *  
***** */  
  
setThermoPackage("GasTbl");  
  
real preFlare_HPC_WcIn;  
real preFlare_HPC_WcOut;  
real preFlare_HPC_Rline;  
real preFlare_HPC_s_WcDes;  
real preFlare_HPC_polyEff;  
real preFlare_HPT_s_WcDes;  
real preFlare_HPC_s_PRdes;  
real preFlare_HPC_s_effDes;  
real s_CmpH_Wc_Flare;  
real base_HPT_s_eff;  
real base_HPT_effDes;  
real base_HPT_parmMap;  
real base_HPT_effMap;  
real HPT_s_eff_Var;  
real preFlare_HPC_NcMap;  
  
//-----  
// User-Defined Elements  
//-----  
#include <print_macros.fnc>  
#include <CompressorMap_mod.int>;  
#include <TurbineNeppMap_mod.int>;  
#include <solver_macrosMAX.fnc>  
#include <solver_macros.fnc>
```

```

#include <cycle.fnc>
#include <Emission.int>
#include <save2.int>

//-----
//                               Model Definition
//-----

cout << "\nMDP Solver Setup - Variant Engine Design\n";

// Save design HPC speed for core defining engine
ADP2.HPC.NcMap.eq-rhs = "preFlare.HPC.NcMap";

// Variant Engine Aerodynamic design point
Element Assembly ADP2 {

    #include <150pax.mdl>
    Brn.Emissions.altName = "";
    // ASSEMBLY DATAVIEWER
    OutFileStream point21Stream { filename = "ADP2.viewOut"; }
    DataView PageViewer point21 {
        #include <npss.point.view>
        outputStreamHandle = "point21Stream";}
    void preexecute() {
        // Load design engine core definition
        #include <save_AtoB.int>
        TrbH.S.map.s_eff = HPT.s_eff.Var;
    }
}

Element Assembly TOC2 {
    #include <150pax.mdl>
    Brn.Emissions.altName = "";
    // ASSEMBLY DATAVIEWER
    OutFileStream point22Stream { filename = "TOC2.viewOut"; }
}

```

```

DataViewer PageViewer point22 {
    #include <npss.point.view>
    outputStreamHandle = "point22Stream";}

// PASS REFERENCE POINT SCALARS TO THIS ASSEMBLY
void preexecute() {
    #include <des.scl2.int>
}
}

Element Assembly TKO2 {
    #include <150pax.mdl>
    Brn.Emissions.altName = "";
    // ASSEMBLY DATAVIEWER
    OutFileStream point23Stream { filename = "TKO2.viewOut"; }
    DataViewer PageViewer point23 {
        #include <npss.point.view>
        outputStreamHandle = "point23Stream";}

// PASS REFERENCE POINT SCALARS TO THIS ASSEMBLY
void preexecute() {
    #include <des.scl2.int>
}
}

//-----
// Changes to Baseline Engine
//-----

// Load core defining engine's relevant
// design variable settings
#include <varsB.list>
#include <independent_initial_AforB.int>

```

```

real PC;

cout << "-----> preVar_HPC_WcIn    = "
      <<preFlare.HPC.WcIn<<endl;
cout << "-----> preVar_HPC_WcOut    = "
      <<preFlare.HPC.WcOut<<endl;
cout << "-----> preVar_HPC_Rline     = "
      <<preFlare.HPC.Rline<<endl;
cout << "-----> preVar_HPC_s_WcDes    = "
      <<preFlare.HPC.s.WcDes<<endl;
cout << "-----> preVar_HPC_s_PRdes    = "
      <<preFlare.HPC.s.PRdes<<endl;
cout << "-----> preVar_HPC_s_effDes    = "
      <<preFlare.HPC.s_effDes<<endl;
cout << "-----> preVar_HPT_s_WcDes    = "
      <<preFlare.HPT.s.WcDes<<endl;
cout << "-----> preVar_HPC_polyEff     = "
      <<preFlare.HPC.polyEff<<endl;
cout << "-----> ADP2.CmpH.delta_eff    = "
      <<ADP2.CmpH.delta_eff<<endl;
cout << "-----> s_CmpH_Wc_Flare        = "
      <<s_CmpH.Wc_Flare<<endl;
cout << "-----> base_HPT_s_eff         = "
      <<base_HPT.s_eff<<endl;
cout << "-----> base_HPT_effDes        = "
      <<base_HPT_effDes<<endl;
cout << "-----> base_HPT_effMap        = "
      <<base_HPT_effMap<<endl;
cout << "-----> ADP2.TrbH.delta_eff    = "
      <<ADP2.TrbH.delta_eff<<endl;

// Adjust variant HPT efficiency based on base core definition
// and technology infusion
HPT_s_eff_Var = (base_HPT_effDes+ADP2.TrbH.delta_eff)
                /base_HPT_effDes*base_HPT_s_eff;

```

```

cout << "-----> HPT_s_eff_Var      = "
      <<HPT_s_eff_Var<<endl;
cout <<" "<< endl;

//-----
//  RUN DESIGN POINT(s)
//-----

// Variant Engine Aerodynamic Design Point
ADP2 {
    // Set variant engine components to
    // DESIGN or OFFDESIGN modes
    setOption( "switchDes", "OFFDESIGN" );
    Amb.setOption( "switchDes", "DESIGN" );
    InEng.setOption( "switchDes", "DESIGN" );
    CmpFan.setOption( "switchDes", "DESIGN" );
    SpltFan.setOption( "switchDes", "DESIGN" );
    D22.setOption( "switchDes", "DESIGN" );
    CmpL.setOption( "switchDes", "DESIGN" );
    B24.setOption( "switchDes", "DESIGN" );
    D241.setOption( "switchDes", "DESIGN" );
    D45.setOption( "switchDes", "DESIGN" );
    B46.setOption( "switchDes", "DESIGN" );
    TrbL.setOption( "switchDes", "DESIGN" );
    D5.setOption( "switchDes", "DESIGN" );
    NozPri.setOption( "switchDes", "DESIGN" );
    B13.setOption( "switchDes", "DESIGN" );
    D15.setOption( "switchDes", "DESIGN" );
    NozSec.setOption( "switchDes", "DESIGN" );
    ShL.setOption( "switchDes", "DESIGN" );
}

// Variant Engine Top of Climb Design Point
TOC2 {
    setOption( "switchDes", "OFFDESIGN" );

```

```

}

// Variant Engine Takeoff Design Point
TKO2 {
    setOption( "switchDes", "OFFDESIGN" );
}

// Input solver limit parameter settings
solver.defaultTolerance = 0.000000000005;
solver.defaultToleranceType = "FRACTIONAL";
solver.maxJacobians = 200;
solver.maxIterations = 5000;
solver.defaultDxLimit = 0.05;

// Prepare NPSS solver for Variant Engine Design run
autoSolverSetup();

solver.addIndependent ( "ADP2_W" );
solver.addDependent ( "TKO2_HPT_NcMap" );

solver.addIndependent ( "TOC2_FAR" );
solver.addDependent ( "ADP2_HPC_NcMap" );

solver.addIndependent ( "ADP2_BPR" );
solver.addDependent ( "ADP2_Ext_Rat" );

solver.addIndependent ( "ADP2_FAR" );
solver.addDependent ( "TOC2_WcRatio" );

solver.addIndependent ( "TKO2_FAR" );
solver.addDependent ( "TKO2_T4" );

solver.addIndependent ( "ADP2_TrBH_Cool1" );
solver.addDependent ( "TKO2_T41" );

```

```
solver.addIndependent ( "ADP2_CmpLPR" );
solver.addDependent ( "ADP2_OPR" );

solver.addIndependent ( "ADP2_CmpLeff" );
solver.addDependent ( "ADP2_CmpLeff_error" );

solver.addIndependent ( "ADP2_ShLNmech" );
solver.addDependent ( "ADP2_ShLNmech_error" );

solver.addIndependent ( "ADP2_SpltFan-O2_MN" );
solver.addDependent ( "ADP2_SpltFan-O2Area" );

solver.addIndependent ( "ADP2_CmpFan-Inlet_MN" );
solver.addDependent ( "TOC2_CmpFan-Inlet_MN" );

solver.addIndependent ( "ADP2_CmpFan-Outlet_MN" );
solver.addDependent ( "TOC2_CmpFan-Outlet_MN" );

solver.addIndependent ( "ADP2_SpltFan-O1_MN" );
solver.addDependent ( "TOC2_SpltFan-O1_MN" );

solver.addIndependent ( "ADP2_CmpL-Outlet_MN" );
solver.addDependent ( "TOC2_CmpL-Outlet_MN" );

solver.addIndependent ( "ADP2_CmpH-Inlet_MN" );
solver.addDependent ( "TOC2_CmpH-Inlet_MN" );

solver.addIndependent ( "ADP2_TrbL-Inlet_MN" );
solver.addDependent ( "TOC2_TrbL-Inlet_MN" );

solver.addIndependent ( "ADP2_TrbL-Outlet_MN" );
solver.addDependent ( "TOC2_TrbL-Outlet_MN" );

solver.addIndependent ( "ADP2_NozPri-Inlet_MN" );
solver.addDependent ( "TOC2_NozPri-Inlet_MN" );
```



```

solver.addIndependent ( "ADP2_D22_Outlet_MN" );
solver.addDependent ( "ADP2_D22_Area_error" );

solver.addIndependent ( "ADP2_B24_Outlet_MN" );
solver.addDependent ( "ADP2_B24_Area_error" );

solver.addIndependent ( "ADP2_B13_Outlet_MN" );
solver.addDependent ( "ADP2_B13_Area_error" );

solver.addIndependent ( "ADP2_D15_Outlet_MN" );
solver.addDependent ( "ADP2_D15_Area_error" );

// Add Ambient Conditions To Solver
solver.addIndependent ( "ADP2_Amb_MN" );
solver.addDependent ( "ADP2_AmbCond_MN" );

solver.addIndependent ( "ADP2_Amb_alt" );
solver.addDependent ( "ADP2_AmbCond_alt" );

solver.addIndependent ( "ADP2_Amb_dTs" );
solver.addDependent ( "ADP2_AmbCond_dTs" );

solver.addIndependent ( "TKO2_Amb_MN" );
solver.addDependent ( "TKO2_AmbCond_MN" );

solver.addIndependent ( "TKO2_Amb_alt" );
solver.addDependent ( "TKO2_AmbCond_alt" );

solver.addIndependent ( "TKO2_Amb_dTs" );
solver.addDependent ( "TKO2_AmbCond_dTs" );

solver.addIndependent ( "TOC2_Amb_MN" );
solver.addDependent ( "TOC2_AmbCond_MN" );

```

```

solver.addIndependent ( "TOC2_Amb.alt" );
solver.addDependent ( "TOC2_AmbCond.alt" );

solver.addIndependent ( "TOC2_Amb.dTs" );
solver.addDependent ( "TOC2_AmbCond.dTs" );

//-----
//  FLARED COMPRESSOR LOGIC
//  Equivalent to the Variant Design Rules
//  Established within the body of this work
//-----

cout << "<<<<<<< FLARED HPC OFF-DESIGN RUN >>>>>>>"<<endl;

// Enforces Variant Design Rule #1
solver.addIndependent ( "Ind_s_CmpH_WcIn.Flared" );
solver.addDependent ( "Dep_s_CmpH_WcIn.Flared" );

// Enforces Variant Design Rule #2
solver.addIndependent ( "Ind_s_CmpH_WcOut.Flared" );
solver.addDependent ( "Dep_s_CmpH_WcOut.Flared" );

// Enforces Variant Design Rule #3
solver.addIndependent ( "Ind_CmpH_RlineMap.Flared" );
solver.addDependent ( "Dep_CmpH_RlineMap.Flared" );

// Enforces Variant Design Rule #4
solver.addIndependent ( "Ind_CmpH_polyEff.Flared" );
solver.addDependent ( "Dep_CmpH_polyEff.Flared" );

verify();
cout << "\nRunning MDP Solver\n\n";
run();

```

```

cout<<"ADP FLARED Off-Design Solver Converged: "
    <<solver.converged<<endl;
cout << " " << endl;

// Ensure core characteristics are as expected
// during model development

cout << "-----> postFlare_HPC_WcIn    = "
    <<ADP2.CmpH.Fl-I.Wc<<endl;
cout << "-----> postFlare_HPC_WcOut    = "
    <<ADP2.CmpH.Fl-O.Wc<<endl;
cout << "-----> postFlare_HPC_Rline    = "
    <<ADP2.CmpH.S_map.RlineMap<<endl;
cout << "-----> postFlare_HPC_s_WcDes = "
    <<ADP2.CmpH.S_map.s_WcDes<<endl;
cout << "-----> postFlare_HPC_s_PRdes = "
    <<ADP2.CmpH.S_map.s_PRdes<<endl;
cout << "-----> postFlare_HPC_s_effDes = "
    <<ADP2.CmpH.S_map.s_effDes<<endl;
cout << "-----> postFlare_HPT_s_WcDes = "
    <<ADP2.TrbH.S_map.s_Wp<<endl;
cout << "-----> postFlare_HPC_polyEff = "
    <<ADP2.CmpH.effPoly<<endl;
cout << "-----> postFlare_HPC_Wc_Scale = "
    <<ADP2.CmpH.Fl-I.Wc/preFlare_HPC_WcIn<<endl;
cout <<" " << endl;

//-----
// Ensure core characteristics are as expected
// during model development
//-----
cout << "-----" <<endl;
cout << "| " <<endl;
cout << "|          Post/Pre Flare Compare" <<endl;

```

```

cout << "| " << endl;
cout << "|>>> FLARE FACTOR: " << s_CmpH.Wc.Flare << endl;
cout << "| " << endl;
cout << "|>>> WcIn Post/Pre          = " << ADP2.CmpH.Fl_I.Wc
    /preFlare_HPC_WcIn << endl;
cout << "|>>> WcOut Post/Pre         = " << ADP2.CmpH.Fl_O.Wc
    /preFlare_HPC_WcOut << endl;
cout << "|>>> Rline Post/Pre         = " << ADP2.CmpH.S_map.RlineMap
    /preFlare_HPC_Rline << endl;
cout << "|>>> polyEff Post/Pre       = " << ADP2.CmpH.effPoly
    /preFlare_HPC_polyEff << endl;
cout << "| " << endl;
cout << "|>>> s_WcDes Post/Pre        = " << ADP2.CmpH.S_map.s_WcDes
    /preFlare_HPC_s_WcDes << endl;
cout << "|>>> s_PRdes Post/Pre       = " << ADP2.CmpH.S_map.s_PRdes
    /preFlare_HPC_s_PRdes << endl;
cout << "|>>> s_effDes Post/Pre      = " << ADP2.CmpH.S_map.s_effDes
    /preFlare_HPC_s_effDes << endl;
cout << "|>>> HPT_s_Wp Post/Pre      = " << ADP2.TrbH.S_map.s_Wp
    /preFlare_HPT_s_WcDes << endl;
cout << "|>>> HPC_NcMap Post/Pre     = " << ADP2.CmpH.S_map.NcMap
    /preFlare_HPC_NcMap << endl;
cout << "-----" << endl;
cout << " " << endl;

cout << "ADP2.CmpH.a_effDes_size = " << ADP2.CmpH.a_effDes_size << endl;
cout << "ADP2.CmpH.S_map.effDes  = " << ADP2.CmpH.S_map.effDes << endl;
cout << "TOC2.CmpH.S_map.effDes  = " << TOC2.CmpH.S_map.effDes << endl;
cout << "TKO2.CmpH.S_map.effDes  = " << TKO2.CmpH.S_map.effDes << endl;
cout << "ADP2.CmpH.Cool1.fracBldW = " << ADP2.CmpH.Cool1.fracBldW << endl;
cout << "TOC2.CmpH.Cool1.fracBldW = " << TOC2.CmpH.Cool1.fracBldW << endl;
cout << "TKO2.CmpH.Cool1.fracBldW = " << TKO2.CmpH.Cool1.fracBldW << endl;

cout << "ADP2.CmpH.ADP_CmpH_effDes = " <<

```

```

ADP2.CmpH.ADP_CmpH_effDes<<endl;
cout << "ADP2.CmpH.ADP_CmpH_effDesBase = "<<
ADP2.CmpH.ADP_CmpH_effDesBase<<endl;
cout << "ADP2.CmpH.ADP_CmpH_a_effDes_size = "<<
ADP2.CmpH.ADP_CmpH_a_effDes_size<<endl;

cout << "ADP2.CmpH.delta_eff = "<<ADP2.CmpH.delta_eff<<endl;

cout << "ADP2.TrbH.eff = "<< ADP2.TrbH.eff << endl;
cout << "ADP2.TrbH.S_map.s_eff = "<< ADP2.TrbH.S_map.s_eff << endl;
cout << "TOC2.TrbH.S_map.s_eff = "<< ADP2.TrbH.S_map.s_eff << endl;
cout << "TKO2.TrbH.S_map.s_eff = "<< ADP2.TrbH.S_map.s_eff << endl;

printPride();

cout << "Print to Point Viewer\n\n";
ADP2.point21.display();
TOC2.point22.display();
TKO2.point23.display();
cout << "NcPct = "<<ADP2.CmpH.NcPct<<endl;
cout << "ADP2.CmpH.CmpHBld.Wbld = "<<ADP2.CmpH.CmpHBld.Wbld<<endl;
cout << "TOC2.CmpH.CmpHBld.Wbld = "<<TOC2.CmpH.CmpHBld.Wbld<<endl;
cout << "TKO2.CmpH.CmpHBld.Wbld = "<<TKO2.CmpH.CmpHBld.Wbld<<endl;

cout << "Saving Cycle and Final Independents\n\n";
//Saves engine data for off-design simulations
save.engine_cycleB();

// Saves final values of solver independent variables
save.final_independentsB();

```

A.3.2 Common Core Variant Engine - NPSS Solver Variables

```
/* *****  
* File Name: solver_vars.fnc *  
* Description: NPSS and WATE++ solver variable *  
*           declaration for Variant Engine Design *  
***** */  
  
//=====  
// Flared HPC NPSS Solver Variables  
// Enforces the Variant Design Rules  
// Established within the body of this work  
//=====  
Independent Ind_s_CmpH_WcIn_Flared {  
    varName = "ADP2.CmpH.S_map.s.WcDes";  
    xRef = 1.0160;  
}  
  
Dependent Dep_s_CmpH_WcIn_Flared {  
    eq_lhs = "ADP2.CmpH.Fl_I.Wc";  
    eq_rhs = "s_CmpH.Wc_Flare*preFlare_HPC.WcIn";  
}  
  
Independent Ind_s_CmpH_WcOut_Flared {  
    varName = "ADP2.CmpH.S_map.s.PRdes";  
    xRef = 0.9994;  
}  
  
Dependent Dep_s_CmpH_WcOut_Flared {  
    eq_lhs = "ADP2.CmpH.Fl_O.Wc";  
    eq_rhs = "preFlare_HPC.WcOut";  
}  
  
Independent Ind_CmpH_RlineMap_Flared {
```

```

    varName = "ADP2.TrbH.S_map.s.Wp";
    xRef = 0.9270;
}

Dependent Dep_CmpH_RlineMap_Flared {
    eq_lhs = "ADP2.CmpH.S_map.RlineMap";
    eq_rhs = "preFlare_HPC_Rline";
}

Independent Ind_CmpH_polyEff_Flared {
    varName = "ADP2.CmpH.S_map.s_effDes";
    xRef = 1.00;
}

Dependent Dep_CmpH_polyEff_Flared {
    eq_lhs = "ADP2.CmpH.effPoly";
    eq_rhs = "preFlare_HPC_polyEff+ADP2.CmpH.delta_eff";
}

//=====
// B_NPSS - Remainder of Variant Engine Independents
//=====

Independent ADP2_W {
    varName = "ADP2.Amb.W";}

Independent ADP2_FAR {
    varName = "ADP2.Brn.FAR"; }

Independent TOC2_FAR {
    varName = "TOC2.Brn.FAR"; }

Independent TKO2_FAR {
    varName = "TKO2.Brn.FAR"; }

```

```

Independent ADP2_BPR {
    varName = "ADP2.SpltFan.BPR"; }

Independent ADP2_CmpLeff {
    varName = "ADP2.CmpL.S_map.effDes"; }

Independent ADP2_CmpLPR {
    varName = "ADP2.CmpL.S_map.PRdes";
    dxLimit = 0.05;
    dxLimitType = "ABSOLUTE";}

Independent ADP2_TrbH_Cool1 {
    varName = "ADP2.B3.Cool1.fracW";
    dxLimit = 0.01;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_TrbL_Cool1 {
    varName = "ADP2.CmpH.Cool1.fracBldW";
    dxLimit = 0.01;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_ShLNmech {
    varName = "ADP2.ShL.Nmech";}

Independent ADP2_ShHNmech {
    varName = "ADP2.ShH.Nmech";}

Independent ADP2_CmpFanNpctDes {
    varName = "ADP2.CmpFan.NpctDes"; }

Independent ADP2_SpltFan_O2_MN {
    varName = "ADP2.SpltFan.Fl_O2.MN"; }

Independent ADP2_CmpFan_Inlet_MN {
    varName = "ADP2.InEng.Fl_O.MN";

```



```

dxLimit = 0.02;
dxLimitType = "ABSOLUTE"; }

Independent ADP2_CmpFan_Outlet_MN {
    varName = "ADP2.CmpFan.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_SpltFan_O1_MN {
    varName = "ADP2.SpltFan.Fl_O1.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_CmpL_Outlet_MN {
    varName = "ADP2.CmpL.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_CmpH_Inlet_MN {
    varName = "ADP2.D241.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_CmpH_Outlet_MN {
    varName = "ADP2.CmpH.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_Brn_Inlet_MN {
    varName = "ADP2.D35.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_TrbH_Inlet_MN {
    varName = "ADP2.B41.Fl_O.MN";

```

```

dxLimit = 0.02;
dxLimitType = "ABSOLUTE"; }

Independent ADP2.TrbH.Outlet_MN {
    varName = "ADP2.TrbH.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.TrbL.Inlet_MN {
    varName = "ADP2.D45.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.TrbL.Outlet_MN {
    varName = "ADP2.TrbL.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.NozPri.Inlet_MN {
    varName = "ADP2.D5.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.D22.Outlet_MN {
    varName = "ADP2.D22.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.B24.Outlet_MN {
    varName = "ADP2.B24.Fl_O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2.B3.Outlet_MN {
    varName = "ADP2.B3.Fl_O.MN";

```

```

dxLimit = 0.02;
dxLimitType = "ABSOLUTE"; }

Independent ADP2_Brn_Outlet_MN {
    varName = "ADP2.Brn.Fl.O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_B13_Outlet_MN {
    varName = "ADP2.B13.Fl.O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

Independent ADP2_D15_Outlet_MN {
    varName = "ADP2.D15.Fl.O.MN";
    dxLimit = 0.02;
    dxLimitType = "ABSOLUTE"; }

//=====
// B_NPSS - Remainder of Variant Engine NPSS Dependents
//=====

Dependent TKO2_Thrust {
    eq_lhs = "TKO2.PERF.myFn";
    eq_rhs = "20000.0";    }

Dependent TOC2_Thrust {
    eq_lhs = "TOC2.PERF.myFn";
    eq_rhs = "5000.0";    }

Dependent TKO2_HPC_NcMap {
    eq_lhs = "TKO2.CmpH.S_map.NcMap";
    eq_rhs = "1.0";    }

Dependent TOC2_HPC_NcMap {
    eq_lhs = "TOC2.CmpH.S_map.NcMap";

```

```

eq.rhs = "1.05";    }

Dependent ADP2_Ext_Rat {
  eq.lhs = "ADP2.NozSec.Fl_O.Pt/ADP2.NozPri.Fl_O.Pt";
  eq.rhs = "1.0184"; }

Dependent TKO2_HPT_NcMap {
  eq.lhs = "TKO2.TrbH.S_map.NpMap";
  eq.rhs = "100.";  }

Dependent TOC2_HPT_NcMap {
  eq.lhs = "TOC2.TrbH.S_map.NpMap";
  eq.rhs = "105.";  }

Dependent ADP2_HPT_NcMap {
  eq.lhs = "ADP2.TrbH.S_map.NpMap";
  eq.rhs = "100.";  }

Dependent ADP2_HPC_NcMap {
  eq.lhs = "ADP2.CmpH.S_map.NcMap";
  eq.rhs = "0.976"; }

Dependent TOC2_T41 {
  eq.lhs = "TOC2.B41.Fl_O.Tt";
  eq.rhs = "2800.0"; }

Dependent TKO2_T41 {
  eq.lhs = "TKO2.B41.Fl_O.Tt";
  eq.rhs = "2800.0"; }

Dependent TKO2_T45 {
  eq.lhs = "TKO2.B46.Fl_O.Tt";
  eq.rhs = "2307.0"; }

Dependent TKO2_T4 {

```

```

eq_lhs = "TKO2.Brn.TtCombOut";
eq_rhs = "3200.0";      }

Dependent TOC2.WcRatio {
  eq_lhs = "TOC2.CmpFan.Fl_I.Wc/ADP2.CmpFan.Fl_I.Wc";
  eq_rhs = "1.03";      }

Dependent ADP2.OPR      {
  eq_lhs = "ADP2.CmpH.Fl_O.Pt/ADP2.CmpFan.Fl_I.Pt";
  eq_rhs = "40";        }

Dependent ADP2.CmpLeff_error {
  eq_lhs = "ADP2.CmpL.S_map. effDes";
  eq_rhs = "0.89";      }

Dependent ADP2.ShLNmech_error {
  eq_lhs = "ADP2.ShL.Nmech";
  eq_rhs = "4800";      }

Dependent ADP2.ShHNmech_error {
  eq_lhs = "ADP2.ShH.Nmech";
  eq_rhs = "10000";     }

Dependent ADP2.SpltFan.O2Area {
  eq_lhs = "ADP2.SpltFan.Fl_02.Aphy";
  eq_rhs = "ADP2.CmpFan.Fl_O.Aphy-ADP2.SpltFan.Fl_01.Aphy"; }

Dependent TOC2.CmpFan_Inlet_MN {
  eq_lhs = "TOC2.InEng.Fl_O.MN";
  eq_rhs = "0.5";      }

Dependent TOC2.CmpFan_Outlet_MN {
  eq_lhs = "TOC2.CmpFan.Fl_O.MN";
  eq_rhs = "0.5";      }

```

```

Dependent TOC2_SpltFan_O1_MN {
    eq_lhs = "TOC2.SpltFan.Fl_01.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_CmpL_Outlet_MN {
    eq_lhs = "TOC2.CmpL.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_CmpH_Inlet_MN {
    eq_lhs = "TOC2.D241.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_CmpH_Outlet_MN {
    eq_lhs = "TOC2.CmpH.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_Brn_Inlet_MN {
    eq_lhs = "TOC2.D35.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_TrbH_Inlet_MN {
    eq_lhs = "TOC2.B41.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_TrbH_Outlet_MN {
    eq_lhs = "TOC2.TrbH.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_TrbL_Inlet_MN {
    eq_lhs = "TOC2.D45.Fl_0.MN";
    eq_rhs = "0.5"; }

Dependent TOC2_TrbL_Outlet_MN {
    eq_lhs = "TOC2.TrbL.Fl_0.MN";
    eq_rhs = "0.5"; }

```

```

Dependent TOC2_NozPri_Inlet_MN {
    eq_lhs = "TOC2.D5.Fl.O.MN";
    eq_rhs = "0.5"; }

Dependent ADP2_D22_Area_error {
    eq_lhs = "ADP2.D22.Fl.O.Aphy/ADP2.SpltFan.Fl.01.Aphy";
    eq_rhs = "1"; }

Dependent ADP2_B24_Area_error {
    eq_lhs = "ADP2.B24.Fl.O.Aphy/ADP2.CmpL.Fl.O.Aphy";
    eq_rhs = "1"; }

Dependent ADP2_B3_Area_error {
    eq_lhs = "ADP2.B3.Fl.O.Aphy/ADP2.D35.Fl.O.Aphy";
    eq_rhs = "1"; }

Dependent ADP2_Brn_Area_error {
    eq_lhs = "ADP2.Brn.Fl.O.Aphy/ADP2.B41.Fl.O.Aphy";
    eq_rhs = "1"; }

Dependent ADP2_B13_Area_error {
    eq_lhs = "ADP2.B13.Fl.O.Aphy/ADP2.SpltFan.Fl.02.Aphy";
    eq_rhs = "1"; }

Dependent ADP2_D15_Area_error {
    eq_lhs = "ADP2.D15.Fl.O.Aphy/ADP2.SpltFan.Fl.02.Aphy";
    eq_rhs = "1"; }

```

APPENDIX B

MONTE CARLO ANALYSIS FOR PROBABILISTIC SURROGATE MODEL TRAINING DATA GENERATION

B.1 Overview of the Probabilistic Sampling and Simulation Strategy

This appendix contains the Matlab script used to perform Monte Carlo probabilistic analyses performed to generate the data necessary to train probabilistic surrogate models. The goal of the script is to provide input/output data that sufficiently samples the probabilistic, common core design space in order to train the probabilistic surrogate models used in all the single and multiple application design problems posed in the present work.

A latin hypercube DoE is first constructed for the control variables (CVs) considered, whose set is made up of design variables and uncertainty distribution shape parameters for both the core defining design engine and the corresponding common core variant engine design. For each unique set of control variable settings, referred to as a DoE case, tens of thousands of design simulations are repeated in order to account for the uncertainty assumed present in the design problem. Noise variable (NVs) distributions are constructed for each DoE case based on the case's unique set of distribution shape parameter settings.

For each repetition of each DoE case, making up a unique input vector of variable settings that would result in a unique set of engine model performance levels, the previously trained deterministic surrogate models are evaluated. The variability in the DoE case's estimated performance due to the inputted uncertainty distributions is then compiled. Quantile performance estimates are determined for each engine performance metric of interest. The quantile performance estimates are then compiled for each DoE case of control variable settings. This resultant data table of DoE input values and corresponding quantile performance estimates are then used to construct the probabilistic surrogate models used in the probabilistic design problems considered in the experiments of the present work.

B.2 Procedure for Robust Design Simulation of an Engine Design Under Consideration

The code that is contained within the following section follows this step by step procedure aiming to assess candidate designs under a wide variety of uncertainty scenarios.

1. Establish the set of control variables whose values will be explored. This set of variables should contain:
 - (a) Design variables for which value selections will be made after exploration of the design space
 - (b) Uncertainty distribution shape parameters whose values will be explored allowing for various uncertainty scenarios to be simulated within the range of values considered
2. Establish ranges for all control variables, defining the boundaries of the design and uncertainty space being explored during the probabilistic assessment.
3. Establish the set of noise variables to which uncertainty distributions will be applied
4. Establish ranges of noise variable values to be considered. Ensure that all uncertainty distributions considered will be contained within the established noise variable ranges.
5. Generate a Design of Experiments (DoE), whose number of dimensions is determined based on the number of control variables considered in the assessment to be performed. A normalized DoE can first be generated, then later resized to explore the entire design and uncertainty space considered.
6. For each DoE case, which is made up of a unique set of control variable settings, generate uncertainty distributions whose shapes are determined by the shape parameter settings for the current DoE case.
7. For each DoE case, perform a Monte-Carlo probabilistic assessment for the design under consideration and the corresponding uncertainty distributions considered in the current DoE case. The number of design repetitions to perform depends on the

number of uncertainty distributions considered and desired resolution of the confidence interval response data to be generated for each DoE case.

8. Using the deterministic surrogate models previously trained that estimate engine performance characteristics of the design and technology space presently considered, evaluate each Monte-Carlo repetition of the current DoE case's design variable settings.
9. For each performance estimate of interest, order the response data from the Monte-Carlo analysis from least to greatest.
10. Determine the entire range of confidence interval performance levels for the current design and uncertainty distributions under consideration.

After carrying out the above procedure, probabilistic surrogate models of performance metrics of interest can be trained based only on the set of control variable settings considered. This allows the designer to instantaneously predict the desired confidence interval performance for a particular design and corresponding uncertainty scenario considered. Effects of the uncertainty distributions of noise variable settings are implicitly accounted for in the set of probabilistic surrogate models, eliminating the need for any further design repetitions in the estimation of the variation in design performance under uncertainty.

The following section contains the actual Matlab code used to generate confidence interval data for probabilistic surrogate model training. For the present work, 3,000 unique DoE designs were evaluated using the previously trained deterministic surrogate models. 50,000 design repetitions were evaluated under the unique uncertainty scenario considered for each DoE case. For the present work, the probabilistic assessment requiring 150 million design evaluations was carried out on a single quad-core machine. The entire assessment required less than two hours to carry out. The resultant set of probabilistic surrogate models are able to instantaneously predict confidence interval performance with a single function call for each performance metric of interest, greatly reducing the amount of time required for such a design evaluation.

B.3 Matlab Code

```
% Preload the kernel based support vector machine that classifies a set of
% input variable settings as expected-feasible or expected-infeasible
load SVMSTRUCT
```

```
%% DECLARE VARIABLES AND CORRESPONDING MIN AND MAX VALUES TO EXPLORE
```

```
% Remaining NPSS Inputs - Not Used when evaluating surrogates, just here
% in order to include entire NPSS model input vector components
```

```
CmpFan.HtoT      = 0.325 ;
CmpH.HtoT        = 0.7 ;
CmpFan.dUtip     = 99.1978 ;
CmpH.dUtip       = -13.2367 ;
a_eRam          = 0 ;
CmpFan.HtoT.B   = 0.325 ;
CmpH.HtoT.B     = 0.7 ;
CmpFan.dUtip.B  = 99.1978 ;
CmpH.dUtip.B    = -13.2367 ;
a_eRam.B        = 0 ;
```

```
% Control Variables - Minimum and Maximum Values to Explore
```

```
FPR_min          = 1.3 ;
FPR_max          = 1.7 ;
HPCPR_min        = 9 ;
HPCPR_max        = 16 ;
OPR_min          = 30 ;
OPR_max          = 55 ;
TKO.T4_min       = 3300 ;
TKO.T4_max       = 3750 ;
Ext.Ratio_min    = 0.9 ;
Ext.Ratio_max    = 1.25 ;
```

```

LPT.NChFracBldW_min      = 0 ;
LPT.NChFracBldW_max      = 0.06 ;
TKO.Thrust_min           = 20000 ;
TKO.Thrust_max           = 30000 ;
TOC.Thrust_min           = 5900 ;
TOC.Thrust_max           = 8850 ;
s_CmpH_Wc_Flare_min      = 1 ;
s_CmpH_Wc_Flare_max      = 1.2 ;
FPR.B_min                = 1.3 ;
FPR.B_max                 = 1.7 ;
OPR.B_min                 = 30 ;
OPR.B_max                 = 55 ;
TKO.T4.B_min              = 3300 ;
TKO.T4.B_max              = 3750 ;
Ext.Ratio.B_min           = 0.9 ;
Ext.Ratio.B_max           = 1.25 ;
LPT.NChFracBldW.B_min    = 0 ;
LPT.NChFracBldW.B_max    = 0.06 ;
TKO.HPT_NcMap.B_min      = 100 ;
TKO.HPT_NcMap.B_max      = 105 ;
ALPHA.CmpFan.deff_min     = 1 ;
ALPHA.CmpFan.deff_max    = 10 ;
ALPHA.CmpL.deff_min       = 1 ;
ALPHA.CmpL.deff_max       = 10 ;
ALPHA.CmpH.deff_min       = 1 ;
ALPHA.CmpH.deff_max       = 10 ;
ALPHA.TrbH.deff_min       = 1 ;
ALPHA.TrbH.deff_max       = 10 ;
ALPHA.TrbL.deff_min       = 1 ;
ALPHA.TrbL.deff_max       = 10 ;
ALPHA.CmpFan.deff.B_min   = 1 ;
ALPHA.CmpFan.deff.B_max   = 10 ;
ALPHA.CmpL.deff.B_min     = 1 ;
ALPHA.CmpL.deff.B_max     = 10 ;
ALPHA.CmpH.dPeff.B_min    = 1 ;

```

ALPHA.CmpH_dPeff_B_max	=	10	;
ALPHA.TrbH_deff_B_min	=	1	;
ALPHA.TrbH_deff_B_max	=	10	;
ALPHA.TrbL_deff_B_min	=	1	;
ALPHA.TrbL_deff_B_max	=	10	;
BETA.CmpFan_deff_min	=	1	;
BETA.CmpFan_deff_max	=	10	;
BETA.CmpL_deff_min	=	1	;
BETA.CmpL_deff_max	=	10	;
BETA.CmpH_deff_min	=	1	;
BETA.CmpH_deff_max	=	10	;
BETA.TrbH_deff_min	=	1	;
BETA.TrbH_deff_max	=	10	;
BETA.TrbL_deff_min	=	1	;
BETA.TrbL_deff_max	=	10	;
BETA.CmpFan_deff_B_min	=	1	;
BETA.CmpFan_deff_B_max	=	10	;
BETA.CmpL_deff_B_min	=	1	;
BETA.CmpL_deff_B_max	=	10	;
BETA.CmpH_dPeff_B_min	=	1	;
BETA.CmpH_dPeff_B_max	=	10	;
BETA.TrbH_deff_B_min	=	1	;
BETA.TrbH_deff_B_max	=	10	;
BETA.TrbL_deff_B_min	=	1	;
BETA.TrbL_deff_B_max	=	10	;
ALPHA.s_Wt_Des_min	=	1	;
ALPHA.s_Wt_Des_max	=	10	;
ALPHA.s_Wt_Var_min	=	1	;
ALPHA.s_Wt_Var_max	=	10	;
BETA.s_Wt_Des_min	=	1	;
BETA.s_Wt_Des_max	=	10	;
BETA.s_Wt_Var_min	=	1	;
BETA.s_Wt_Var_max	=	10	;
ALPHA.s_CmpHCustBld_min	=	1	;
ALPHA.s_CmpHCustBld_max	=	10	;

ALPHA.s_ShHHPX_min = 1 ;
ALPHA.s_ShHHPX_max = 10 ;
BETA.s_CmpHCustBld_min = 1 ;
BETA.s_CmpHCustBld_max = 10 ;
BETA.s_ShHHPX_min = 1 ;
BETA.s_ShHHPX_max = 10 ;
ALPHA.s_CmpHCustBld.B_min = 1 ;
ALPHA.s_CmpHCustBld.B_max = 10 ;
ALPHA.s_ShHHPX.B_min = 1 ;
ALPHA.s_ShHHPX.B_max = 10 ;
BETA.s_CmpHCustBld.B_min = 1 ;
BETA.s_CmpHCustBld.B_max = 10 ;
BETA.s_ShHHPX.B_min = 1 ;
BETA.s_ShHHPX.B_max = 10 ;
ALPHA.CmpLstg1MaxPR_min = 1 ;
ALPHA.CmpLstg1MaxPR_max = 10 ;
ALPHA.TrbH.Loading_min = 1 ;
ALPHA.TrbH.Loading_max = 10 ;
ALPHA.TrbL.Loading_min = 1 ;
ALPHA.TrbL.Loading_max = 10 ;
BETA.CmpLstg1MaxPR_min = 1 ;
BETA.CmpLstg1MaxPR_max = 10 ;
BETA.TrbH.Loading_min = 1 ;
BETA.TrbH.Loading_max = 10 ;
BETA.TrbL.Loading_min = 1 ;
BETA.TrbL.Loading_max = 10 ;
ALPHA.CmpLstg1MaxPR.B_min = 1 ;
ALPHA.CmpLstg1MaxPR.B_max = 10 ;
ALPHA.TrbH.Loading.B_min = 1 ;
ALPHA.TrbH.Loading.B_max = 10 ;
ALPHA.TrbL.Loading.B_min = 1 ;
ALPHA.TrbL.Loading.B_max = 10 ;
BETA.CmpLstg1MaxPR.B_min = 1 ;
BETA.CmpLstg1MaxPR.B_max = 10 ;
BETA.TrbH.Loading.B_min = 1 ;

```

BETA.TrbH>Loading_B_max      = 10 ;
BETA.TrbL>Loading_B_min      = 1  ;
BETA.TrbL>Loading_B_max      = 10 ;
ALPHA.T41_Max_min            = 1  ;
ALPHA.T41_Max_max            = 10 ;
BETA.T41_Max_min             = 1  ;
BETA.T41_Max_max             = 10 ;
ALPHA.T41_Max_B_min          = 1  ;
ALPHA.T41_Max_B_max          = 10 ;
BETA.T41_Max_B_min           = 1  ;
BETA.T41_Max_B_max           = 10 ;

```

% Noise Variables - Minimum and Maximum Values that Bound the Uncertainty

% Distributions

```

T41_Max_min                  = 3186.5;
T41_Max_max                  = 3300  ;
s_CmpHCustBld_min           = 0.5;
s_CmpHCustBld_max           = 2   ;
s_ShHHPX_min                 = 0.5;
s_ShHHPX_max                 = 2   ;
CmpLstg1MaxPR_min           = 1.28;
CmpLstg1MaxPR_max           = 1.536 ;
TrbH>Loading_min            = 0.339;
TrbH>Loading_max            = 0.4068 ;
TrbL>Loading_min            = 0.25;
TrbL>Loading_max            = 0.3  ;
CmpFan.deff_min              = 0.03195;
CmpFan.deff_max              = 0.04195 ;
CmpL.deff_min                = 0.02899;
CmpL.deff_max                = 0.03899 ;
CmpH.deff_min                = 0.001097;
CmpH.deff_max                = 0.011097 ;
TrbH.deff_min                = 0.00255;
TrbH.deff_max                = 0.01255 ;

```

```

TrbL.deff_min           = -0.0504;
TrbL.deff_max           = -0.0404 ;
s_eRam_min              = 0.99;
s_eRam_max               = 1   ;
s_Fan_Blade_rho_min     = 0.9;
s_Fan_Blade_rho_max     = 1   ;
s_Fan_Stator_rho_min    = 0.9;
s_Fan_Stator_rho_max    = 1   ;
s_CmpL_Blade_rho_min    = 0.9;
s_CmpL_Blade_rho_max    = 1   ;
s_CmpL_Stator_rho_min   = 0.9;
s_CmpL_Stator_rho_max   = 1   ;
s_CmpH_Blade_rho_min    = 0.9;
s_CmpH_Blade_rho_max    = 1   ;
s_CmpH_Stator_rho_min   = 0.9;
s_CmpH_Stator_rho_max   = 1   ;
s_CmpH_Blade2_rho_min   = 0.9;
s_CmpH_Blade2_rho_max   = 1   ;
s_CmpH_Stator2_rho_min  = 0.9;
s_CmpH_Stator2_rho_max  = 1   ;
s_TrbH_Blade_rho_min    = 0.9;
s_TrbH_Blade_rho_max    = 1   ;
s_TrbH_Stator_rho_min   = 0.9;
s_TrbH_Stator_rho_max   = 1   ;
s_TrbL_Blade_rho_min    = 0.9;
s_TrbL_Blade_rho_max    = 1   ;
s_TrbL_Stator_rho_min   = 0.9;
s_TrbL_Stator_rho_max   = 1   ;
s_NozPri_Wt_min         = 0.9;
s_NozPri_Wt_max         = 1   ;
s_NozSec_Wt_min         = 0.9;
s_NozSec_Wt_max         = 1   ;
s_ShL_Wt_min            = 0.9;
s_ShL_Wt_max            = 1   ;
s_ShH_Wt_min            = 0.9;

```



```

s_ShH_Wt_max           = 1 ;
T41_Max_B_min         = 3186.5;
T41_Max_B_max         = 3300 ;
s_CmpHCustBld_B_min   = 0.5;
s_CmpHCustBld_B_max   = 2 ;
s_ShHHPX_B_min        = 0.5;
s_ShHHPX_B_max        = 2 ;
CmpLstg1MaxPR_B_min   = 1.28;
CmpLstg1MaxPR_B_max   = 1.536 ;
TrbH_Loading_B_min    = 0.339;
TrbH_Loading_B_max    = 0.4068 ;
TrbL_Loading_B_min    = 0.25;
TrbL_Loading_B_max    = 0.3 ;
CmpFan_deff_B_min     = 0.03195;
CmpFan_deff_B_max     = 0.04195 ;
CmpL_deff_B_min       = 0.02899;
CmpL_deff_B_max       = 0.03899 ;
CmpH_dPeff_B_min      = 0;
CmpH_dPeff_B_max      = 0.01 ;
TrbH_deff_B_min       = 0;
TrbH_deff_B_max       = 0.01 ;
TrbL_deff_B_min       = -0.0504;
TrbL_deff_B_max       = -0.0404 ;
s_eRam_B_min          = 0.99;
s_eRam_B_max          = 1 ;
s_Fan_Blade_rho_B_min = 0.9;
s_Fan_Blade_rho_B_max = 1 ;
s_Fan_Stator_rho_B_min = 0.9;
s_Fan_Stator_rho_B_max = 1 ;
s_CmpL_Blade_rho_B_min = 0.9;
s_CmpL_Blade_rho_B_max = 1 ;
s_CmpL_Stator_rho_B_min = 0.9;
s_CmpL_Stator_rho_B_max = 1 ;
s_CmpH_Blade_rho_B_min = 0.9;
s_CmpH_Blade_rho_B_max = 1 ;

```

```

s_CmpH_Stator_rho_B_min    = 0.9;
s_CmpH_Stator_rho_B_max    = 1   ;
s_CmpH_Blade2_rho_B_min    = 0.9;
s_CmpH_Blade2_rho_B_max    = 1   ;
s_CmpH_Stator2_rho_B_min   = 0.9;
s_CmpH_Stator2_rho_B_max   = 1   ;
s_TrbH_Blade_rho_B_min     = 0.9;
s_TrbH_Blade_rho_B_max     = 1   ;
s_TrbH_Stator_rho_B_min    = 0.9;
s_TrbH_Stator_rho_B_max    = 1   ;
s_TrbL_Blade_rho_B_min     = 0.9;
s_TrbL_Blade_rho_B_max     = 1   ;
s_TrbL_Stator_rho_B_min    = 0.9;
s_TrbL_Stator_rho_B_max    = 1   ;
s_NozPri_Wt_B_min          = 0.9;
s_NozPri_Wt_B_max          = 1   ;
s_NozSec_Wt_B_min          = 0.9;
s_NozSec_Wt_B_max          = 1   ;
s_ShL_Wt_B_min             = 0.9;
s_ShL_Wt_B_max             = 1   ;
s_ShH_Wt_B_min             = 0.9;
s_ShH_Wt_B_max             = 1   ;

```

```

%% SETUP THE EXPLORATION TO BE PERFORMED

```

```

%Normalized lhs:

```

```

nLhs_total=3000; % Number of Unique Explorative Samples
nLhs = 50; % Number of samples considered at a time, prevents memory issues
num_batches = nLhs_total/nLhs;
nCvs = 63; % Number of CVs to explore
nSURRs = 103; % Number of surrogate models being evaluated

```

```

% Generate latin hypercube doe

```

```

disp('Starting Latin Hypercube Generation!');

```

```

lhsNonDim = lhsdesign(nLhs,nCVs,'iterations',10000);
disp('Finished Latin Hypercube Generation!');

% Set Minimum and Maximum Values to Control Variables
indep_mins=[FPR_min HPCPR_min OPR_min TKO_T4_min ...

indep_maxs=[FPR_max HPCPR_max OPR_max TKO_T4_max ...

%Convert normalized latin hypercube settings to actual CV values:
lhsDim=bsxfun(@times,lhsNonDim,indep_maxs-indep_mins);
lhsDim=bsxfun(@plus,lhsDim,indep_mins);

OutMat = zeros(nLhs_total,91);%104);

%% Set Control Variable Values for each
%DoE Case in order to Evaluate Surrogates
for k = 1:num_batches
start_case = 1+(k-1)*nLhs;
end_case = k*nLhs;
%i=1:nLhs;
i=start_case:end_case;
disp(['<<<-- Initializing Batch ' num2str(k) ' out of ' ...
      num2str(num_batches) ', cases ' num2str(start_case) ' through ' ...
      num2str(end_case) '. -->>']);
%Variables:
ii = 1;

disp('Setting Control Variable Values for all DoE Cases!');

% CONTROL VARIABLES - Apply lhs dimensionalize values to CVs
FPR = lhsDim(i,ii); ii = ii+1;
HPCPR = lhsDim(i,ii); ii = ii+1;

```

```

OPR                = lhsDim(i,ii); ii = ii+1;
TKO.T4            = lhsDim(i,ii); ii = ii+1;
ALPHA.CmpFan.deff = lhsDim(i,ii); ii = ii+1;
BETA.CmpFan.deff  = lhsDim(i,ii); ii = ii+1;
% ! Remainder of CVs not included for brevity of Appendix

%%
%-----
%Perform Monte-Carlo Probabilistic Analysis
%-----

% Number of times each LHS design is repeated
% to account for uncertainty distributions
numMonteCarlo = 50000;

% Expand CVs out to vectors of all repetitions of all designs:
disp('Expanding CVs for all Monte Carlo Points for Each DoE Case!');

% CONTROL VARIABLES
FPR                = repmat(FPR',numMonteCarlo,1);
HPCPR              = repmat(HPCPR',numMonteCarlo,1);
OPR                = repmat(OPR',numMonteCarlo,1);
TKO.T4            = repmat(TKO.T4',numMonteCarlo,1);
ALPHA.CmpFan.deff = repmat(ALPHA.CmpFan.deff',numMonteCarlo,1);
BETA.CmpFan.deff  = repmat(BETA.CmpFan.deff',numMonteCarlo,1);
% ! Remainder of CVs not included for brevity of Appendix

disp('Transforming CV Arrays in Preparation of Simulations!');
% TRANSPOSE CONTROL VARIABLE VECTORS
FPR                = FPR(:);
HPCPR              = HPCPR(:);

```

```

OPR                =   OPR(:);
TKO_T4             =   TKO_T4(:);
ALPHA_CmpFan_deff =   ALPHA_CmpFan_deff(:);
BETA_CmpFan_deff  =   BETA_CmpFan_deff(:);
% ! Remainder of CVs not included for brevity of Appendix

%% Set Noise Variable Values for Each DoE Case

disp('Setting Noise Variable Shape Parameters!');
% Alpha and Beta Shape Parameter Values - Some Fixed and Some are CVs
alp_T41_Max        =   ALPHA_T41_Max   ;
bet_T41_Max        =   BETA_T41_Max    ;
alp_s_CmpHCustBld =   ALPHA_s_CmpHCustBld ;
bet_s_CmpHCustBld =   BETA_s_CmpHCustBld ;
alp_s_ShHHPX      =   ALPHA_s_ShHHPX   ;
bet_s_ShHHPX      =   BETA_s_ShHHPX    ;
alp_CmpLstg1MaxPR =   ALPHA_CmpLstg1MaxPR ;
bet_CmpLstg1MaxPR =   BETA_CmpLstg1MaxPR ;
% ! Remainder of NVs not included for brevity of Appendix

disp(['Constructing Noise Variable Distributions for All DoE Cases']);
% Set Noise Variable Values for every Repetition of Every Design
T41_Max            =   T41_Max_min +( T41_Max_max - T41_Max_min ) ...
    *betarnd( alp_T41_Max , bet_T41_Max );
s_CmpHCustBld      =   s_CmpHCustBld_min +( s_CmpHCustBld_max - ...
    s_CmpHCustBld_min )*betarnd( alp_s_CmpHCustBld , bet_s_CmpHCustBld );
s_ShHHPX           =   s_ShHHPX_min +( s_ShHHPX_max - s_ShHHPX_min )...
    *betarnd( alp_s_ShHHPX , bet_s_ShHHPX );
CmpLstg1MaxPR      =   CmpLstg1MaxPR_min +( CmpLstg1MaxPR_max - ...
    CmpLstg1MaxPR_min )*betarnd( alp_CmpLstg1MaxPR , bet_CmpLstg1MaxPR );
TrbH_Loading       =   TrbH_Loading_min +( TrbH_Loading_max - ...
    TrbH_Loading_min )*betarnd( alp_TrbH_Loading , bet_TrbH_Loading );

```

```

TrbL.Loading          = TrbL.Loading_min +( TrbL.Loading_max - ...
    TrbL.Loading_min )*betarnd( alp_TrbL.Loading , bet_TrbL.Loading );
% ! Remainder of NVs not included for brevity of Appendix

%% Fixed NV Distributions
s_eRam = s_eRam_min +( s_eRam_max - s_eRam_min )*betarnd( alp_s_eRam ...
    , bet_s_eRam , numMonteCarlo,1);
s_eRam.B = s_eRam.B_min +( s_eRam.B_max - s_eRam.B_min )*...
    betarnd( alp_s_eRam.B , bet_s_eRam.B , numMonteCarlo,1);

% Duplicate Fixed NV Distributions for Every DoE Case
s_eRam          = repmat( s_eRam',1,nLhs);
s_eRam.B        = repmat( s_eRam.B',1,nLhs);

% Transpose Fixed NV Distribution Vectors
s_eRam          = s_eRam(:);
s_eRam.B        = s_eRam.B(:);

%% Reshape Every Input Variable, CVs and NVs

disp('Reshaping all Input Variables to Prepare for DoE Simulation Loop!');
% Control Variables
FPR              =reshape( FPR , numMonteCarlo,nLhs);
HPCPR            =reshape( HPCPR , numMonteCarlo,nLhs);
OPR              =reshape( OPR , numMonteCarlo,nLhs);
TKO.T4           =reshape( TKO.T4 , numMonteCarlo,nLhs);
ALPHA.CmpFan.deff =reshape( ALPHA.CmpFan.deff, numMonteCarlo,nLhs);
BETA.CmpFan.deff  =reshape( BETA.CmpFan.deff, numMonteCarlo,nLhs);
% ! Remainder of CVs not included for brevity of Appendix

% Noise Variables
T41.Max          =reshape( T41.Max , numMonteCarlo,nLhs);
s_CmpHCustBld    =reshape( s_CmpHCustBld , numMonteCarlo,nLhs);

```

```

s_ShHHPX          =reshape(  s_ShHHPX      ,numMonteCarlo,nLhs);
CmpLstg1MaxPR     =reshape(  CmpLstg1MaxPR   ,numMonteCarlo,nLhs);
TrbH.Loading      =reshape(  TrbH.Loading   ,numMonteCarlo,nLhs);
TrbL.Loading      =reshape(  TrbL.Loading   ,numMonteCarlo,nLhs);
CmpFan.deff       =reshape(  CmpFan.deff   ,numMonteCarlo,nLhs);
% ! Remainder of NVs not included for brevity of Appendix

%% SWEEP THROUGH THE DOE DESIGNS AND EVALUATE DETERMINISTIC SURROGATES FOR
% EVERY REPETITION, ELIMINATING REPS THAT ARE INFEASIBLE ACCORDING TO SVMSTRUCT

disp(['Performing Monte Carlo Simulations for Batch ' num2str(k) ' of ' ...
      num2str(num_batches)]);

for j = 1:nLhs
    caseNum = start_case+j-1;
    disp(['-- Starting Design ' num2str(caseNum) ' Analysis! --']);

    SVMtestInputArray(:,1) = FPR(:,j);
    SVMtestInputArray(:,2) = HPCPR(:,j);
    SVMtestInputArray(:,3) = OPR(:,j);
    SVMtestInputArray(:,4) = TKO_T4(:,j);
    SVMtestInputArray(:,5) = Ext_Ratio(:,j);
    SVMtestInputArray(:,6) = T41_Max(:,j);
    SVMtestInputArray(:,7) = LPT_NChFracBldW(:,j);
    SVMtestInputArray(:,8) = TKO_Thrust(:,j);
    SVMtestInputArray(:,9) = TOC_Thrust(:,j);
    SVMtestInputArray(:,10) = s_CmpHCustBld(:,j);
    SVMtestInputArray(:,11) = s_ShHHPX(:,j);
    SVMtestInputArray(:,12) = TrbH.Loading(:,j);
    SVMtestInputArray(:,13) = TrbL.Loading(:,j);
    SVMtestInputArray(:,14) = CmpFan.deff(:,j);
    SVMtestInputArray(:,15) = CmpL.deff(:,j);
    SVMtestInputArray(:,16) = CmpH.deff(:,j);
    SVMtestInputArray(:,17) = TrbH.deff(:,j);

```

```

SVMtestInputArray(:,18) = TrbL_deff(:,j);
SVMtestInputArray(:,19) = s_eRam(:,j);
SVMtestInputArray(:,20) = s_CmpH_Wc_Flare(:,j);
SVMtestInputArray(:,21) = FPR_B(:,j);
SVMtestInputArray(:,22) = OPR_B(:,j);
SVMtestInputArray(:,23) = TKO_T4_B(:,j);
SVMtestInputArray(:,24) = Ext_Ratio_B(:,j);
SVMtestInputArray(:,25) = T41_Max_B(:,j);
SVMtestInputArray(:,26) = LPT_NChFracBldW_B(:,j);
SVMtestInputArray(:,27) = TKO_HPT_NcMap_B(:,j);
SVMtestInputArray(:,28) = s_CmpHCustBld_B(:,j);
SVMtestInputArray(:,29) = s_ShHHPX_B(:,j);
SVMtestInputArray(:,30) = TrbH_Loading_B(:,j);
SVMtestInputArray(:,31) = TrbL_Loading_B(:,j);
SVMtestInputArray(:,32) = CmpFan_deff_B(:,j);
SVMtestInputArray(:,33) = CmpL_deff_B(:,j);
SVMtestInputArray(:,34) = CmpH_dPeff_B(:,j);
SVMtestInputArray(:,35) = TrbH_deff_B(:,j);
SVMtestInputArray(:,36) = TrbL_deff_B(:,j);
SVMtestInputArray(:,37) = s_eRam_B(:,j);

% Determine number of design repetitions expected to be infeasible
% according to SVM model
GROUP = svmclassify(SVMSTRUCT, SVMtestInputArray);
GROUP = reshape(GROUP,numMonteCarlo,[]);
GROUP = logical(GROUP);
passcount = sum(GROUP);
% Used to scale confidence intervals for which probabilistic
% performance estimates will be provided
ScaleFactor = passcount/numel(GROUP);

disp(['Design ' num2str(caseNum) ' had ' num2str(passcount) ...
      ' passes out of ' num2str(numMonteCarlo) ' repetitions. ' ...
      num2str(passcount/numMonteCarlo*100) '% Success rate.']);

```



```

% Eliminate all repetitions that are not feasible according to
% SVMSTRUCT

FPR_col           = FPR(GROUP, j);
HPCPR_col        = HPCPR(GROUP, j);
OPR_col          = OPR(GROUP, j);
TKO-T4_col       = TKO-T4(GROUP, j);
ALPHA-CmpFan-deff_col = ALPHA-CmpFan-deff(GROUP, j);
ALPHA-CmpFan-deff_B_col = ALPHA-CmpFan-deff_B(GROUP, j);
BETA-CmpFan-deff_col = BETA-CmpFan-deff(GROUP, j);
BETA-CmpFan-deff_B_col = BETA-CmpFan-deff_B(GROUP, j);
T41_Max_col      = T41_Max(GROUP, j);
s-CmpHCustBld_col = s-CmpHCustBld(GROUP, j);
s-ShHHPX_col     = s-ShHHPX(GROUP, j);
CmpLstg1MaxPR_col = CmpLstg1MaxPR(GROUP, j);
T41_Max_B_col    = T41_Max_B(GROUP, j);
s-CmpHCustBld_B_col = s-CmpHCustBld_B(GROUP, j);
s-ShHHPX_B_col   = s-ShHHPX_B(GROUP, j);
CmpLstg1MaxPR_B_col = CmpLstg1MaxPR_B(GROUP, j);

% ! Remainder of input variables not included for brevity of Appendix

disp('Beginning Surrogate Evaluation!');

quants=[0.01 0.05 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 0.95 0.99];

% Inputted variables for each deterministic surrogate evaluation
% The variable 'INP_ARRAY' was not used in the actual script, it is
% just declared here to shorten the code provided in this appendix
INP_ARRAY=(FPR_col,HPCPR_col,OPR_col,TKO-T4_col,Ext_Ratio_col,...
           T41_Max_col,LPT.NChFracBldW_col,TKO.Thrust_col,TOC.Thrust_col,...
           s-CmpHCustBld_col,s-ShHHPX_col,CmpLstg1MaxPR_col,...
           TrbH.Loading_col,TrbL.Loading_col,CmpFan-deff_col,CmpL.deff_col,...
           CmpH.deff_col,TrbH.deff_col,TrbL.deff_col,s.eRam_col,...

```

```

s_Fan.Blade_rho_col, s_Fan.Stator_rho_col, s_CmpL.Blade_rho_col, ...
s_CmpL.Stator_rho_col, s_CmpH.Blade_rho_col, ...
s_CmpH.Stator_rho_col, s_CmpH.Blade2_rho_col, ...
s_CmpH.Stator2_rho_col, s_TrbH.Blade_rho_col, ...
s_TrbH.Stator_rho_col, s_TrbL.Blade_rho_col, ...
s_TrbL.Stator_rho_col, s_NozPri.Wt_col, s_NozSec.Wt_col, ...
s_ShL.Wt_col, s_ShH.Wt_col, s_CmpH.Wc_Flare_col, FPR.B_col, ...
OPR.B_col, TKO_T4.B_col, Ext.Ratio.B_col, T41.Max.B_col, ...
LPT.NChFracBldW.B_col, TKO.HPT.NcMap.B_col, s_CmpHCustBld.B_col, ...
s_ShHPX.B_col, CmpLstg1MaxPR.B_col, TrbH.Loading.B_col, ...
TrbL.Loading.B_col, CmpFan.deff.B_col, CmpL.deff.B_col, ...
CmpH.dPeff.B_col, TrbH.deff.B_col, TrbL.deff.B_col, s_eRam.B_col, ...
s_Fan.Blade_rho.B_col, s_Fan.Stator_rho.B_col, ...
s_CmpL.Blade_rho.B_col, s_CmpL.Stator_rho.B_col, ...
s_CmpH.Blade_rho.B_col, s_CmpH.Stator_rho.B_col, ...
s_CmpH.Blade2_rho.B_col, s_CmpH.Stator2_rho.B_col, ...
s_TrbH.Blade_rho.B_col, s_TrbH.Stator_rho.B_col, ...
s_TrbL.Blade_rho.B_col, s_TrbL.Stator_rho.B_col, ...
s_NozPri.Wt.B_col, s_NozSec.Wt.B_col, s_ShL.Wt.B_col, s_ShH.Wt.B_col);

% Evaluate Deterministic surrogate models predicting core defining
% design engine performance
A_TKO_HPC_Nc      =  A_TKO_HPC_Nc(INP_ARRAY) ;
qA_TKO_HPC_Nc    =quantile( A_TKO_HPC_Nc      ,quants);
clear A_TKO_HPC_Nc

A_TKO_HPT_Nc      =  A_TKO_HPT_Nc(INP_ARRAY) ;
qA_TKO_HPT_Nc    =quantile( A_TKO_HPT_Nc      ,quants);
clear A_TKO_HPT_Nc

A_TKO_ShH_Nmech   =  A_TKO_ShH_Nmech(INP_ARRAY) ;
qA_TKO_ShH_Nmech =quantile( A_TKO_ShH_Nmech ,quants);
clear A_TKO_ShH_Nmech

A_TKO_ShH_Pwr     =  A_TKO_ShH_Pwr(INP_ARRAY) ;

```

```

qA.TKO_ShH_Pwr =quantile( A.TKO_ShH_Pwr ,quants);
clear A.TKO_ShH_Pwr

A.TKO_ShL_Nmech = A.TKO_ShL_Nmech (INP_ARRAY) ;
qA.TKO_ShL_Nmech =quantile( A.TKO_ShL_Nmech ,quants);
clear A.TKO_ShL_Nmech

A.TKO_ShL_Pwr = A.TKO_ShL_Pwr (INP_ARRAY) ;
qA.TKO_ShL_Pwr =quantile( A.TKO_ShL_Pwr ,quants);
clear A.TKO_ShL_Pwr

A.TSFC_ADP = A.TSFC_ADP (INP_ARRAY) ;
qA.TSFC_ADP =quantile( A.TSFC_ADP ,quants);
clear A.TSFC_ADP

A.Pod_Wt = A.Pod_Wt (INP_ARRAY) ;
qA.Pod_Wt =quantile( A.Pod_Wt ,quants);
clear A.Pod_Wt

A.Fan_Diam = A.Fan_Diam (INP_ARRAY) ;
qA.Fan_Diam =quantile( A.Fan_Diam ,quants);
clear A.Fan_Diam

A.ADP_Wc3 = A.ADP_Wc3 (INP_ARRAY) ;
qA.ADP_Wc3 =quantile( A.ADP_Wc3 ,quants);
clear A.ADP_Wc3

% ! Remainder of Core Defining Design Engine Surrogate Evaluations
% Removed for Appendix Brevity

% Evaluate Deterministic surrogate models predicting common core
% variant engine performance

B.BPR_ADP = B.BPR_ADP (INP_ARRAY) ;
qB.BPR_ADP =quantile( B.BPR_ADP ,quants);

```

```

clear B.BPR_ADP

B.Fn_ADP          = B.Fn_ADP (INP_ARRAY) ;
qB.Fn_ADP        =quantile( B.Fn_ADP    ,quants);
clear B.Fn_ADP

B.Fn_TKO          = B.Fn_TKO (INP_ARRAY) ;
qB.Fn_TKO        =quantile( B.Fn_TKO    ,quants);
clear B.Fn_TKO

B.Fn_TOC          = B.Fn_TOC (INP_ARRAY) ;
qB.Fn_TOC        =quantile( B.Fn_TOC    ,quants);
clear B.Fn_TOC

B.Min.hook.NcMap  = B.Min.hook.NcMap (INP_ARRAY) ;
qB.Min.hook.NcMap =quantile( B.Min.hook.NcMap ,quants);
clear B.Min.hook.NcMap

B.Min.hook.TSFC   = B.Min.hook.TSFC (INP_ARRAY) ;
qB.Min.hook.TSFC =quantile( B.Min.hook.TSFC ,quants);
clear B.Min.hook.TSFC

B.T3_TKO          = B.T3_TKO (INP_ARRAY) ;
qB.T3_TKO        =quantile( B.T3_TKO    ,quants);
clear B.T3_TKO

B.T4_TKO          = B.T4_TKO (INP_ARRAY) ;
qB.T4_TKO        =quantile( B.T4_TKO    ,quants);
clear B.T4_TKO

B.T41_TKO         = B.T41_TKO (INP_ARRAY) ;
qB.T41_TKO       =quantile( B.T41_TKO    ,quants);
clear B.T41_TKO

B.T45_TKO         = B.T45_TKO (INP_ARRAY) ;

```

```

qB.T45_TKO =quantile( B.T45_TKO ,quants);
clear B.T45_TKO

B.TKO_HPC_Nc = B.TKO_HPC_Nc(INP_ARRAY) ;
qB.TKO_HPC_Nc =quantile( B.TKO_HPC_Nc ,quants);
clear B.TKO_HPC_Nc

B.TKO_HPT_Nc = B.TKO_HPT_Nc(INP_ARRAY) ;
qB.TKO_HPT_Nc =quantile( B.TKO_HPT_Nc ,quants);
clear B.TKO_HPT_Nc

B.TKO_ShH_Nmech = B.TKO_ShH_Nmech(INP_ARRAY) ;
qB.TKO_ShH_Nmech =quantile( B.TKO_ShH_Nmech ,quants);
clear B.TKO_ShH_Nmech

B.TKO_ShH_Pwr = B.TKO_ShH_Pwr(INP_ARRAY) ;
qB.TKO_ShH_Pwr =quantile( B.TKO_ShH_Pwr ,quants);
clear B.TKO_ShH_Pwr

B.TKO_ShL_Nmech = B.TKO_ShL_Nmech(INP_ARRAY) ;
qB.TKO_ShL_Nmech =quantile( B.TKO_ShL_Nmech ,quants);
clear B.TKO_ShL_Nmech

B.TKO_ShL_Pwr = B.TKO_ShL_Pwr(INP_ARRAY) ;
qB.TKO_ShL_Pwr =quantile( B.TKO_ShL_Pwr ,quants);
clear B.TKO_ShL_Pwr

B.TSFC_ADP = B.TSFC_ADP(INP_ARRAY) ;
qB.TSFC_ADP =quantile( B.TSFC_ADP ,quants);
clear B.TSFC_ADP

B.TSFC_TKO = B.TSFC_TKO(INP_ARRAY) ;
qB.TSFC_TKO =quantile( B.TSFC_TKO ,quants);
clear B.TSFC_TKO

```

```

B_TSFC_TOC          = B_TSFC_TOC(INP_ARRAY) ;
qB_TSFC_TOC =quantile( B_TSFC_TOC ,quants);
clear B_TSFC_TOC

B_Pod_Wt           = B_Pod_Wt(INP_ARRAY) ;
qB_Pod_Wt  =quantile( B_Pod_Wt ,quants);
clear B_Pod_Wt

B_Fan_Diam         = B_Fan_Diam(INP_ARRAY) ;
qB_Fan_Diam =quantile( B_Fan_Diam ,quants);
clear B_Fan_Diam

% ! Remainder of Common Core Variant Engine Surrogate Evaluations
% Removed for Appendix Brevity

disp('pause');

%Scale Confidence Intervals for each design according to percentage of
%repetitions that are expected to be feasible
quants = quants*ScaleFactor;

%Save I/O data for design samples
OutMat(caseNum,:) = [caseNum FPR_col(1) HPCPR_col(1) OPR_col(1) ...
    TKO_T4_col(1) Ext_Ratio_col(1) LPT_NChFracBldW_col(1) ...
    TKO_Thrust_col(1) TOC_Thrust_col(1) s_CmpH_Wc_Flare_col(1) ...
    FPR_B_col(1) OPR_B_col(1) TKO_T4_B_col(1) Ext_Ratio_B_col(1) ...
    LPT_NChFracBldW_B_col(1) TKO_HPT_NcMap_B_col(1) ...
    ALPHA_CmpFan_deff_col(1) ALPHA_CmpL_deff_col(1) ...
    ALPHA_CmpH_deff_col(1) ALPHA_TrbH_deff_col(1) ...
    ALPHA_TrbL_deff_col(1) ALPHA_CmpFan_deff_B_col(1) ...
    ALPHA_CmpL_deff_B_col(1) ALPHA_CmpH_dPeff_B_col(1) ...
    ALPHA_TrbH_deff_B_col(1) ALPHA_TrbL_deff_B_col(1) ...
    BETA_CmpFan_deff_col(1) BETA_CmpL_deff_col(1) ...
    BETA_CmpH_deff_col(1) BETA_TrbH_deff_col(1) ...
    BETA_TrbL_deff_col(1) BETA_CmpFan_deff_B_col(1) ...

```

```

BETA_CmpL_deff_B_col(1) BETA_CmpH_dPeff_B_col(1) ...
BETA_TrbH_deff_B_col(1) BETA_TrbL_deff_B_col(1) ...
ALPHA_s_Wt_Des_col(1) ALPHA_s_Wt_Var_col(1) BETA_s_Wt_Des_col(1) ...
BETA_s_Wt_Var_col(1) ALPHA_s_CmpHCustBld_col(1) ...
ALPHA_s_ShHHPX_col(1) BETA_s_CmpHCustBld_col(1) ...
BETA_s_ShHHPX_col(1) ALPHA_s_CmpHCustBld_B_col(1) ...
ALPHA_s_ShHHPX_B_col(1) BETA_s_CmpHCustBld_B_col(1) ...
BETA_s_ShHHPX_B_col(1) ALPHA_CmpLstg1MaxPR_col(1) ...
ALPHA_TrbH_Loading_col(1) ALPHA_TrbL_Loading_col(1) ...
BETA_CmpLstg1MaxPR_col(1) BETA_TrbH_Loading_col(1) ...
BETA_TrbL_Loading_col(1) ALPHA_CmpLstg1MaxPR_B_col(1) ...
ALPHA_TrbH_Loading_B_col(1) ALPHA_TrbL_Loading_B_col(1) ...
BETA_CmpLstg1MaxPR_B_col(1) BETA_TrbH_Loading_B_col(1) ...
BETA_TrbL_Loading_B_col(1) ALPHA_T41_Max_col(1) ...
BETA_T41_Max_col(1) ALPHA_T41_Max_B_col(1) ...
BETA_T41_Max_B_col(1) ScaleFactor quants qA_ADp_Wc3];

```

end

end

```

disp('Monte Carlo Simulations Complete!! Writing data to excel sheet.');
```

```

filename = 'Prob_Surr_Training_Data.xlsx';
namez={'caseNum', 'FPR', 'HPCPR', 'OPR', 'TKO_T4', 'Ext_Ratio', ...
      'LPT_NChFracBldW', 'TKO_Thrust', 'TOC_Thrust', ...
      's_CmpH_Wc_Flare', 'FPR_B', 'OPR_B', 'TKO_T4_B', ...
      'Ext_Ratio_B', 'LPT_NChFracBldW_B', 'TKO_HPT_NcMap_B', ...
      'ALPHA_CmpFan_deff', 'ALPHA_CmpL_deff', 'ALPHA_CmpH_deff', ...
      'ALPHA_TrbH_deff', 'ALPHA_TrbL_deff', 'ALPHA_CmpFan_deff_B', ...
      'ALPHA_CmpL_deff_B', 'ALPHA_CmpH_dPeff_B', 'ALPHA_TrbH_deff_B', ...
      'ALPHA_TrbL_deff_B', 'BETA_CmpFan_deff', 'BETA_CmpL_deff', ...
      'BETA_CmpH_deff', 'BETA_TrbH_deff', 'BETA_TrbL_deff', ...
      'BETA_CmpFan_deff_B', 'BETA_CmpL_deff_B', 'BETA_CmpH_dPeff_B', ...
      'BETA_TrbH_deff_B', 'BETA_TrbL_deff_B', 'ALPHA_s_Wt_Des', ...
      'ALPHA_s_Wt_Var', 'BETA_s_Wt_Des', 'BETA_s_Wt_Var', ...

```

```

'ALPHA_s_CmpHCustBld', 'ALPHA_s_ShHHPX', 'BETA_s_CmpHCustBld', ...
'BETA_s_ShHHPX', 'ALPHA_s_CmpHCustBld_B', 'ALPHA_s_ShHHPX_B', ...
'BETA_s_CmpHCustBld_B', 'BETA_s_ShHHPX_B', ...
'ALPHA_CmpLstg1MaxPR', 'ALPHA_TrbH.Loading', ...
'ALPHA_TrbL.Loading', 'BETA_CmpLstg1MaxPR', 'BETA_TrbH.Loading', ...
'BETA_TrbL.Loading', 'ALPHA_CmpLstg1MaxPR_B', ...
'ALPHA_TrbH.Loading_B', 'ALPHA_TrbL.Loading_B', ...
'BETA_CmpLstg1MaxPR_B', 'BETA_TrbH.Loading_B', ...
'BETA_TrbL.Loading_B', 'ALPHA_T41_Max', 'BETA_T41_Max', ...
'ALPHA_T41_Max_B', 'BETA_T41_Max_B', 'ScaleFactor', 'quants', ...
'quants', 'quants', 'quants', 'quants', 'quants', 'quants', ...
'quants', 'quants', 'quants', 'quants', 'quants', 'quants', ...
'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', ...
'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', ...
'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', 'qA_ADP_Wc3', ...
'qA_ADP_Wc3'};

```

```

xlswrite(filename, namez, 1, 'A1');
xlswrite(filename, OutMat, 1, 'A2');

```


APPENDIX C

DETERMINISTIC SURROGATE MODEL FIT STATISTICS

C.1 Overview

Mathematical representations of the physics based common core engine model were created. These deterministic surrogate models are in the form of Artificial Neural Network (ANN) models, and are functions of the control and noise variables. The ANN architectures were explored in order to find the number of hidden nodes that offered the best model fit quality within the imposed training time limits. The training of these surrogate models enable extensive exploration of the common core design and technology space without the need for additional simulations to be performed within the confines of the physics based engine model, greatly reducing the computational burden of subsequent simulations. The models constructed are used to evaluate the 150 million Monte Carlo simulations (3,000 DoE cases each with 50,000 simulation repetitions) performed by the script found in Appendix B.

For a given Artificial Neural Network (ANN) model being trained, the model architecture was explored. The number of hidden nodes of the ANN model form were explored, and the architecture with the best model fit quality achieved within the time limit considered was selected. A surrogate model's quality of fit is evaluated base on the following metrics. This is true for the deterministic surrogate models trained and shown in this appendix. These quality characteristics are also required of the probabilistic surrogate models trained, whose plots of fit quality are displayed in Appendix D.

- Model Fit Error (MFE), the distribution of surrogate model error that is determined through the comparison of predicted performance to the actual performance levels used in the training of the surrogate model
 - Shape of MFE distribution should resemble a normal distribution
 - Mean value of MFE distribution should be close to zero

– MFE standard deviation (STD) should be less than one

- Model Representation Error (MRE), the distribution of surrogate model error that is determined through the comparison of predicted performance to the actual performance levels of the reserve data not used in the training of the surrogate model. MRE quality determined based on the same metrics used in the evaluation of the MFE distribution.
- The quotient of MRE and MFE standard deviations should have a value less than two.
- A high coefficient of determination (R^2) must be achieved, having a value greater than 0.99. A value close to one indicates the model is able to explain the vast majority of variation in the metric response of interest. The model's R^2 can be determined by referring back to Equation 16 provided in Chapter 3, which is repeated below.
- When visualizing the actual-by-predicted plot, the data should not contain tails where actual-by-predicted values depart from the perfect fit line at low and high values of response data.
- The residual error should be at least two orders of magnitude less than the actual response levels.

$$R^2 = 1 - \frac{SS_{error}}{SS_{total}} \quad (16a \text{ Revisited})$$

$$SS = \sum (Y - \bar{Y})^2 \quad (16b \text{ Revisited})$$

$$\bar{Y} = \frac{\sum Y}{N} \quad (16c \text{ Revisited})$$

The following two sections contain figures visualizing the fit quality and statistics of each surrogate model of interest. Models were trained for both the core defining engine characteristics as well as the common core engine variant design. The core defining engine surrogates are functions of the control and noise variables defining the design engine. However, the common core engine surrogates are functions of the control and noise variables

corresponding to both the design and variant engines. This is due to the common core definition originating from the design engine and being applied to the variant engine.

C.2 Core Defining Engine - Fit Statistics

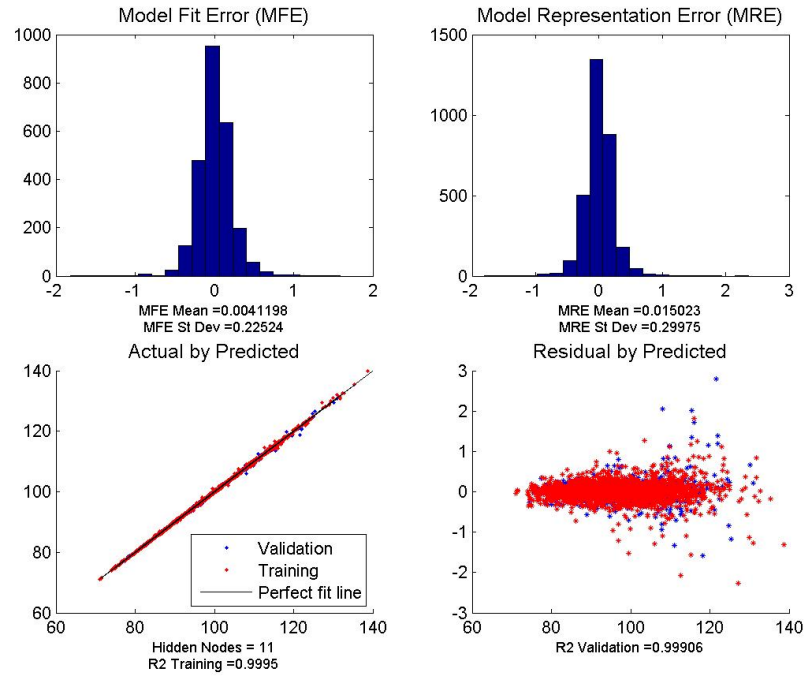


Figure 109: Core Defining Engine - Deterministic model fit statistics - LPC inlet corrected flow at ADP.

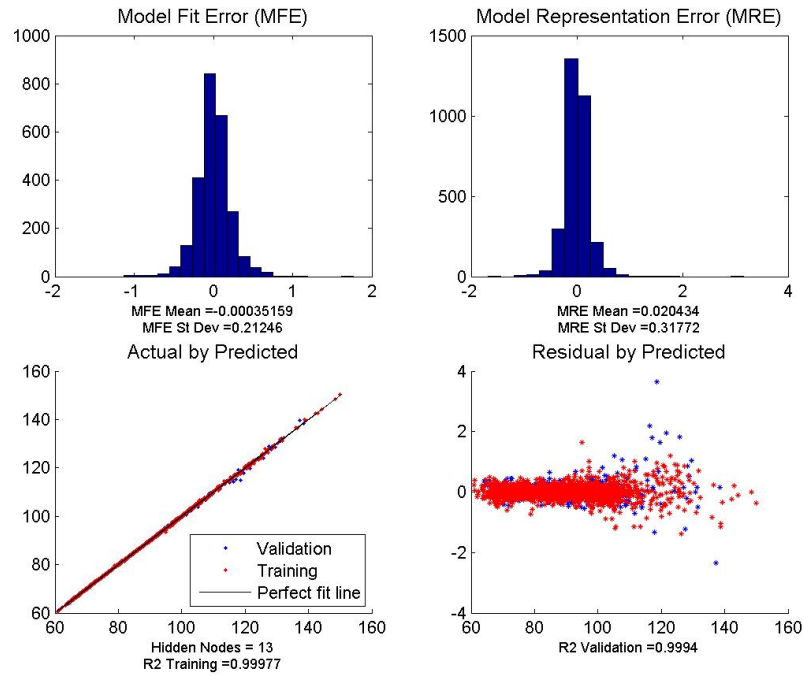


Figure 110: Core Defining Engine - Deterministic model fit statistics - LPC inlet corrected flow at TKO.

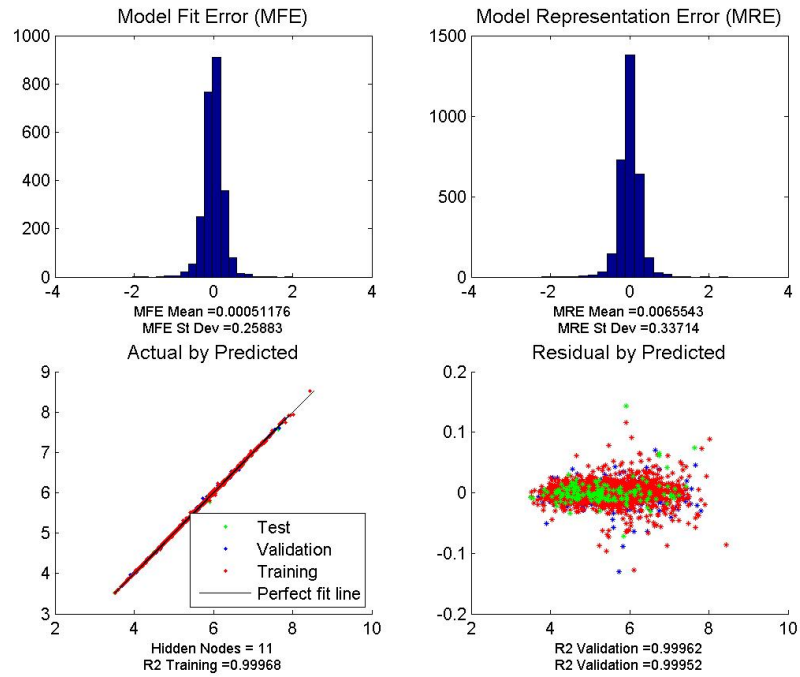


Figure 111: Core Defining Engine - Deterministic model fit statistics - HPC exit corrected flow at ADP.

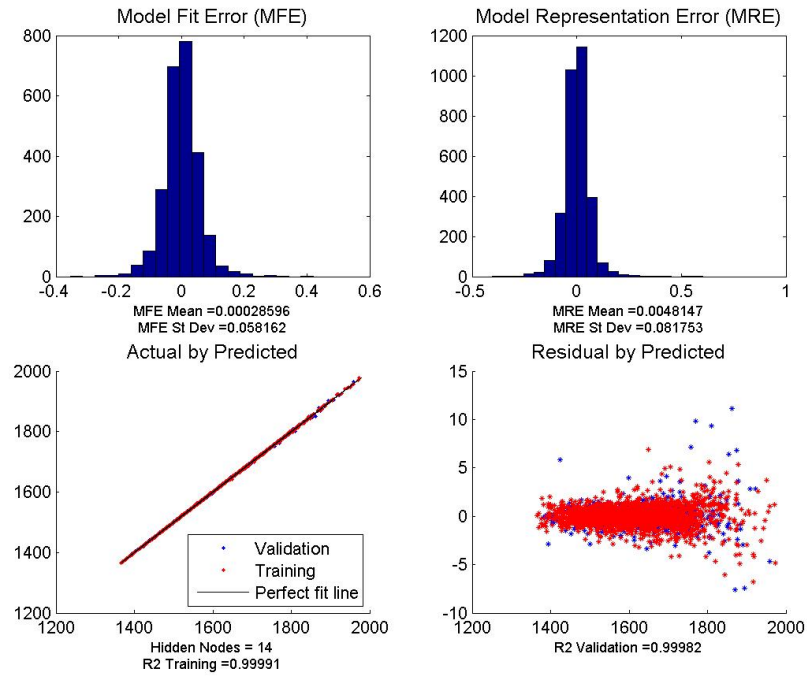


Figure 112: Core Defining Engine - Deterministic model fit statistics - HPC exit total temperature (T3) at TKO.

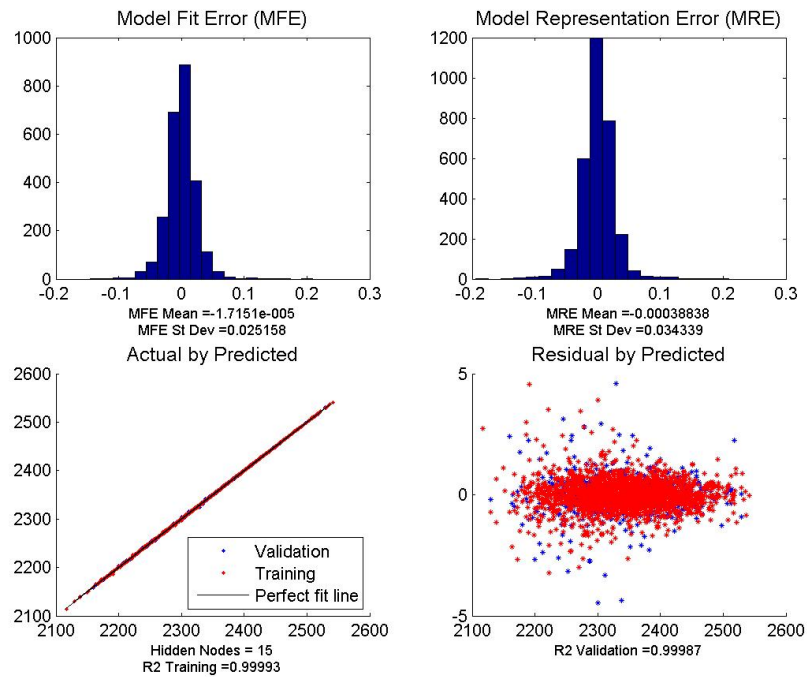


Figure 113: Core Defining Engine - Deterministic model fit statistics - LPT inlet total temperature (T45) at TKO.

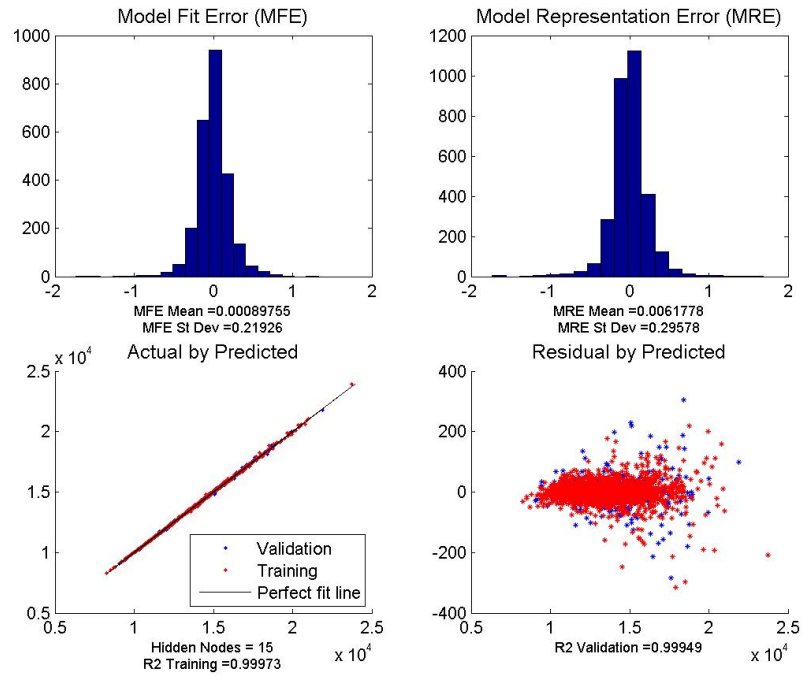


Figure 114: Core Defining Engine - Deterministic model fit statistics - HP shaft power at ADP.

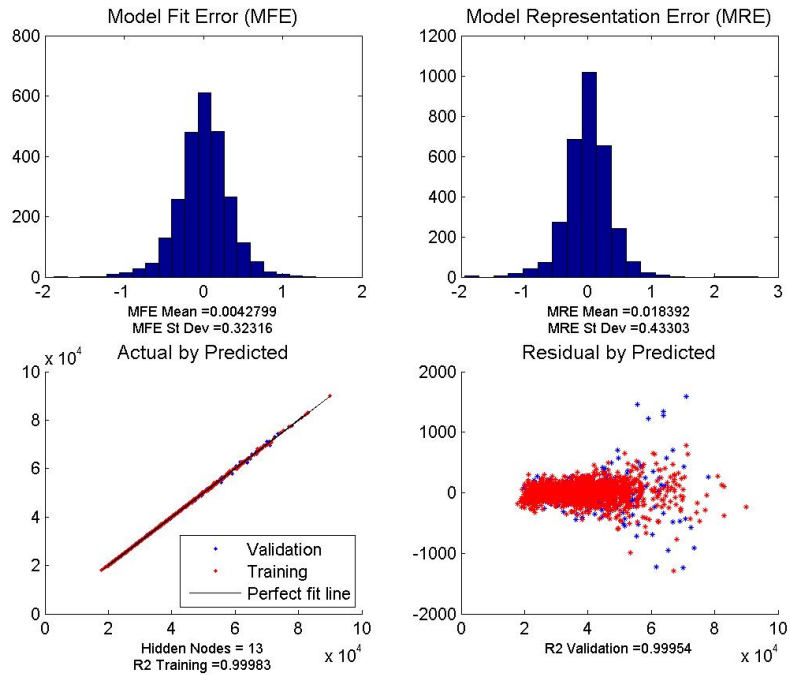


Figure 115: Core Defining Engine - Deterministic model fit statistics - HP shaft power at TKO.

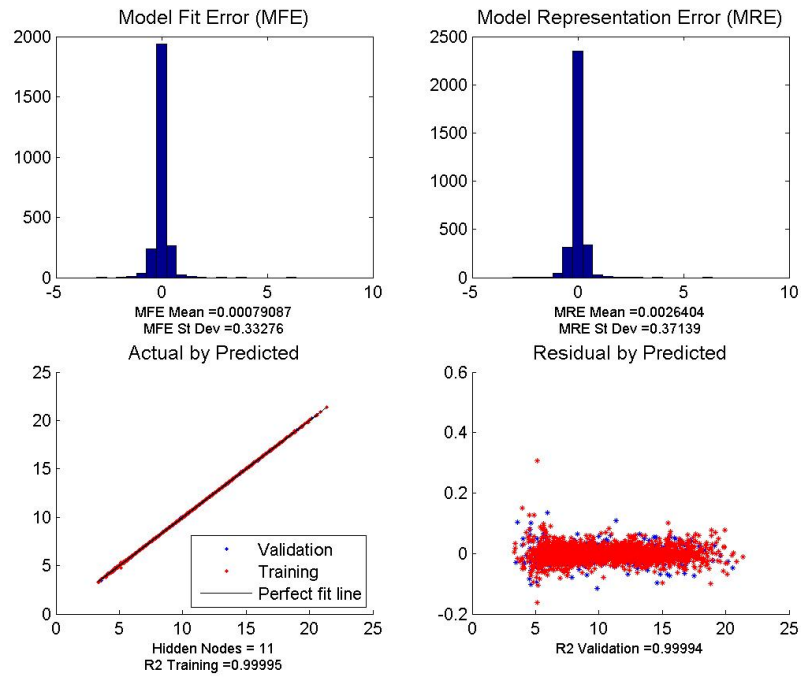


Figure 116: Core Defining Engine - Deterministic model fit statistics - Bypass ratio at ADP.

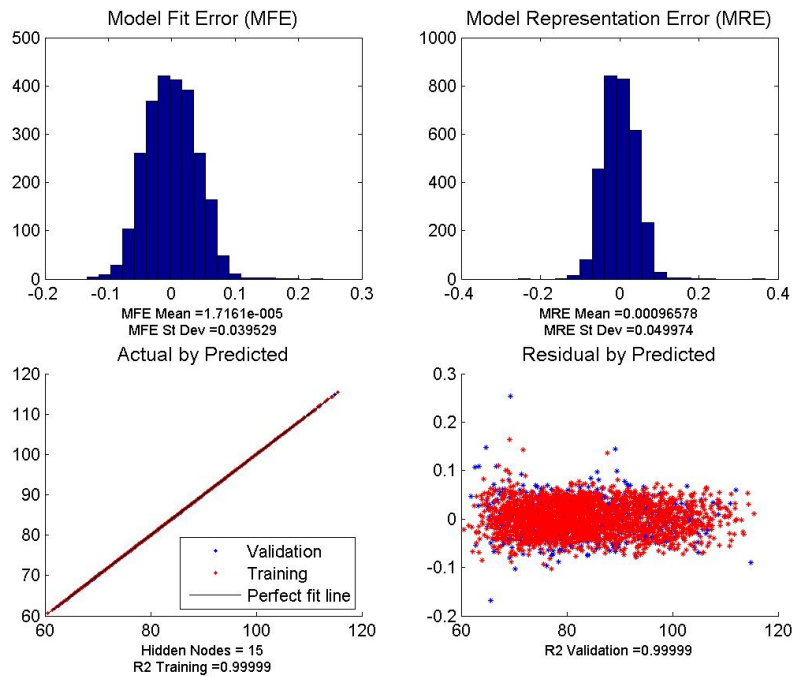


Figure 117: Core Defining Engine - Deterministic model fit statistics - Fan diameter.

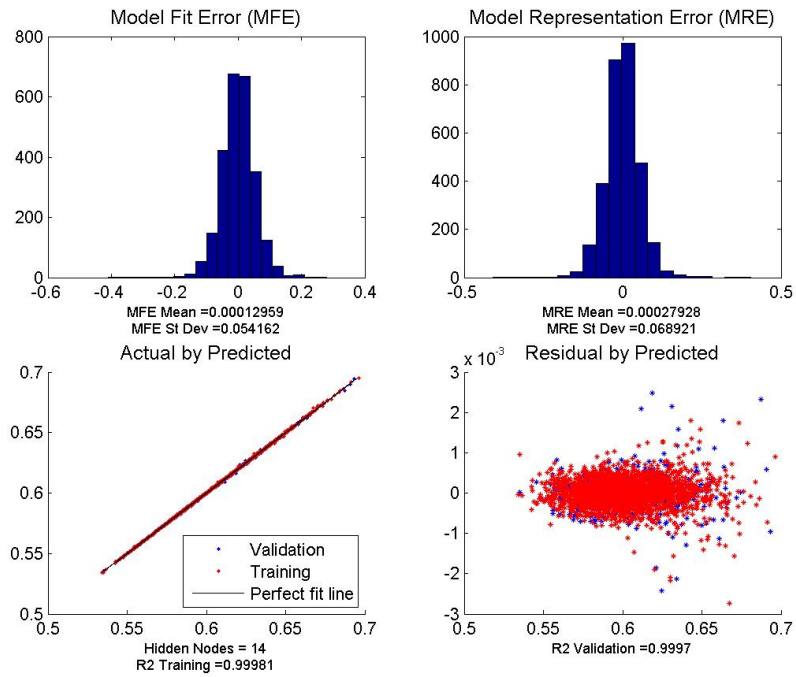


Figure 118: Core Defining Engine - Deterministic model fit statistics - TSFC at ADP.

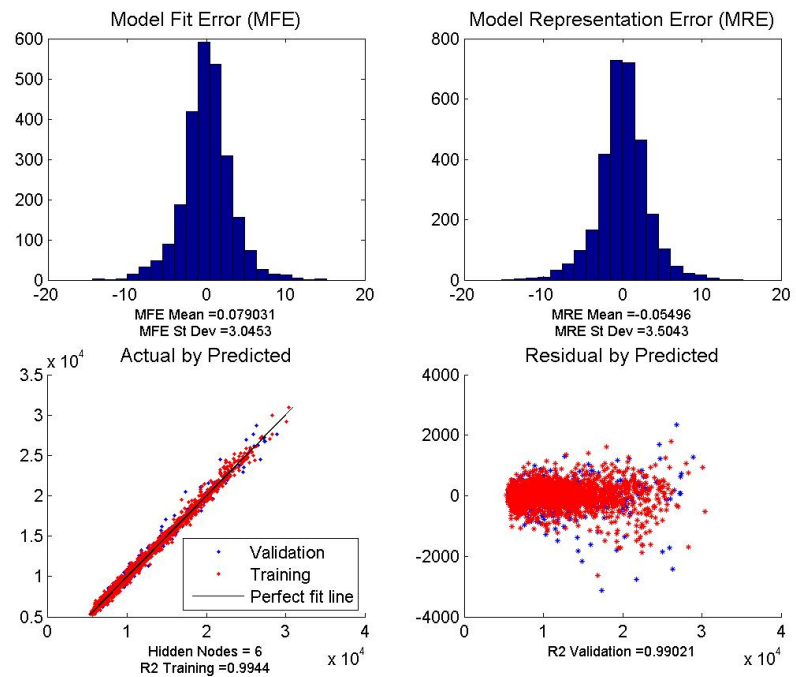


Figure 119: Core Defining Engine - Deterministic model fit statistics - Engine pod weight.

C.3 Common Core Variant Engine - Fit Statistics

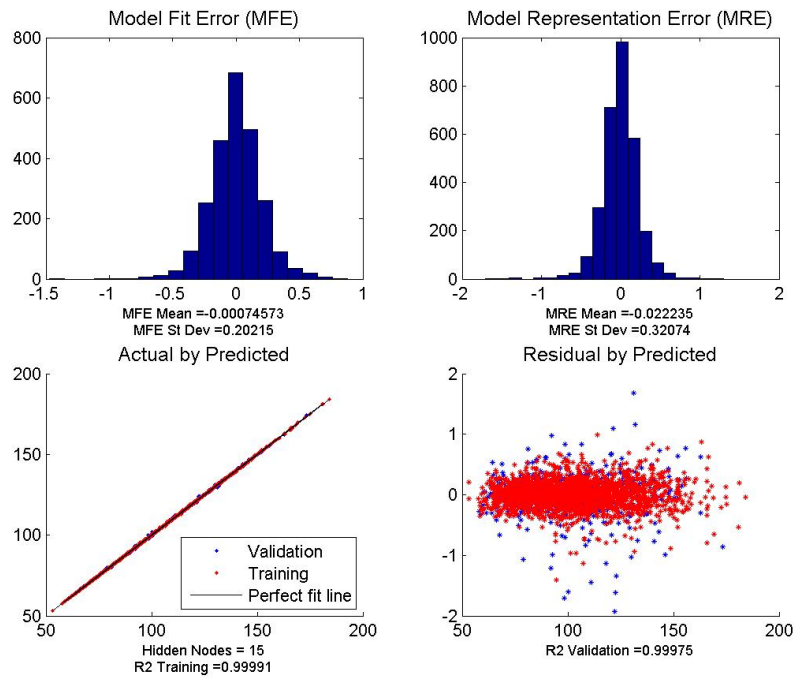


Figure 120: Common Core Variant Engine - Deterministic model fit statistics - LPC inlet corrected flow at ADP.

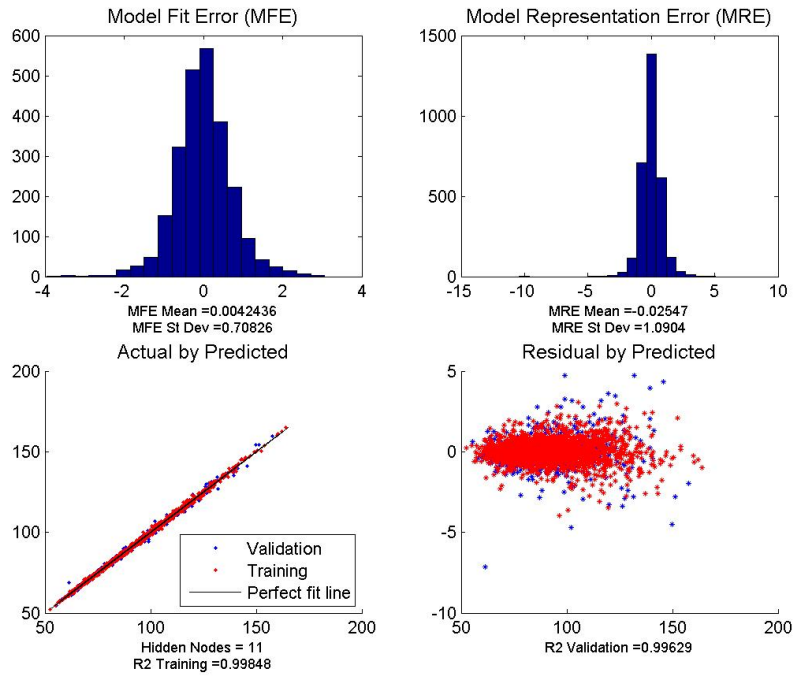


Figure 121: Common Core Variant Engine - Deterministic model fit statistics - LPC inlet corrected flow at TKO.

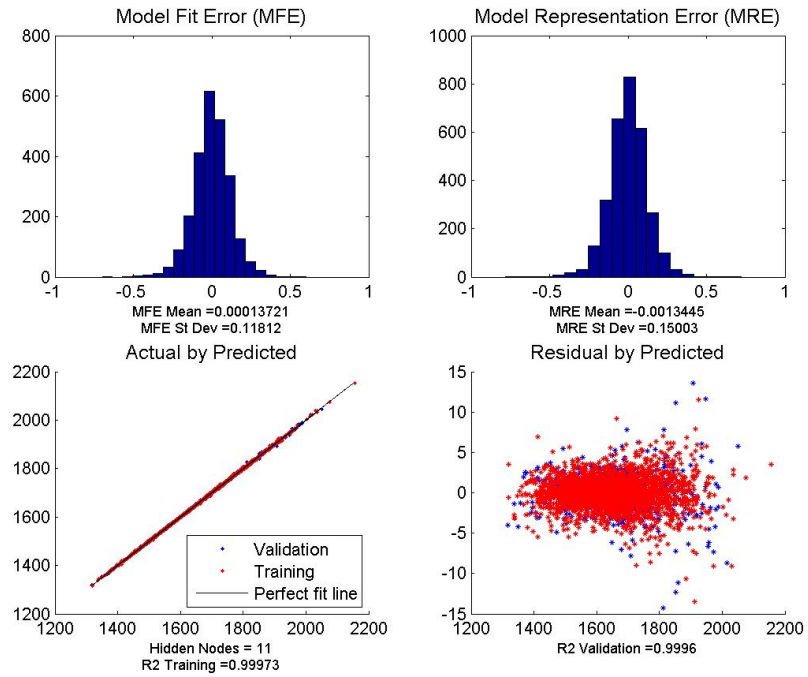


Figure 122: Common Core Variant Engine - Deterministic model fit statistics - HPC exit total temperature (T3) at TKO.

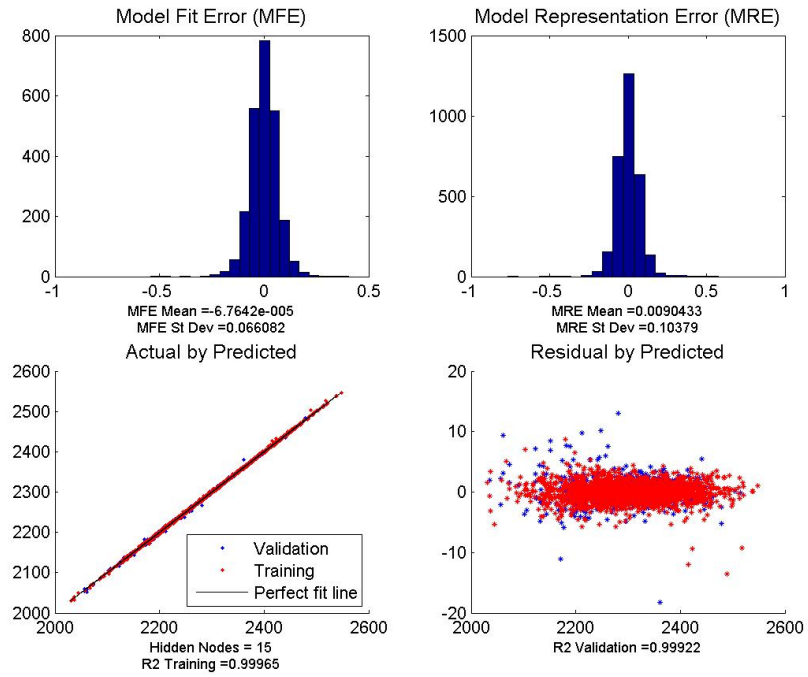


Figure 123: Common Core Variant Engine - Deterministic model fit statistics - LPT inlet total temperature (T45) at TKO.

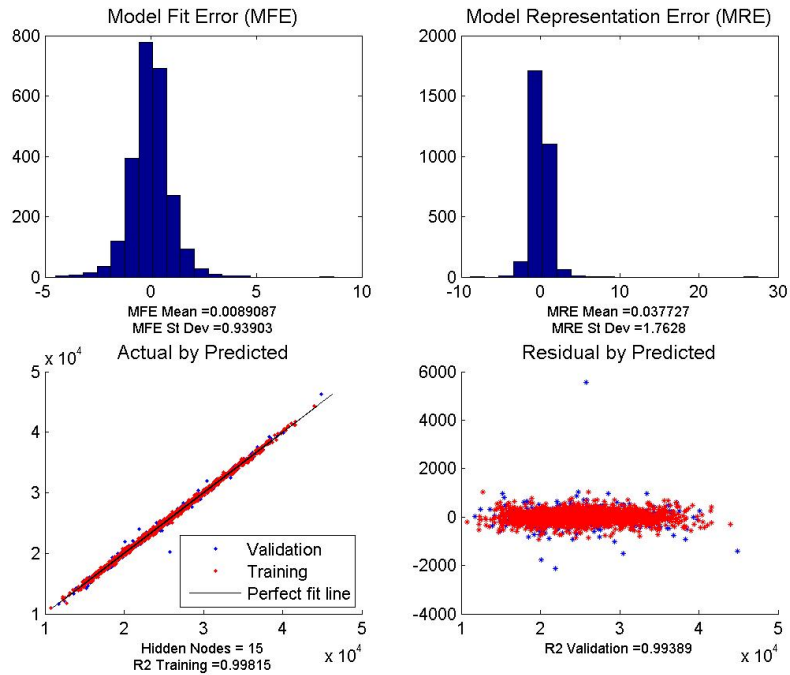


Figure 124: Common Core Variant Engine - Deterministic model fit statistics - Net thrust at TKO.

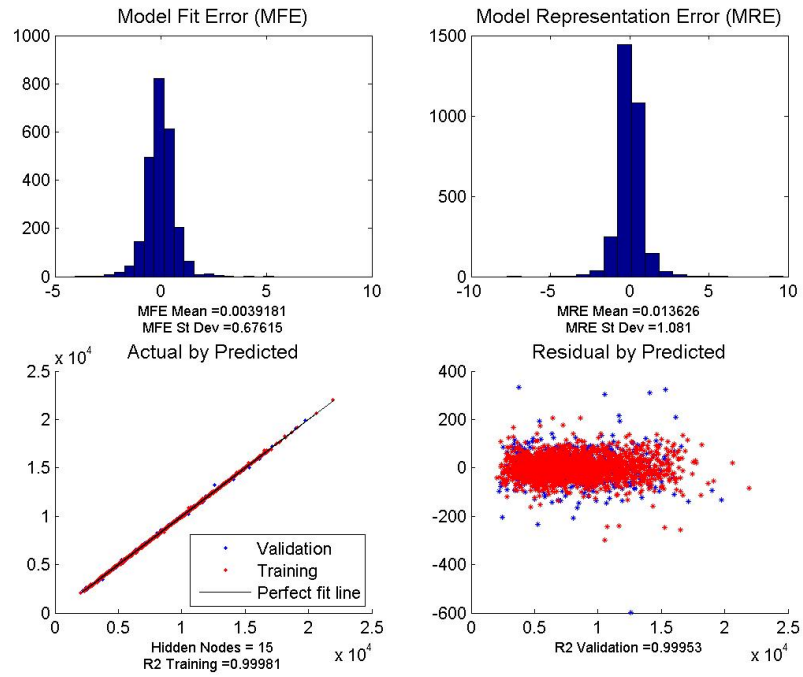


Figure 125: Common Core Variant Engine - Deterministic model fit statistics - Net thrust at TOC.

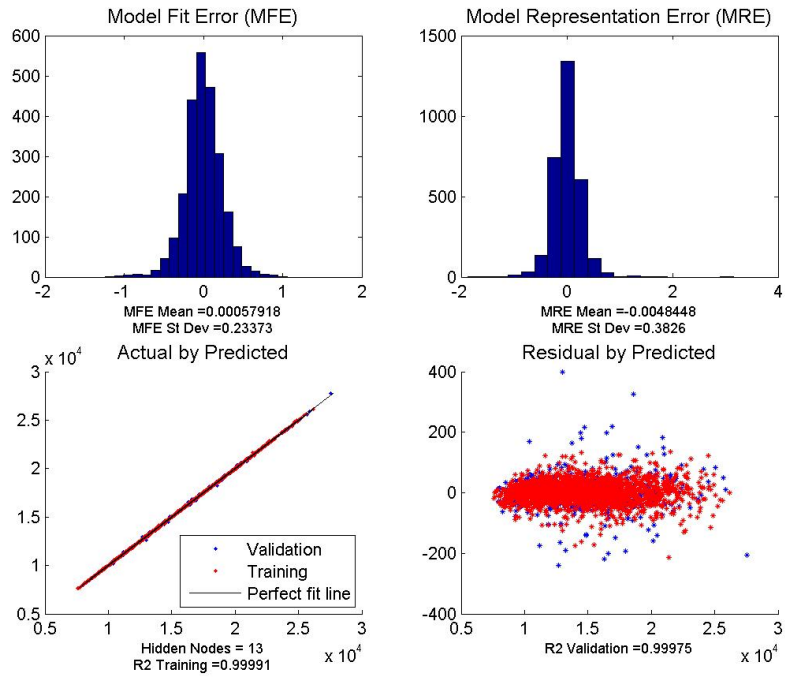


Figure 126: Common Core Variant Engine - Deterministic model fit statistics - HP shaft power at ADP.

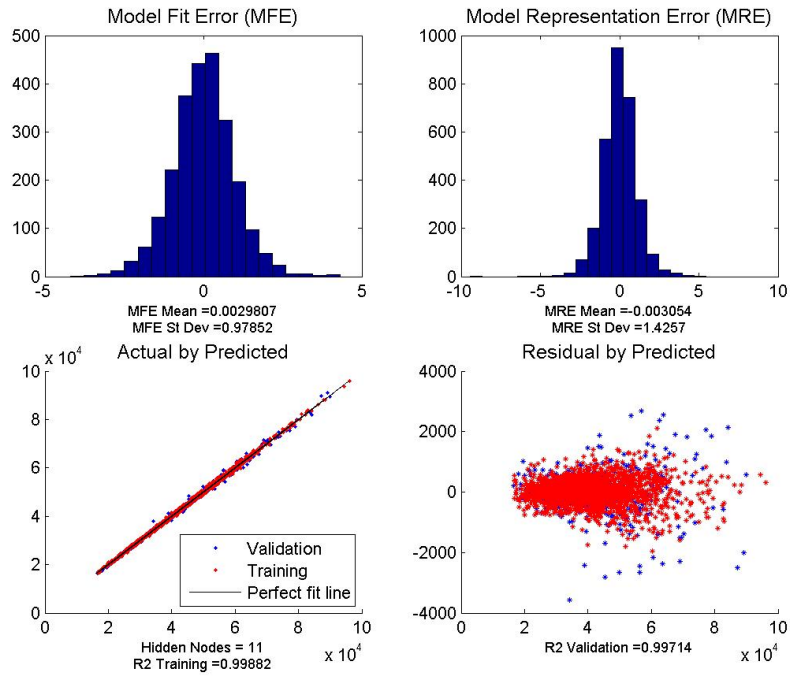


Figure 127: Common Core Variant Engine - Deterministic model fit statistics - HP shaft power at TKO.

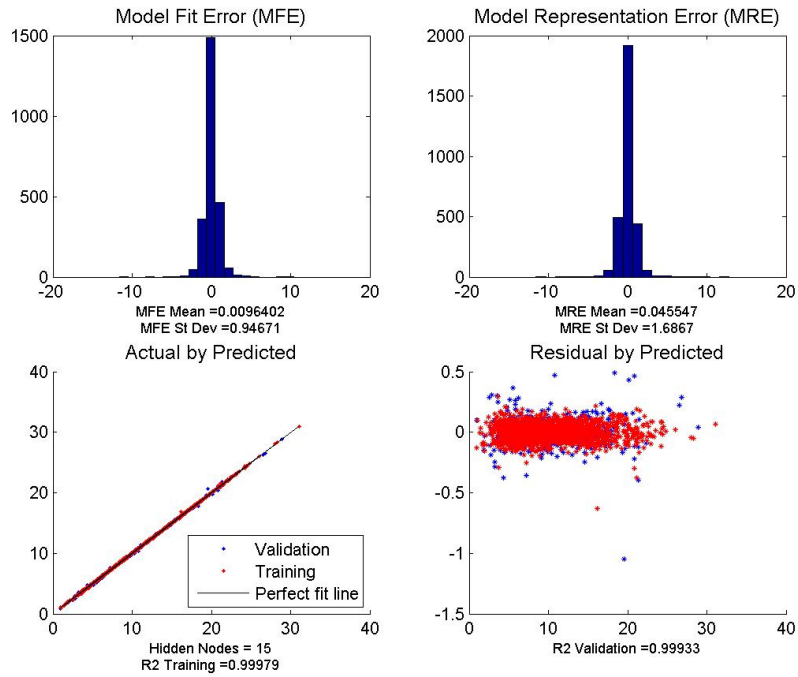


Figure 128: Common Core Variant Engine - Deterministic model fit statistics - Bypass ratio at ADP.

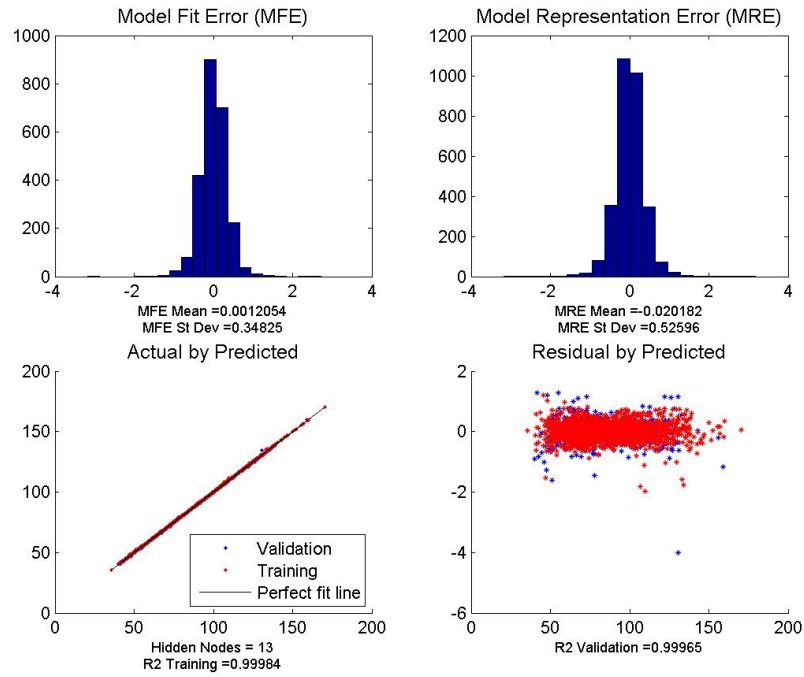


Figure 129: Common Core Variant Engine - Deterministic model fit statistics - Fan diameter.

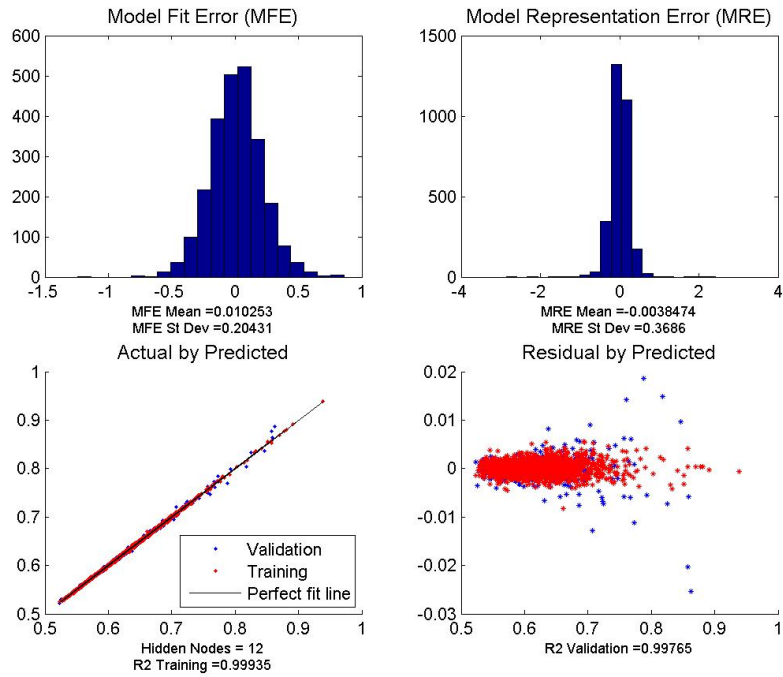


Figure 130: Common Core Variant Engine - Deterministic model fit statistics - TSFC at ADP.

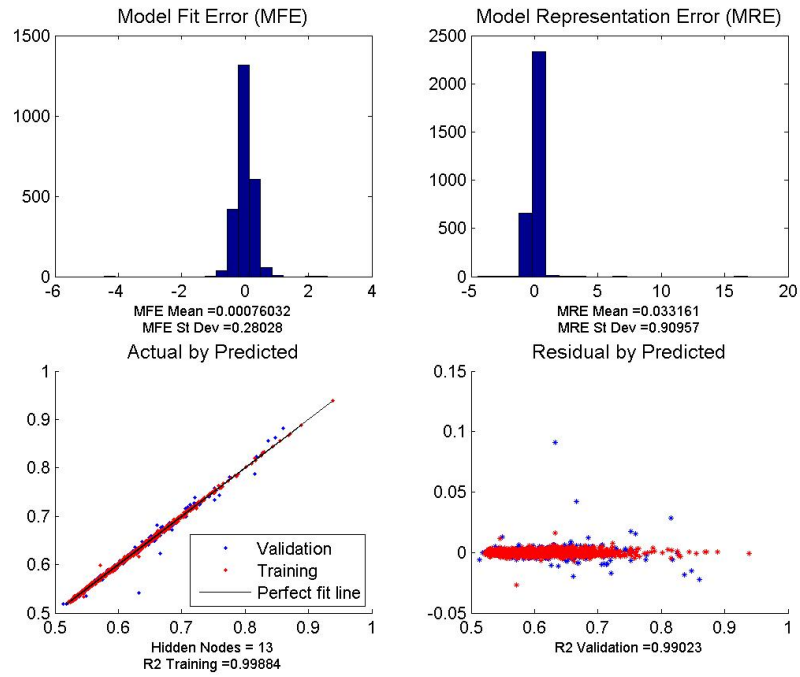


Figure 131: Common Core Variant Engine - Deterministic model fit statistics - Minimum TSFC at CRZ.

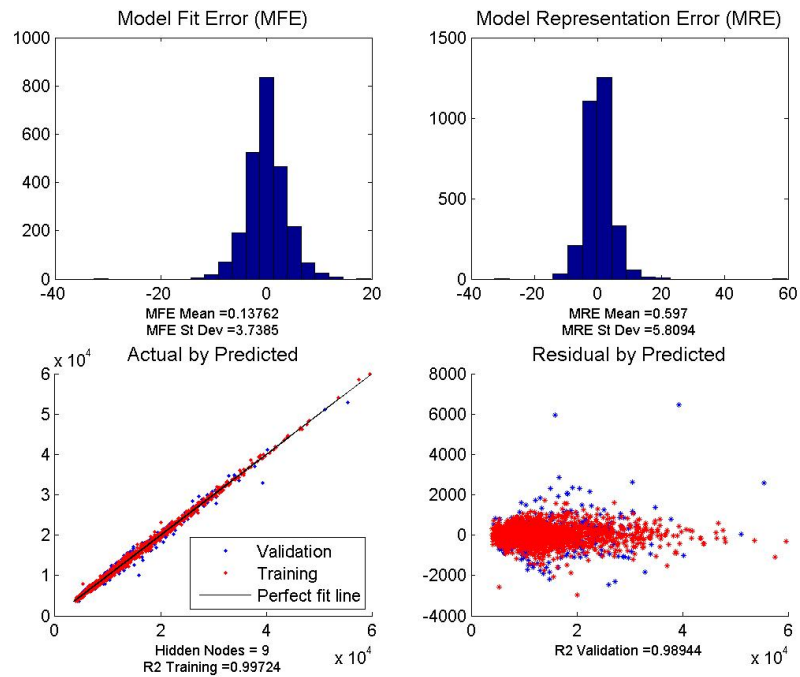


Figure 132: Common Core Variant Engine - Deterministic model fit statistics - Engine pod weight.

APPENDIX D

PROBABILISTIC SURROGATE MODEL FIT STATISTICS

D.1 Overview

Probabilistic surrogate models were generated to estimate an inputted confidence interval's estimated performance for an inputted set of unique control variable settings. Again, these surrogate models are in the form of Artificial Neural Network models. The models are functions of the control variables, made up of design parameters and uncertainty distribution shape parameters. For each desired model, the ANN architecture was explored in order to find the number of hidden nodes that offered the best model fit quality within the imposed training time limits. The data generated by the Monte Carlo analysis script shown in Appendix B was used as training data for these probabilistic surrogate models.

Just as was done when evaluating the quality of deterministic surrogate models, several measures are used to evaluate the quality of the probabilistic surrogate models. Refer to the requirements of surrogate fit quality statistics listed in Section C.1. A model whose characteristics meet these requirements are considered to adequately represent the model and corresponding space explored during the generation of data used to train and evaluate the model under consideration.

The following two sections contain figures visualizing the fit quality and statistics of each probabilistic surrogate model of interest. Models were trained for both the core defining engine characteristics as well as the common core engine variant design. The core defining engine surrogates are functions of the control variables defining the design engine. However, the common core engine surrogates are functions of the control variables corresponding to both the design and variant engines. This is due to the common core definition originating from the design engine and being applied to the variant engine. Implicit in these probabilistic surrogates are the effects of the range of uncertainty distributions considered within the range of shape parameter settings explored.

D.2 Core Defining Engine - Fit Statistics

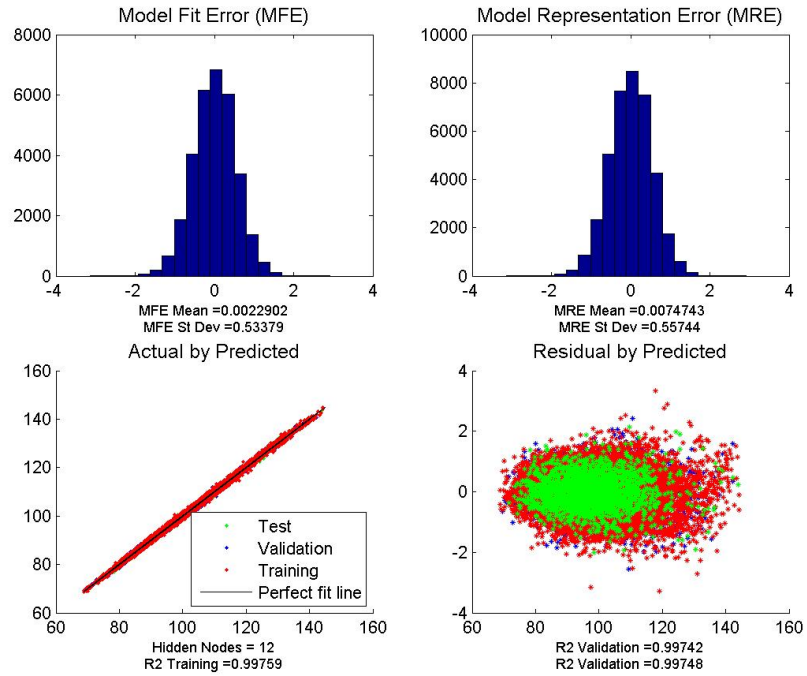


Figure 133: Core Defining Engine - Probabilistic model fit statistics - LPC inlet corrected flow at ADP.

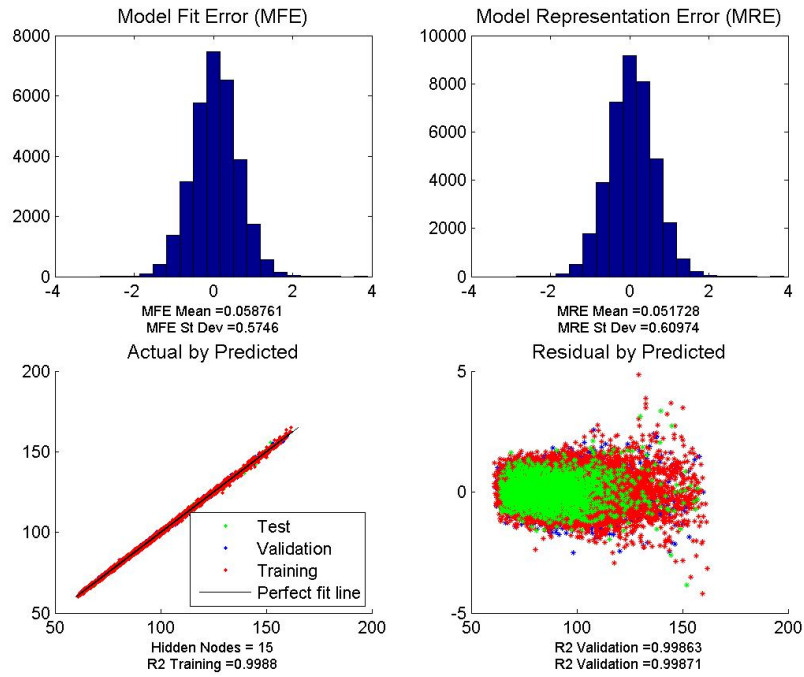


Figure 134: Core Defining Engine - Probabilistic model fit statistics - LPC inlet corrected flow at TKO.

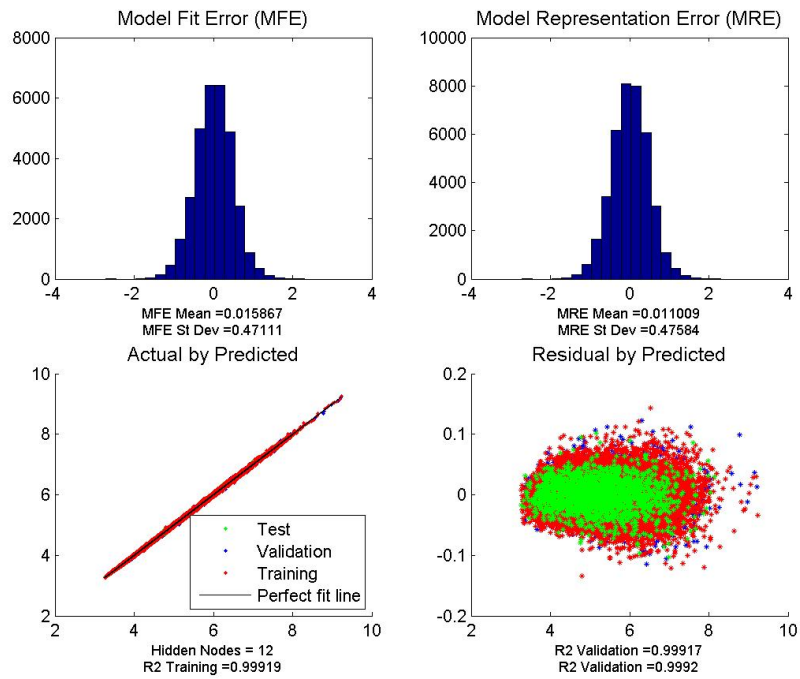


Figure 135: Core Defining Engine - Probabilistic model fit statistics - HPC exit corrected flow at ADP.

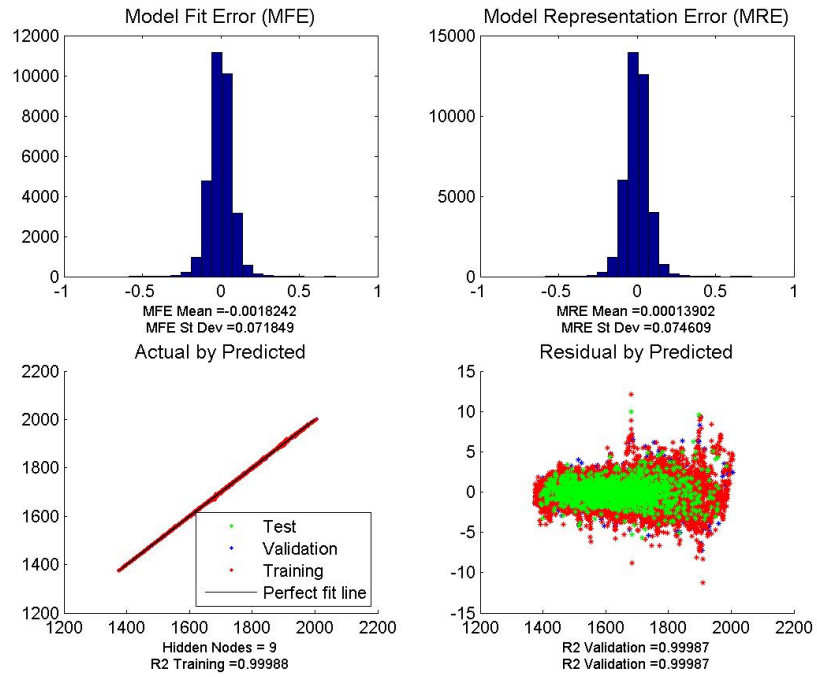


Figure 136: Core Defining Engine - Probabilistic model fit statistics - HPC exit total temperature (T3) at TKO.

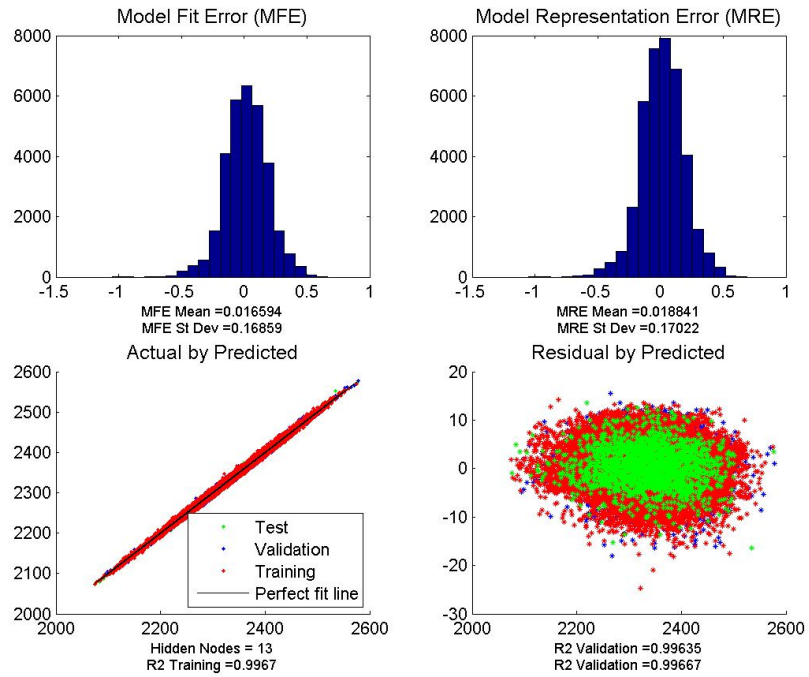


Figure 137: Core Defining Engine - Probabilistic model fit statistics - LPT inlet total temperature (T45) at TKO.

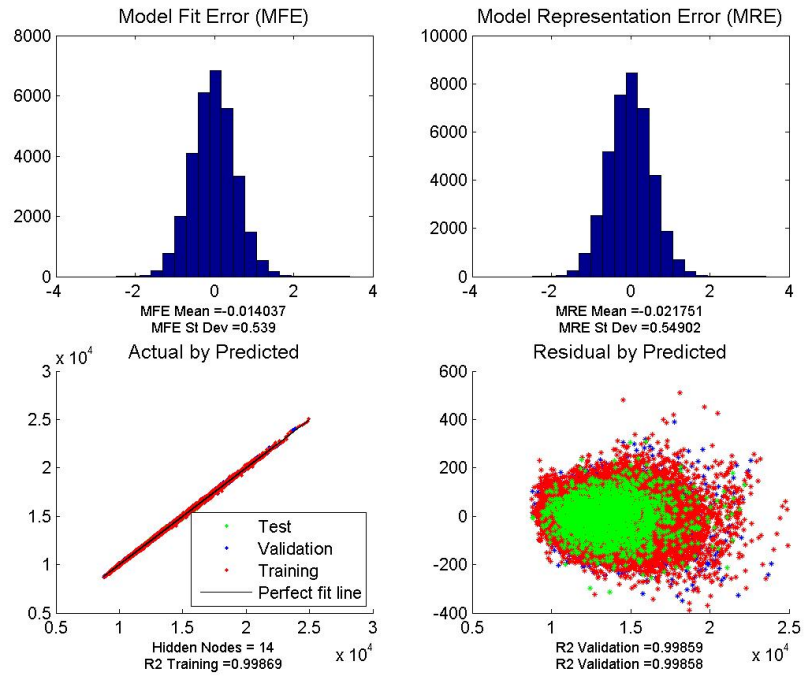


Figure 138: Core Defining Engine - Probabilistic model fit statistics - HP shaft power at ADP.

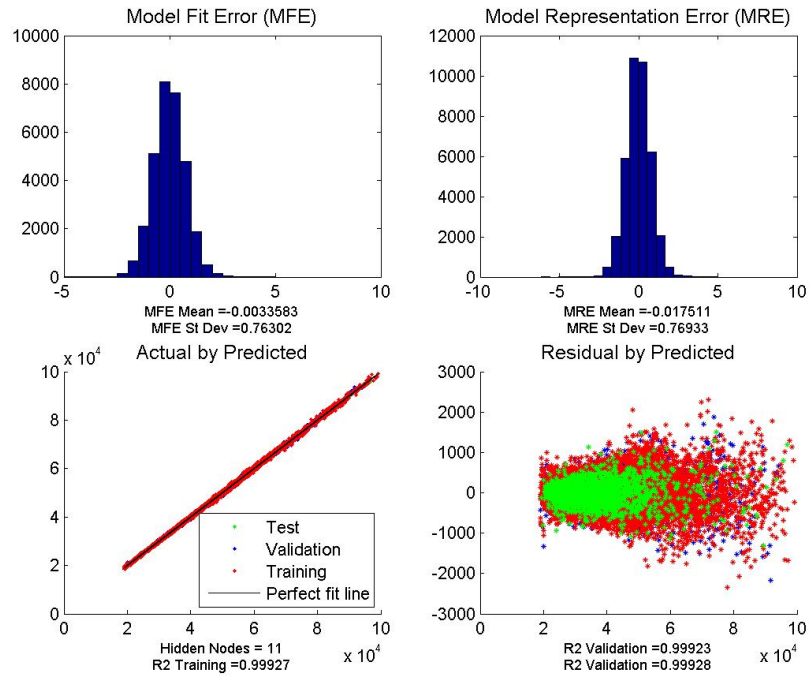


Figure 139: Core Defining Engine - Probabilistic model fit statistics - HP shaft power at TKO.

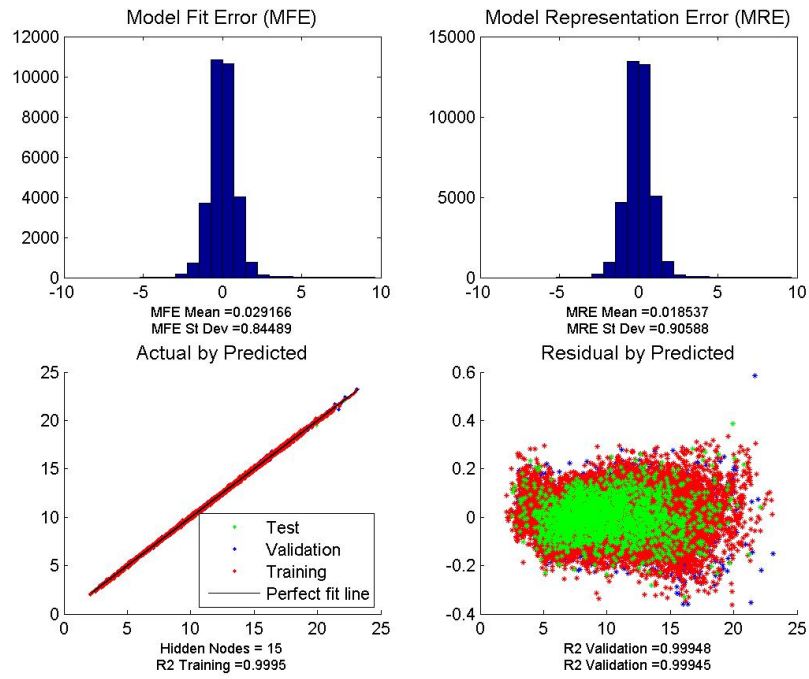


Figure 140: Core Defining Engine - Probabilistic model fit statistics - Bypass ratio at ADP.

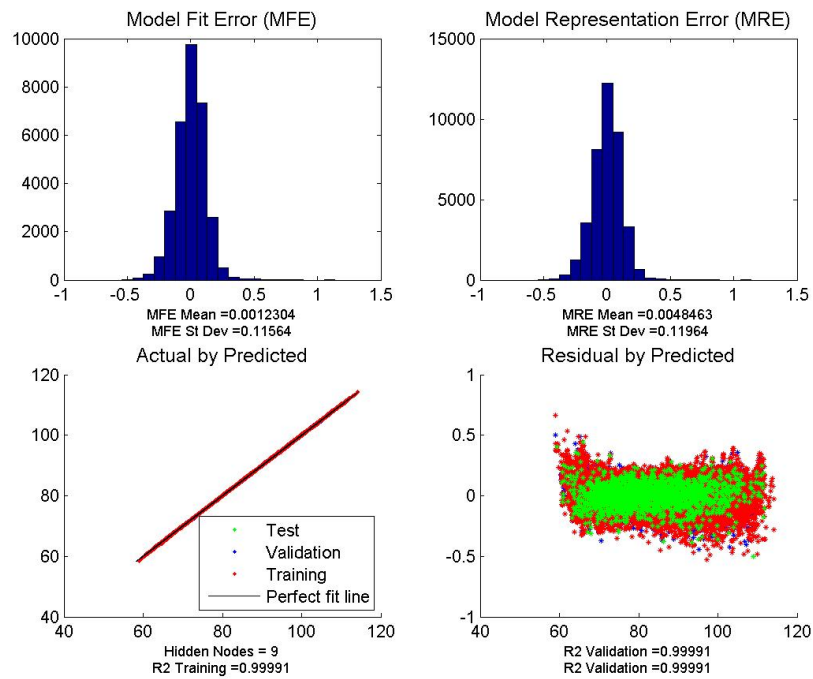


Figure 141: Core Defining Engine - Probabilistic model fit statistics - Fan diameter.

D.3 Common Core Variant Engine - Fit Statistics

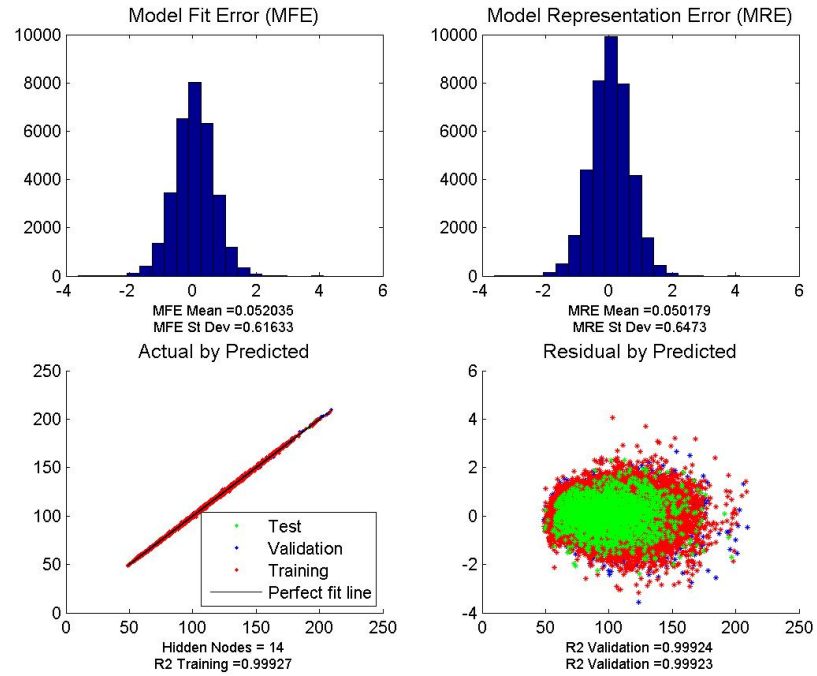


Figure 142: Common Core Variant Engine - Probabilistic model fit statistics - LPC inlet corrected flow at ADP.

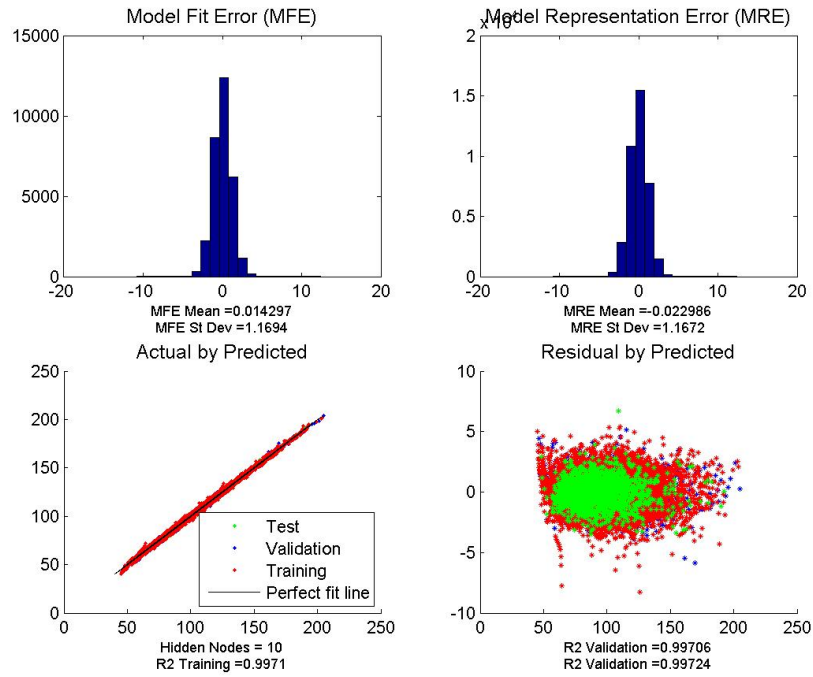


Figure 143: Common Core Variant Engine - Probabilistic model fit statistics - LPC inlet corrected flow at TKO.

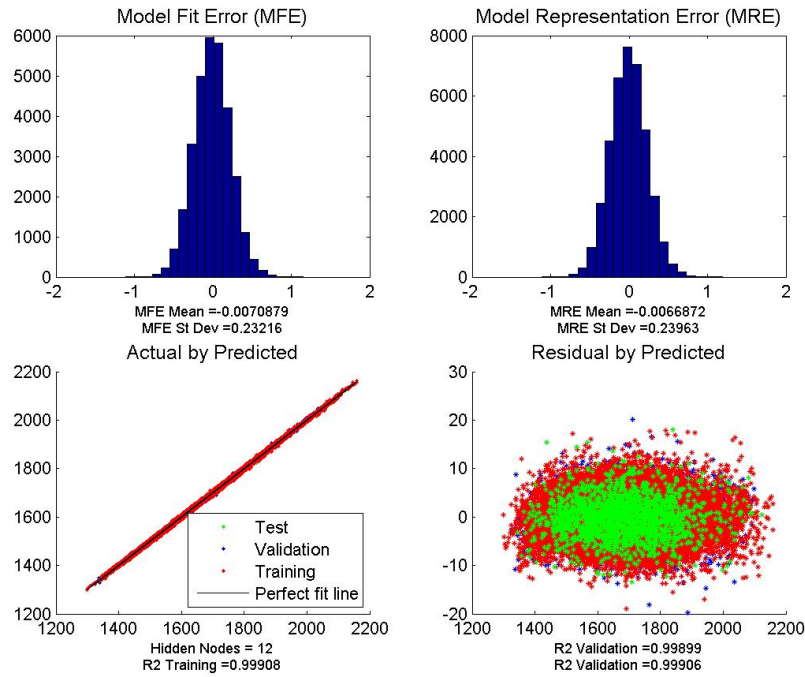


Figure 144: Common Core Variant Engine - Probabilistic model fit statistics - HPC exit total temperature (T_3) at TKO.

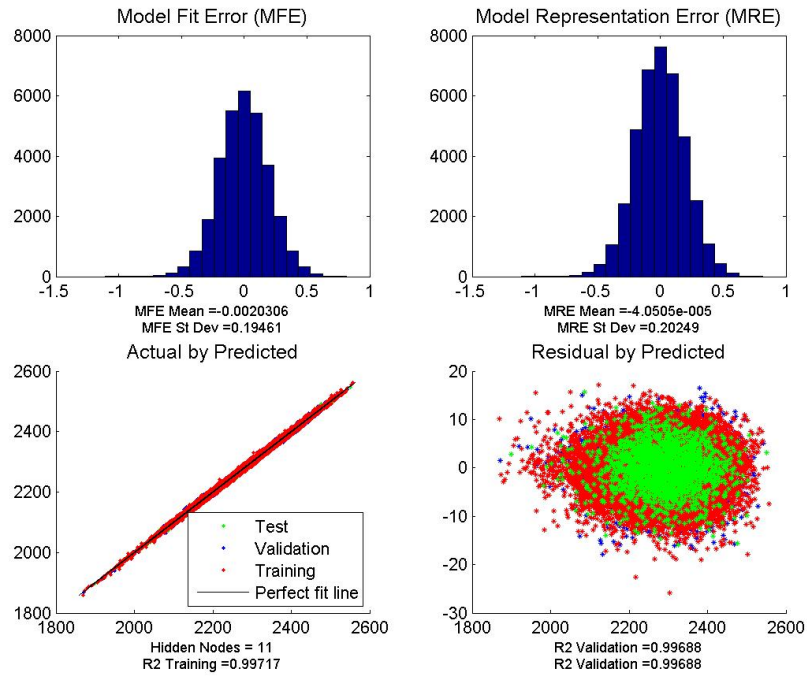


Figure 145: Common Core Variant Engine - Probabilistic model fit statistics - LPT inlet total temperature (T45) at TKO.

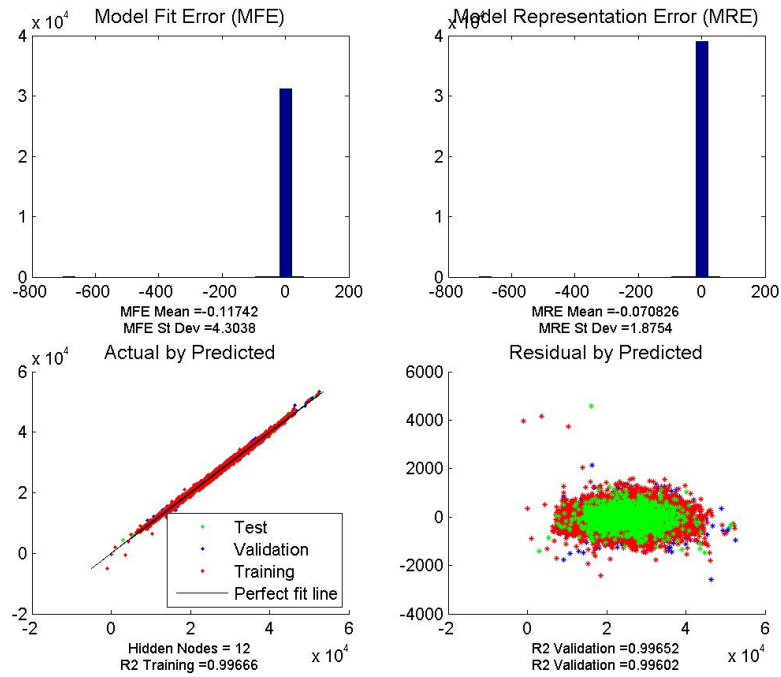


Figure 146: Common Core Variant Engine - Probabilistic model fit statistics - Net thrust at TKO.

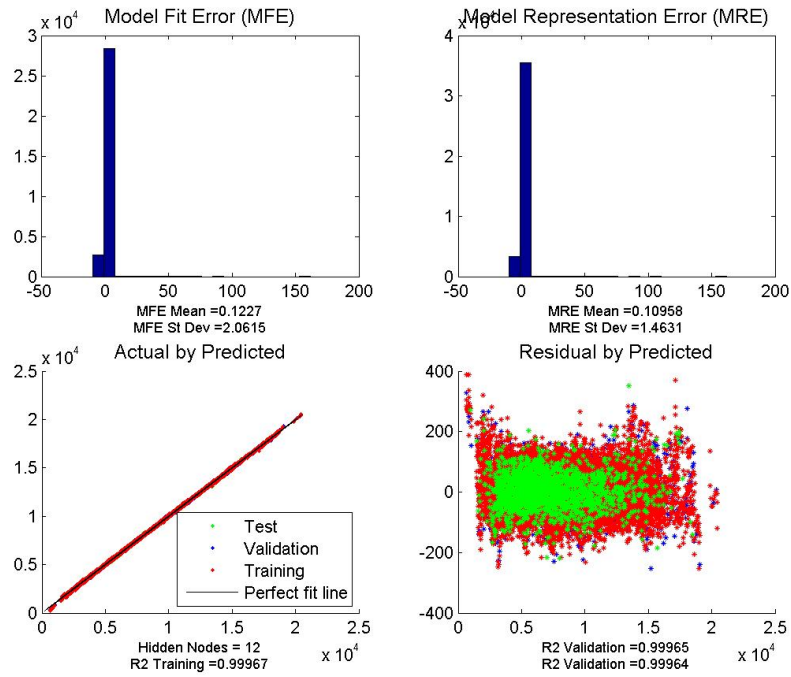


Figure 147: Common Core Variant Engine - Probabilistic model fit statistics - Net thrust at TOC.

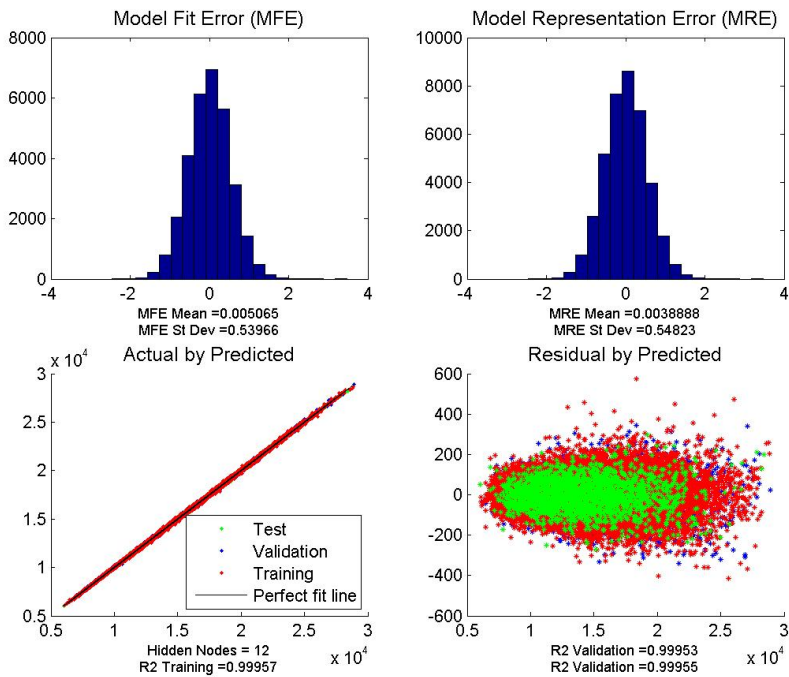


Figure 148: Common Core Variant Engine - Probabilistic model fit statistics - HP shaft power at ADP.

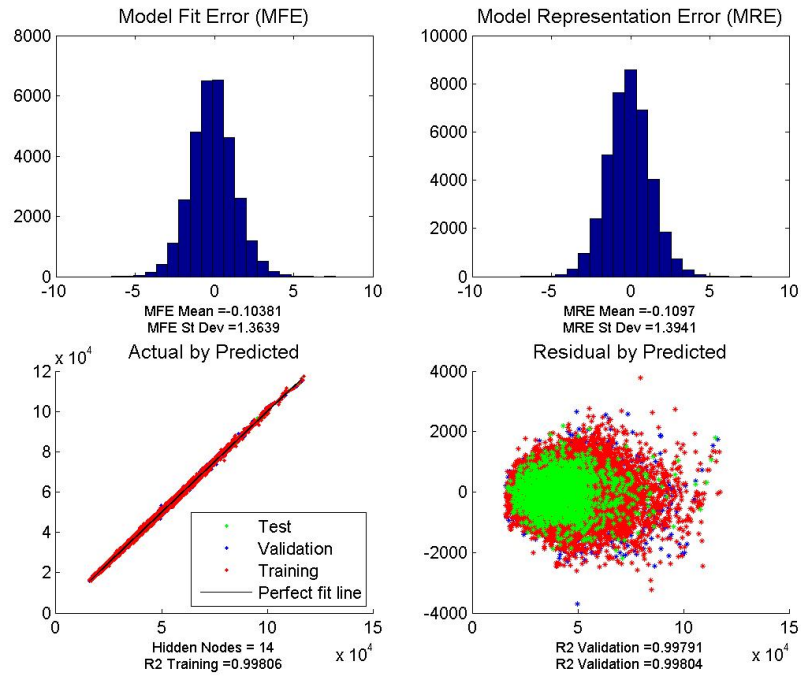


Figure 149: Common Core Variant Engine - Probabilistic model fit statistics - HP shaft power at TKO.

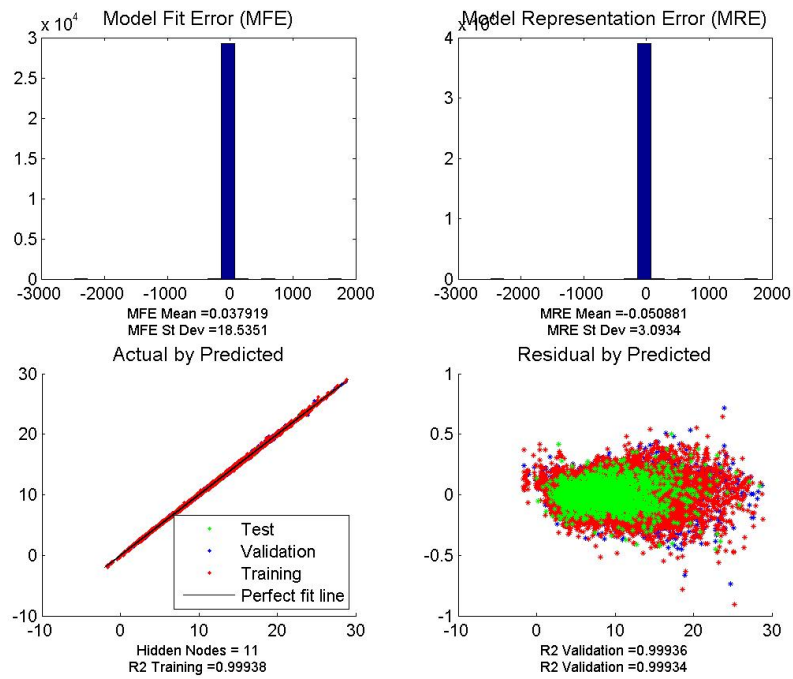


Figure 150: Common Core Variant Engine - Probabilistic model fit statistics - Bypass ratio at ADP.

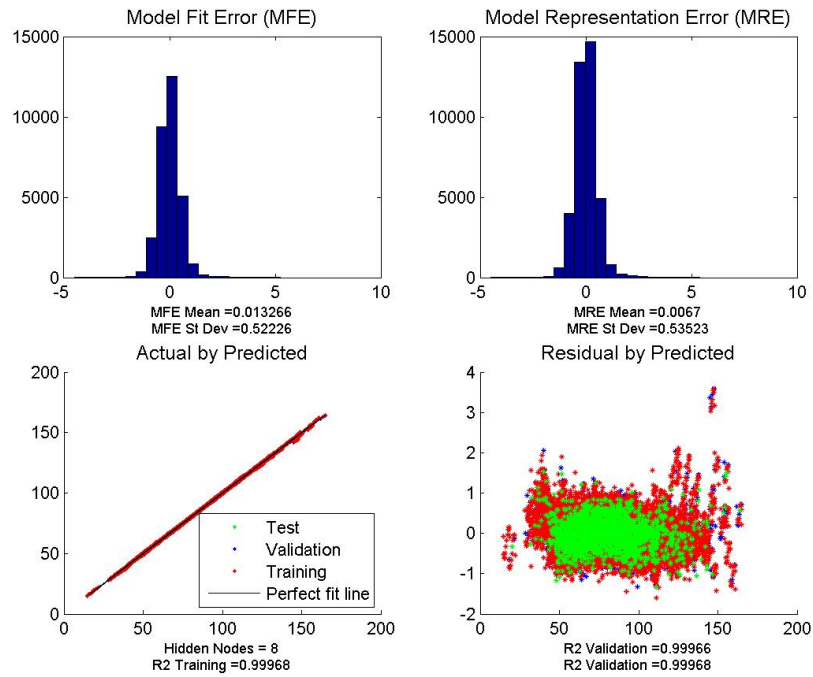


Figure 151: Common Core Variant Engine - Probabilistic model fit statistics - Fan diameter.

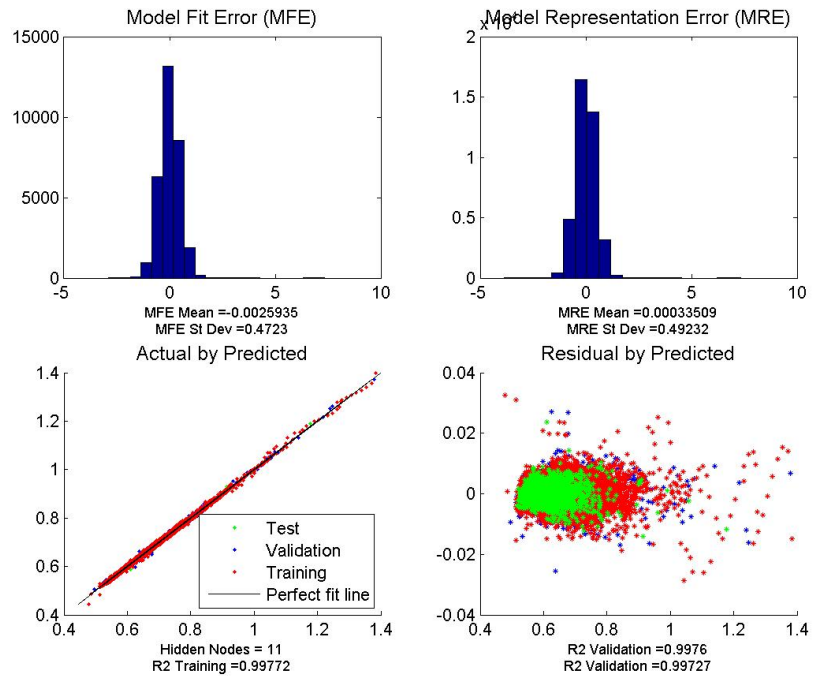


Figure 152: Common Core Variant Engine - Probabilistic model fit statistics - Minimum TSFC at CRZ.

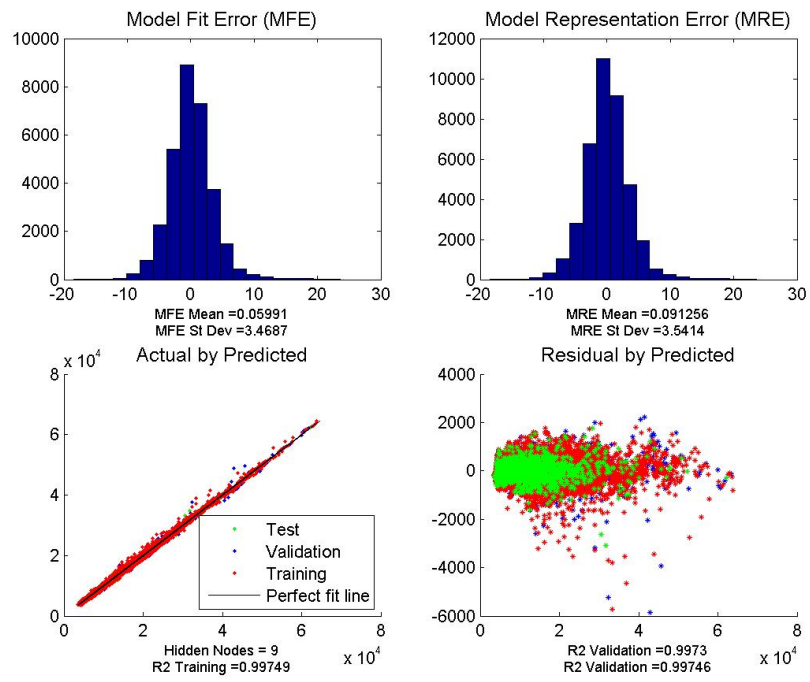


Figure 153: Common Core Variant Engine - Probabilistic model fit statistics - Engine pod weight.

APPENDIX E

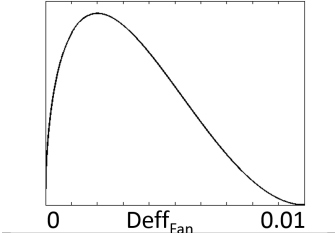
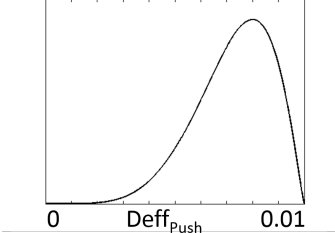
EXPERIMENT 1C: ROBUST CYCLE AND TECHNOLOGY DESIGN

To highlight the added features of the ERDS process compared to the RDS process before modification, this experiment allows the designer to select designs for different corresponding technology scenarios, all with the use of a single set of probabilistic surrogate models. With the ERDS method, the designer is no longer limited to a single technology scenario when performing a robust design study. The designer can now input the desired distribution shape parameters as control variables, and design selections can be made for any uncertainty scenario with shape parameter values within the ranges considered during surrogate training.

E.1 Technology Scenarios

The efficiency distributions considered in Table 35 are redisplayed below in Table 73, showing the uncertainty distributions considered for the various component efficiency improvements considered in this experiment. The most likely efficiency improvement for each component was assumed to be a quarter of a point in efficiency. The present experiment considers effects of various technology advancements, each advancement's impact on the best design selection, and their corresponding achievable performance. Five discrete technology scenarios are considered. Each scenario allows a different component of the engine to have its original efficiency improvement distribution replaced by a more advanced improvement distribution. The *push* distribution has an improved mean value with a reduction in variability around the improved mean. The *push* distribution is shown in Figure 73 next to the baseline distribution used for each of the five turbomachinery baseline efficiency improvement distributions. Replacing the baseline distribution with the pushed distribution increases the most likely efficiency improvement from 0.25 points to 0.75 percent in efficiency and also has a higher probability in achieving that most likely improvement level.

Table 73: Experiment 1c - Baseline and Pushed Efficiency Improvement Distributions

Impact	Min, Max, Alpha, Beta	PDF
$\Delta\eta_{\text{Base}}$	0.00, 0.01, 1.5, 3.0	
$\Delta\eta_{\text{Push}}$	0.00, 0.01, 6.0, 2.25	

The five technology pushes represent maturation in each component’s most likely efficiency and an increase in the likelihood of achieving the improved efficiency increase. Note that the absolute range of improvement is consistent between technology scenarios. Only the probability of achieving values throughout the fixed range is modified.

E.2 Engine Design Space Exploration for Various Technology Scenarios

The probabilistic surrogate models referred to as *Surrogate Set C* are again employed for this experiment. For the present experiment, six unique technology scenarios are considered. The first scenario, referred to as the *base* scenario, assumes all the original efficiency improvement distributions considered in the previous experiment. The remaining five scenarios each apply the *pushed* efficiency improvement distribution to one of the five turbomachinery components. The efficiency improvement uncertainty distributions used for each of the six technology scenarios are listed in Table 74.

For each scenario, the design cycle space is explored, considering 20,000 unique candidate cycles. The efficiency improvement distribution parameters contained in **A** and **B** are

Table 74: Experiment 1c - Component efficiency improvement distribution shape parameter settings for each of the technology scenarios considered.

Distribution	Parameter	Technology Scenario					
		Base	Fan Push	LPC Push	HPC Push	HPT Push	LPT Push
$\Delta\eta_{Fan}$	$\alpha_{\Delta\eta_{Fan}}$	1.5	6.0	1.5	1.5	1.5	1.5
	$\beta_{\Delta\eta_{Fan}}$	3.0	2.25	3.0	3.0	3.0	3.0
$\Delta\eta_{LPC}$	$\alpha_{\Delta\eta_{LPC}}$	1.5	1.5	6.0	1.5	1.5	1.5
	$\beta_{\Delta\eta_{LPC}}$	3.0	3.0	2.25	3.0	3.0	3.0
$\Delta\eta_{HPC}$	$\alpha_{\Delta\eta_{HPC}}$	1.5	1.5	1.5	6.0	1.5	1.5
	$\beta_{\Delta\eta_{HPC}}$	3.0	3.0	3.0	2.25	3.0	3.0
$\Delta\eta_{HPT}$	$\alpha_{\Delta\eta_{HPT}}$	1.5	1.5	1.5	1.5	6.0	1.5
	$\beta_{\Delta\eta_{HPT}}$	3.0	3.0	3.0	3.0	2.25	3.0
$\Delta\eta_{LPT}$	$\alpha_{\Delta\eta_{LPT}}$	1.5	1.5	1.5	1.5	1.5	6.0
	$\beta_{\Delta\eta_{LPT}}$	3.0	3.0	3.0	3.0	3.0	2.25

fixed to each scenario's settings, and the five cycle design parameters contained in \mathbf{X} are varied throughout the ranges considered. The probabilistic surrogate models previously generated are used to predict the likely performance due to both the cycle settings and the distribution shape parameter settings for each of the candidate cycles. The designer is then able to select a candidate cycle that achieves maximum performance among the feasible candidates for each of the six scenarios. In the end, the designer is then able to select both the technology scenario of his choice and the corresponding best candidate cycle for that particular technology scenario.

The design cycle explorations that are performed for each technology scenario aim to answer two questions:

- *Which engine component should be matured in order to get the most performance benefit?*
- *How should the design cycle be selected in order to achieve the best performance for each technology scenario considered?*

Figure 154 contains the parallel coordinate plot for the base scenario, where all the engine components' efficiency improvements have the assumed distributions as in the previous

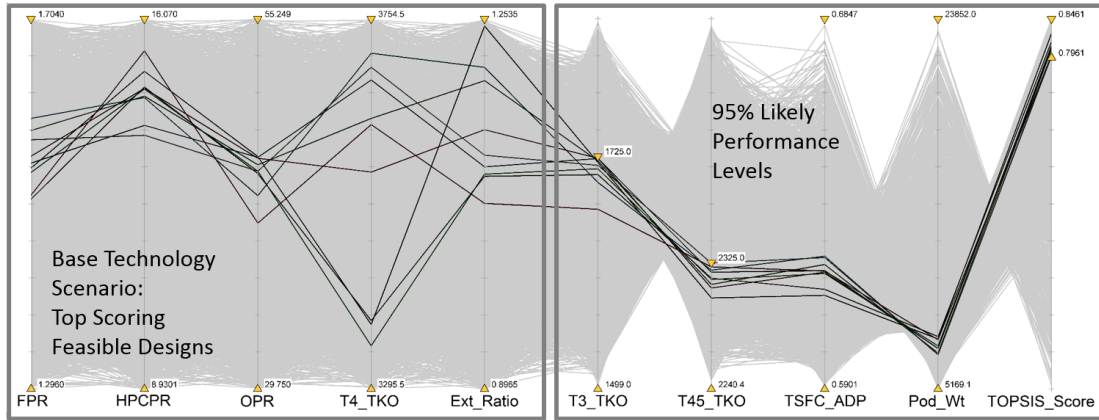


Figure 154: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, base efficiency distributions.

experiment. Similar plots are constructed for each scenario, showing the candidate cycles that are feasible and score highest in terms of specific fuel consumption and engine weight. Comparing each to the base plot identifies how the best cycle settings shift in response to maturation of the different turbomachinery components. Each plot shows the impact the associated efficiency improvements have on the number of feasible candidate designs, the cycle preference of each technology scenario, and the corresponding performance improvement possible from each scenario.

Figures 155 through 159 display the top scoring feasible cycle designs for each of the technology advancement scenarios, each corresponding to the *push* of one of the turbomachinery components' design efficiency level.

Table 75 displays the top scoring feasible candidate design samples for each technology push scenario considered, along with their 95% likely performance estimates. Since the selections were made from a populations of design candidate samples considered, it is likely for each scenario that performance would be able to be slightly improved beyond the values of these selections if optimization of the feasible design space were performed. However, these candidates give the decision maker good indications of which component(s) offer the most performance improvement over the baseline scenario. Considering the top design performance of each technology scenario, the *LPT Push* offered the most improvement

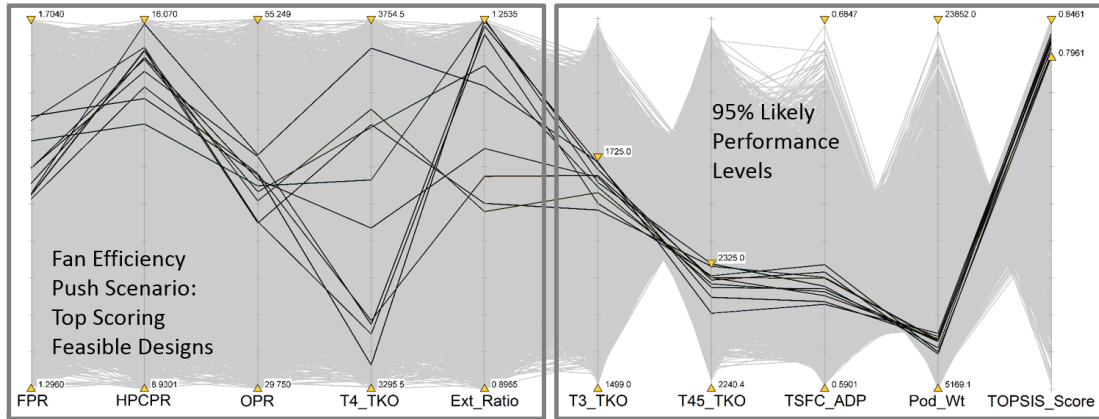


Figure 155: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, fan efficiency distribution push.

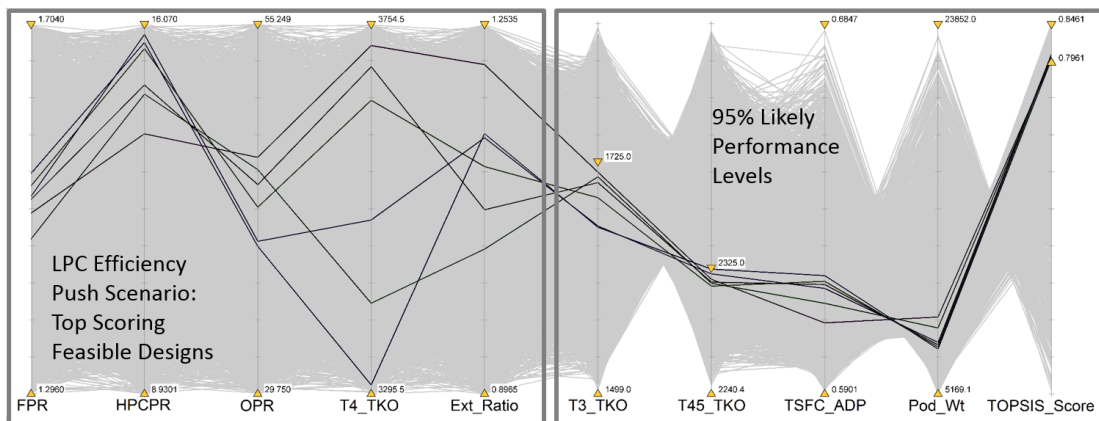


Figure 156: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, LPC efficiency distribution push.

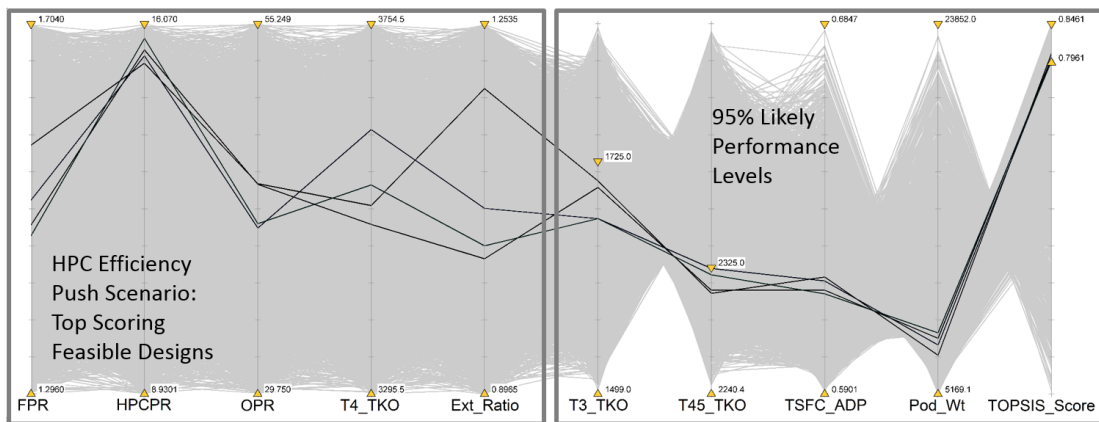


Figure 157: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, HPC efficiency distribution push.

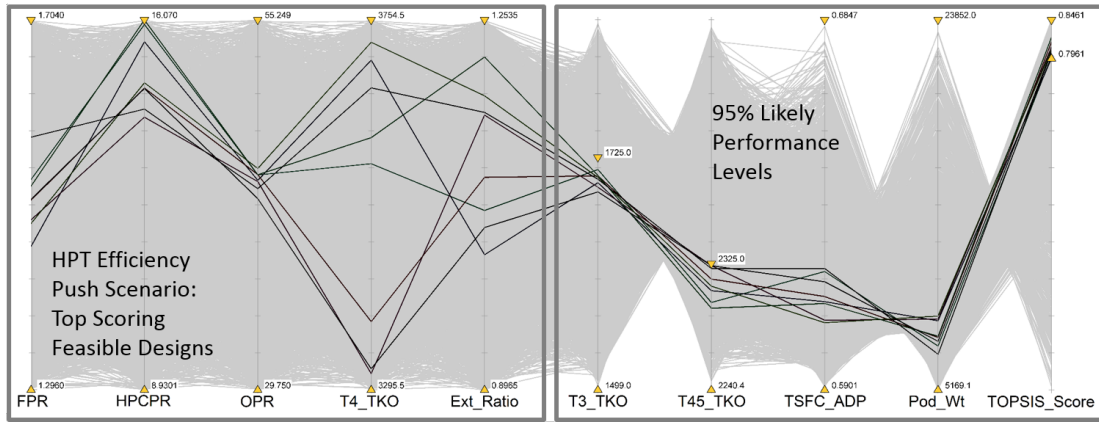


Figure 158: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, HPT efficiency distribution push.

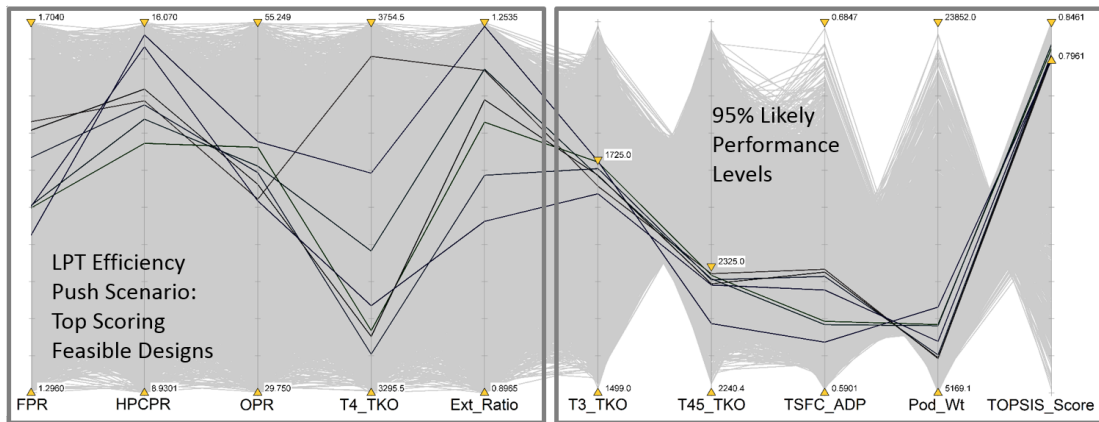


Figure 159: Experiment 1c - Constrained parallel coordinate chart of 95% likely performance, LPT efficiency distribution push.

Table 75: Experiment 1c - Highest scoring candidate cycle designs for each technology scenario considered and corresponding 95% likely performance levels.

Parameter	Technology Scenario					
	Base	Fan Push	LPC Push	HPC Push	HPT Push	LPT Push
π_{Fan}	1.363	1.364	1.364	1.355	1.364	1.364
π_{HPC}	15.875	15.880	15.880	15.836	15.880	15.880
$\pi_{Overall}$	44.937	45.060	45.060	45.402	45.060	45.060
$T_{4,max}, ^\circ R$	3,627.	3,612.	3,612.	3,612.	3,612.	3,612.
ER	1.199	1.176	1.176	1.212	1.176	1.176
TSFC at ADP, $lbm/(lb_f \cdot hr)$	0.6057	0.6041	0.6051	0.6045	0.6041	0.6032
Pod Weight, lbm	9,475.	9,481.	9,501.	9,467.	9,516.	9,497.

in the TSFC, while having a negligible increase in the engine weight. Therefore, given the technology advances assumed available, and assuming each advancement being equally expensive to develop, the *LPT Push* would be selected for advancement. However, it is likely that the variability in the remaining sources of uncertainty may need to be addressed and possibly reduced through testing in order to improve the 95% likely performance of the candidate engine cycles. Since significant uncertainty exists due to technology uncertainty, which is epistemic in nature and knowledge of the actual technology impacts could be improved, reduction of the uncertainty of these impacts could be pursued in order to further improve both the likely performance estimates as well as the resulting cycle selections.

E.3 Conclusions

This follow-on to Experiment 1 highlights an added capability of the ERDS method making it very flexible. By producing probabilistic surrogate models as functions of uncertainty distribution shape parameter settings, a wide variety of uncertainty scenarios can be simulated with a single set of surrogate models. Initial design decisions not only aim at the selection of the proper design variable settings, but concurrently consider various technology development options and corresponding design implications. The ERDS allows for such concurrent exercises to be performed in an efficient manner.

REFERENCES

- [1] “CFM Fleet Statistics.” <http://www.cfmaeroengines.com/engines>. Accessed Nov 29, 2012.
- [2] “Aircraft propulsion system performance station designation and nomenclature.” Aerospace Recommended Practice ARP755B, Society of Automotive Engineers, 1994.
- [3] “Aircraft propulsion system performance station designation.” SAE Aerospace Standard AS755, Revision F, 2014.
- [4] AIZERMAN, M. A., BRAVERMAN, E. M., and ROZONOER, L. I., “Theoretical foundations of the potential function method in pattern recognition learning,” *Automation and Remote Control*, vol. 25, pp. 821–837, 1964.
- [5] ATKINSON, K. E., *An Introduction to Numerical Analysis*. John Wiley & Sons, 1989.
- [6] AVERY, C. B., “Electrical generation and distribution for the more electric aircraft,” in *University of Bristol, UK, UPEC*, 2007.
- [7] BANDTE, O., MAVRIS, D. N., and DELAURENTIS, D. A., “Viable designs through a joint probabilistic estimation technique,” in *Proc. of the 4th World Aviation Congress and Exposition*, no. SAE-1999-01-5623, (San Francisco, CA), October 1999.
- [8] BATHIE, W. W., *Fundamentals of Gas Turbines*. John Wiley & Sons, Inc., 1996.
- [9] BATSON, R. G., “Cost risk analysis methodology: A state-of-the-art review,” *National Estimator: Journal of the National Estimating Society*, vol. 8, pp. 21–49, Spring 1989.
- [10] BEALE, M. H., HAGAN, M. T., and DEMUTH, H. B., *Neural network tool for use with MATLAB: user’s guide, R2011b*. Natick, MA: Mathworks, 2011.
- [11] BERTON, J. J. and GUYNN, M. D., “Multi-objective optimization of a turbofan for an advanced, single-aisle transport,” *Journal of Aircraft*, vol. 48, pp. 1795–1805, September-October 2011.
- [12] BILIEN, J. and MATTA, R., “The CFM56 venture,” in *Proc. of AIAA/AHS/ASEE Aircraft Design, Systems and Operations Conference*, AIAA-89-2038, (Seattle, WA), 1989.
- [13] BOYCE, M. P., *Gas Turbine Engineering Handbook*. Gulf Professional Publishing, 2006.
- [14] BOZER, B. E., GUYON, I. M., and VAPNIK, V. N., “A training algorithm for optimal margin classifiers,” in *Proc. of the fifth annual workshop on Computational learning theory - COLT*, p. 144, 1992.

- [15] BRADBROOK, S. J., “Common solutions to commercial and military propulsion requirements,” in *Proc. of 24th International Congress of the Aeronautical Sciences*, (Yokohama, Japan), 2004.
- [16] CAME, P. M. and ROBINSON, C. J., “Centrifugal compressor design,” *Proc. of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Sciences*, vol. 213, pp. 139–155, February 1998.
- [17] CARLIN, B. P. and LOUIS, T. A., *Bayes and Empirical Bayes Methods for Data Analysis*. Chapman and Hall, 1996.
- [18] CHARNIAK, E., “Bayesian networks witwith tears,” *AI Magazine*, vol. 12, pp. 50–63, 1991.
- [19] CHEN, W., JIN, R., and SUDJIANTO, A., “Analytical variance-based global sensitivity analysis in simulation-based design under uncertainty,” *Journal of Mechanical Design*, vol. 127, pp. 875–886, September 2005.
- [20] CLARK, J. P., KOCH, P. J., PUTERBAUGH, S. L., SCHMITZ, J. T., MORRIS, S. C., MA, R., and CORKE, T. C., “Highly loaded low-pressure turbine: Design, numerical and experimental analysis,” Interim Report AFRL-RZ-WP-TP-2010-2143, AFRL, Wright-Patterson AFB, OH, 2010.
- [21] COHEN, H., ROGERS, G. F. C., and SARAVANAMUTTOO, H. I. H., *Gas Turbine Theory*. Longman Scientific and Technical, 1987.
- [22] COLLOPY, P. D., “Surplus value in propulsion system design,” in *Proc. of 33rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, AIAA 97-3159, (Seattle, WA), 1997.
- [23] CORTES, C. and VAPNIK, V., “Support-vector networks,” *Machine Learning*, vol. 20, no. 3, p. 273, 1995.
- [24] CUMPSTY, N., *Jet Propulsion: A simple guide to the aerodynamic and thermodynamic design and performance of jet engines*. Cambridge University Press, 2nd ed., 2003.
- [25] CURLETT, B. P. and FELDER, J. L., “Object-oriented approach for gas turbine engine simulation,” Technical Memorandum TM-1995-106970, NASA, July 1995.
- [26] DASKILEWICZ, M. J., GERMAN, B. J., TAKAHASHI, T. T., DONOVAN, S., and SHAJANIAN, A., “Effects of disciplinary uncertainty on multi-objective optimization in aircraft conceptual design,” *Structural and Multidisciplinary Optimization*, vol. 44, pp. 831–846, December 2011.
- [27] DICKENS, T. and DAY, I., “The design of hihigh loload axial compressors,” *Journal of Turbomachinery*, vol. 133, pp. 031007–1 – 031007–10, July 2011.
- [28] DIETER, G. E. and SCHMIDT, L. C., *Engineering Design*. McGraw-Hill, fourth ed., 2009.
- [29] DIX, D. M. and GISSENDANNER, D. A., “Derivative engines versus new engines: What determines the choice?,” *Journal of Engineering for Gas Turbines and Power*, vol. 107, no. 4, pp. 808–814, 1985.

- [30] DU, X. and CHEN, W., “Sequential optimization and reliability assessment method for efficient probabilistic design,” *Journal of Mechanical Design*, vol. 126, pp. 225–233, March 2004.
- [31] EGLAJS, V. and AUDZE, P., “New approach to the design of multifactor experiments,” *Problems of Dynamics and Strengths*, vol. 35, pp. 104–107, 1977.
- [32] EPSTEIN, N., “CFM56-3 high by-pass technology for single aisle twins,” in *Proc. of AIAA/SAE/ASCE/ATRIF/TRB International Air Transportation Conference*, AIAA-81-0808, (Atlantic City, NJ), 1981.
- [33] FARLEY, B. G. and CLARK, W. A., “Simulation of self-organizing systems by digital computer,” *IRE Transactions on Information Theory*, vol. 4, no. 4, pp. 76–84, 1954.
- [34] FELLINI, R., KOKKOLARAS, M., and PAPALAMBROS, P. Y., “Quantitative platform selection in optimal design of product families, with application to automotive engine design,” *Journal of Engineering Design*, vol. 17, pp. 429–446, October 2006.
- [35] FLACK, R. D., *Fundamentals of Jet Propulsion with Applications*. Cambridge University Press, 2005.
- [36] FOWLER, T. W., *Jet Engines and Propulsion Systems for Engineers*. GE Aircraft Engines, 1989.
- [37] GLADIN, J., SANDS, J., KESTNER, B., and MAVRIS, D., “Effects of boundary layer ingesting (BLI) propulsion systems on engine cycle selection and HWB vehicle sizing,” in *Proc. of 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA 2012-0837, (Nashville, TN), 2012.
- [38] GREGORY, B. B., “Small engine technology investigation,” Contractor Report CR-2003-212519, NASA, 2003.
- [39] HEBB, D., *The Organization of Behavior*. New York: Wiley, 1949.
- [40] HWANG, C. L., LAI, Y. J., and LIU, T. Y., “A new approach for multiple objective decision making,” *Computers and Operational Research*, vol. 20, pp. 889–899, 1993.
- [41] HWANG, C. L. and YOON, K., *Multiple Attribute Decision Making: Methods and Applications*. New York: Springer-Verlag, 1981.
- [42] IMAN, R. L., DAVENPORT, J. M., and ZEIGLER, D. K., *Latin Hypercube sampling (program user’s guide)*. 1980.
- [43] IMAN, R. L., HELTON, J. C., and CAMPBELL, J. E., “An approach to sensitivity analysis of computer models, part 1. introduction, input variable selection and preliminary variable assessment,” *Journal of Quality Technology*, vol. 13, no. 3, pp. 174–183, 1981.
- [44] INSELBERG, A., “The plane with parallel coordinates,” *Visual Computer*, vol. 1, no. 4, pp. 69–91, 1985.
- [45] INSELBERG, A., *Parallel Coordinates: VISUAL Multidimensional Geometry and its Applications*. Springer, 2009.

- [46] JACQUET, J. and SEIWERT, D., “Methodology for commercial engine/aircraft optimization,” in *Proc. of AIAA/SAE/ASME/ASEE 29th Joint Propulsion Conference and Exhibit*, AIAA 93-1807, (Monterey, CA), 1993.
- [47] JAGGER, D. H., “New technology in the A320,” in *Proc. of AIAA/AHS/ASEE Aircraft Design, Systems and Operations Meeting*, AIAA-84-2444, (San Diego, CA), 1984.
- [48] JARRELL, S. B., *Basic Statistics*. Dubuque, Iowa: Wm. C. Brown Pub, 1994.
- [49] JONES, S. M., “An introduction to thermodynamic performance analysis of aircraft gas turbine engine cycles using the Numerical Propulsion System Simulation code,” Technical Memorandum TM-2007-214690, NASA, 2007.
- [50] KESTNER, B. K., PERULLO, C. A., SANDS, J. S., and MAVRIS, D. N., “Bayesian belief network for robust engine design and architecture selection,” in *Proc. of the ASME Turbo Expo 2014*, GT2014-27017, (Düsseldorf, Germany), 2014.
- [51] KESTNER, B., JIMENEZ, H., PERULLO, C., SCHUTTE, J., and MAVRIS, D., “Identifying key technology areas to fundamentally reduce risk in engine performance and noise for future commercial applications,” in *Proc. of ASME Turbo Expo*, GT2013-95695, (San Antonio, TX), 2013.
- [52] KESTNER, B., SCHUTTE, J., GLADIN, J., and MAVRIS, D., “Ultra high bypass ratio engine sizing and cycle selection study for a subsonic commercial aircraft in the N+2 timeframe,” in *Proc. of ASME Turbo Expo*, GT2011-45370, (Vancouver, British Columbia, Canada), 2011.
- [53] KESTNER, B. K., MARTIN, K., PERULLO, C. A., SCHUTTE, J., and MAVRIS, D. N., “Integrated system design using bayesian belief networks,” in *Proc. of 51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, AIAA 2013-0617, (Grapevine, Texas), January 2013.
- [54] KESTNER, B. K., SCHUTTE, J. S., TAI, J. C. M., PERULLO, C. A., and MAVRIS, D. N., “Surrogate modeling for simultaneous engine cycle and technology optimization for next generation subsonic aircraft,” in *Proc. of ASME Turbo Expo*, GT2012-68724, (Copenhagen, Denmark), 2012.
- [55] KIRBY, M. R. and MAVRIS, D. N., “Forecasting technology uncertainty in preliminary aircraft design,” in *World Aviation Conference*, 1999-01-5631, (San Francisco, CA), AIAA and SAE, 1999.
- [56] KOFF, B. L., “F100-PW-229 higher thrust in same frame size,” *Journal of Engineering for Gas Turbines and Power*, vol. 111, pp. 187–192, 1989.
- [57] KOFF, B. L., “Gas turbine technology evolution: A designer’s perspective,” *Journal of Propulsion and Power*, vol. 20, no. 4, pp. 577–595, 2004.
- [58] KURZKE, J., “Gas turbine cycle design methodology: A comparison of parameter variation with numerical optimization,” *Journal of Engineering for Gas Turbines and Power*, vol. 121, pp. 6–11, 1999.

- [59] KYRITSIS, V. E. and PILIDIS, P., “Principles of thermodynamic preliminary design of civil turbofan engines,” in *Proc. of ASME Turbo Expo 2009: Power for Land, Sea and Air*, GT2009-59815, (Orlando, FL), 2009.
- [60] LEE, Y. K., MAVRIS, D. N., VOLOVOI, V. V., YUAN, M., and FISHER, T., “A fault diagnosis method for industrial gas turbines using bayesian data analysis,” *Journal of Engineering for Gas Turbines and Power*, vol. 132, p. 041602, January 2010.
- [61] LEHMANN, E. A., “Multiple application core engine: Sizing and usage criteria,” *Journal of Aircraft*, vol. 17, no. 11, pp. 802–809, 1980.
- [62] LI, H. and AZARM, S., “Product design selection under uncertainty and with competitive advantage,” *Journal of Mechanical Design*, vol. 122, pp. 411–418, December 2000.
- [63] LYTLE, J., FOLLEN, G., NAIMAN, C., EVANS, A., VERES, J., OWEN, K., and LOPEZ, I., “Numerical Propulsion System Simulation (NPSS) 1999 industry review,” Technical Memorandum TM-2000-209795, NASA, 2000.
- [64] LYTLE, J., FOLLEN, G., NAIMAN, C., VERES, J., OWEN, K., and LOPEZ, I., “2001 Numerical Propulsion System Simulation review,” Technical Memorandum TM-2002-211197, NASA, 2002.
- [65] LYTLE, J. K., “The Numerical Propulsion System Simulation: A multidisciplinary design system for aerospace vehicles,” Technical Memorandum TM-1999-209194, NASA, 1999.
- [66] LYTLE, J. K., “The Numerical Propulsion System Simulation: An overview,” Technical Memorandum TM-2000-209915, NASA, 2000.
- [67] MAST, T. A., REED, A. T., YURKOVICH, S., ASHBY, M., and ADIBHATLA, S., “Bayesian belief networks for fault identification in aircraft gas turbine engines,” *Proc. of the 1999 IEEE International Conference on Control Applications*, vol. 1, pp. 39–44, 1999.
- [68] MATTINGLY, J. D., *Elements of Gas Turbine Propulsion*. McGraw-Hill, 1996.
- [69] MATTINGLY, J. D., HEISER, W. H., and PRATT, D. T., *Aircraft Engine Design*. American Institute of Aeronautics and Astronautics, 2002.
- [70] MAVRIS, D. and BANDTE, O., “Comparison of two probabilistic techniques for the assessment of economic uncertainty,” in *Proc. of 19th Annual Conference of the International Society of Parametric Analysts*, June 1997.
- [71] MAVRIS, D. N., BANDTE, O., and DELAURENTIS, D. A., “Robust design simulation: A probabilistic approach to multidisciplinary design,” *Journal of Aircraft*, vol. 36, pp. 298–307, January-February 1999.
- [72] MAVRIS, D. N., MACSOTAI, N. I., and ROTH, B., “A probabilistic design methodology for commercial aircraft engine cycle selection,” in *Proc. of 3rd World Aviation Congress and Exposition*, AIAA 1998-5510, (Anaheim, CA), 1998.

- [73] McCULLOCH, W. and PITTS, W., “A logical calculus of ideas immanent in nervous activity,” *Mathematical Biophysics*, vol. 5, no. 4, pp. 115–133, 1943.
- [74] MCKAY, M. D., BECKMAN, R. J., and CONOVER, W. J., “A comparison of three methods for selecting values of input variables in the analysis of output from a computer code,” *Technometrics (JSTOR) Abstract - American Statistical Association*, vol. 21, no. 2, pp. 239–245, 1979.
- [75] MCLOUGHLIN, A., “More electric - ready for take off?,” in *Proc. of 13th European Conference Power Electronics and Applications*, 2009.
- [76] MOLINO, N., SANDS, J., DUNCAN, S., OSVALDS, E., and MAVRIS, D., “Improved pareto optimal engine cycle designs through the use of a new pareto quality indicator,” in *Proc. of 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, AIAA 2012-3814, (Atlanta, GA), 2012.
- [77] MONTGOMERY, D., *Design and Analysis of Experiments*. Hoboken, NJ: John Wiley & Sons, 8 ed., 2013.
- [78] MORELLO, J. R., “Economic benefits of engine technology to commercial airline operators,” in *Proc. of AIAA/SAE 11th Propulsion Conference*, AIAA-75-1205, (Anaheim, CA), 1975.
- [79] MOUSTAFA, R. and WEGMAN, E. J., *Multivariate continuous data Parallel Coordinates, found in Graphics of Large DataData: Visualizing a Million*. Springer, 2006.
- [80] MOXON, J., “GE joins fight for world’s largest turbofan,” *Flight International*, p. 0151, 1990.
- [81] NASA, *N.P.S.S. developer’s guide, software release: N.P.S.S_1.6.4, rev:Q*, 2006.
- [82] NASA, *N.P.S.S. user guide, software release: N.P.S.S_1.6.4, rev:Q*, 2006.
- [83] NASA, *N.P.S.S. reference sheets, software release: N.P.S.S_1.6.4, rev:Q*, 2007.
- [84] NATO, “Performance prediction and simulation of gas turbine engine operation for aircraft, marine, vehicular, and power generation,” Tech. Rep. Tech. Rep. TR-AVT-036, Annex B.1 - Compressor Systems Performance, 2007.
- [85] OATES, G. C., *Aircraft Propulsion Systems Technology and Design*. American Institute of Aeronautics and Astronautics, 1989.
- [86] OATES, G. C., *Aerothermodynamics of gas turbine and rocket propulsion*. American Institute of Aeronautics and Astronautics, 1997.
- [87] PANCHENKO, Y., HOUSTAPHA, H., MAH, S., PATEL, K., DOWHAN, M. J., and HALL, D., “Preliminary multi-disciplinary optimization in turbomachinery design,” in *Proc. of The RTO Applied Vehicle Technology Panel (AVT) Symposium*, (Paris, France), 2002.
- [88] PEIRCE, C. S., “Note on the theory of the economy of research,” *Operations Research*, vol. 15, pp. 643–648, July-August 1967.

- [89] PRESS, W. H., TEUKOLSKY, S. A., VETTERLING, W. T., and FLANNERY, B. P., *Numerical Recipes: the Art of Scientific Computing (3rd ed) - Section 16.5. Support Vector Machines*. New York: Cambridge University Press, 2007.
- [90] ROCHESTER, N., HOLLAND, J., HABIT, L., and DUDA, W., “Ttest on a cell assembly theory of the action of the brain, using a large digital computer,” *IRE Transactions on Information Theory*, vol. 2, no. 3, pp. 80–93, 1956.
- [91] ROLLS-ROYCE, “Low density materials - titanium alloys.” Web. Accessed 09/16/2013.
- [92] ROMESSIS, C. and MATHIOUDAKIS, K., “Bayesian network approach for gas path fault diagnosis,” *Journal of Engineering for Gas Turbines and Power*, vol. 128, p. 64, 2006.
- [93] ROSENBLATT, F., “The perceptron: A probabilistic model for information storage and organization in the brain,” *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958.
- [94] ROTH, B., GERMAN, B., MAVRIS, D., and MACSOTAI, N., “Adaptive selection of engine technology solution sets from a large combinatorial space,” in *Proc. of 37th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, AIAA2001-3208, (Salt Lake City, UT), AIAA, July 2001.
- [95] ROTH, B., GRAHAM, M., MAVRIS, D., and MACSOTAI, N., “Adaptive selection of pareto optimal engine technology solution sets,” in *Proc. of 23rd Congress of International Council of the Aeronautical Sciences*, 2002.
- [96] ROTH, B. and MAVRIS, D., “Commercial engine architecture selection in the presence of uncertainty and evolving requirements,” in *Proc. of Fifteenth International Symposium on Air Breathing Engines*, ISABE 2001-1169, (Bangalore, India), 2001.
- [97] ROTH, B., MAVRIS, D., and ELLIOTT, D., “A probabilistic approach to UCAV engine sizing,” in *Proc. of 34th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, AIAA98-3264, (Cleveland, OH), July 1998.
- [98] RYABEN’KIL, V. S. and TSYNKOV, S. V., *A Theoretical introduction to numerical analysis*. CRC Press, 2006.
- [99] SANDS, J. S., PERULLO, C. A., KESTNER, B. K., and MAVRIS, D. N., “Enhanced robust design simulation and application to engine cycle and technology design,” in *Proc. of the ASME Turbo Expo 2014*, GT2014-27137, (Düsseldorf, Germany), 2014.
- [100] SANDS, J., GLADIN, J., KESTNER, B., and MAVRIS, D., “Hybrid wing body engine cycle design exploration for boundary layer ingesting (BLI) propulsion systems under design uncertainty,” in *Proc. of 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, AIAA 2012-3918, (Atlanta, GA), 2012.
- [101] SCHRAGE, D. P. and MAVRIS, D. N., “Technology for affordability - how to define, measure, evaluate, and implement it,” in *Proc. of 50th National Forum of the American Helicopter Society*, (Washington, D.C.), May 1994.

- [102] SCHUTTE, J., *Simultaneous Multi-Design Point Approach to Gas Turbine On-Design Cycle Analysis for Aircraft Engines*. PhD thesis, Georgia Institute of Technology, 2009.
- [103] SCHUTTE, J., TAI, J., SANDS, J., and MAVRIS, D., “Cycle design exploration using multi-design point approach,” in *Proc. of ASME Turbo Expo*, GT2012-69334, (Copenhagen, Denmark), 2012.
- [104] SIMPSON, T. W., “Balancing commonality and performance within the concurrent design of multiple products in a product family,” *Concurrent Engineering: Research and Applications*, vol. 9, pp. 177–190, September 2001.
- [105] SMARSLY, W., “Aero engine materials,” in *In Seminar, Faculty of Mechanical Engineering*, (Cracow University of Technology), 2004.
- [106] SNYDER, C. A. and THURMAN, D. R., “Gas turbine characcharacter for a large civil tilt-rotor (LCTR),” Technical Memorandum NASA/TM-2010-216089, NASA, 2010.
- [107] STEINMETZ, R. B. and WAGNER, M. J., “Turbofan engine cycle design selection - year 2000,” in *Proc. of AIAA/AHS/ASEE Aircraft Systems Design & Technology Meeting*, AIAA 86-2705, (Dayton, OH), 1986.
- [108] STRICKER, J. M., “Turbine engine affordability,” in *Proc. of 38th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, AIAA 2002-3619, (Indianapolis, IN), 2002.
- [109] SUDER, K. L., “Overview of the NASA Environmentally Responsible Aviation project’s propulsion technology portfolio,” in *Proc. of 48th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, AIAA 2012-4038, (Atlanta, GA), July 2012.
- [110] TOLSTYKH, M., KARL, A., and FARRIS, B., “Robust design technique used in multi-variable probabilistic design of gas turbine components,” in *Proc. of 12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference*, AIAA-2012-5636, (Indianapolis, IN), 2012.
- [111] TONG, M. T., HALLIWELL, I., and GHOSN, L. J., “A computer code for gas turbine weight and disk life estimation,” *Journal of Engineering for Gas Turbines and Power*, vol. 126, pp. 265–270, April 2004.
- [112] UTTS, J. M., *Seeing Through Statistics*. Thomson Brooks/Cole, 3 ed., 2005.
- [113] VANDERPLAATS, G. N., *Multidiscipline Design Optimization*. Vanderplaats Research & Development, Inc., 2007.
- [114] VERES, J. P. and THURMAN, D. R., “Conceptual design of a two spool compressor for the NASA Large Civil Tilt Rotor engine,” in *Proc. of 66th Annual AHS International Forum & Technology Display*, (Phoenix, Arizona), May 2010.
- [115] WALSTON, W. S., O’HARA, K. S., ROSS, E. W., POLLOCK, T. M., and MURPHY, W. H., “Rene N6: Third generation single crystal superalloy,” in *Proc. of Superalloys 1996 Eighth International Symposium*, pp. 27–34, 1996.

- [116] WU, C. F. J. and HAMADA, M. S., *Experiments: Planning, Analysis, and Optimization*. Wiley, 2009.
- [117] YAFFEE, M. L., “Developers face 1975 CFM56 decision,” in *Aviation Week & Space Technology*, p. 41, February 24 1975.
- [118] YOON, K., “A reconciliation among discrete compromise situations,” *Journal of Operational Research Society*, vol. 38, pp. 277–286, 1987.
- [119] YOUNGHANS, J. L., DONALDSON, R. M., WALLACE, D. R., LONG, L. L., and STEWART, R. B., “Preliminary design of low cost propulsion systems using next generation cost modeling techniques,” *Journal of Engineering for Gas Turbines and Power*, vol. 121, no. 1, pp. 1–5, 1999.
- [120] YPMA, T. J., “Historical development of the newton-raphson method,” *SIAM Review*, vol. 37, no. 4, pp. 531–551, 1995.

VITA

Jonathan Sands grew up in the Triangle area of North Carolina, where he attended North Carolina State University. He obtained his Bachelor of Science degree in aerospace engineering in 2008. He then attended the Georgia Institute of Technology in Atlanta Georgia, receiving his Master of Science degree in aerospace engineering in 2010. He continued his academic career at Georgia Tech pursuing his Doctor of Philosophy degree. His research focused on gas turbine propulsion systems design methods, producing several publications. After completing his coursework at Georgia Tech, he has since begun his career in industry as a performance design engineer at Rolls-Royce in Indianapolis Indiana.