On the expressiveness of spatial constraint systems

Michell Guzman

universite PARIS-SACLAY

NNT : 2017SACLX064

On the Expressiveness of Spatial Constraint Systems

Thèse de doctorat de l'Université Paris-Saclay
préparée à l'École Polytechnique

École doctorale n°580 Sciences et technologies de l'information et de la communication (STIC): Informatique

Thèse présentée et soutenue à Palaiseau, le 26 septembre 2017, par

M Michell Guzmán Cancimance

Composition du Jury :

Eric Goubault Professeur, École Polytechnique (LIX)	Président
Jean-Louis Giavitto Directeur de Recherche, CNRS (IRCAM)	Rapporteur
Marco Comini Professeur, Università di Udine (Dipartimento di Informatica)	Rapporteur
Julián Gutierrez Maître de Conférences, University of Oxford (Comp. Sci. Dept.)	Examinateur
Catuscia Palamidessi Directrice de Recherche, INRIA (LIX)	Directrice de thèse
Frank Valencia Professeur & CR1, Universidad Javeriana Cali & CNRS (LIX)	Co-Directeur de thèse

Thèse de doctorat

Abstract

Epistemic, mobile and spatial behaviour are commonplace in today's distributed systems. The intrinsic *epistemic* nature of these systems arises from the interactions of the elements of which they are comprised. Most people are familiar with digital systems where users share their *beliefs*, *opinions* and even intentional *lies* (hoaxes). Models of such systems must take into account the interactions with others as well as the distributed quality presented by them. Spatial and mobile behaviour are exhibited by applications and data moving across possibly nested spaces defined by, for example, friend circles, groups, and shared folders. Thus a solid understanding of the notion of *space* and *spatial mobility* as well as the flow of epistemic information is relevant in many models of today's distributed systems.

In order to analyze knowledge, space, and mobility in distributed systems, we expand upon the mathematically simple and elegant theory of *constraint systems* (cs), used to represent information and information change in concurrent systems. In the formal declarative model known as *concurrent constraint programming*, constraint systems provide the basic domains and operations for the semantic foundations of this model. *Spatial constraint systems* (scs's) are algebraic structures that extend cs's for reasoning about basic spatial and epistemic behaviour such as *belief* and *extrusion*. Both, spatial and epistemic assertions, can be viewed as specific modalities. Other modalities can be used for assertions about time, knowledge and even the analysis of groups among other concepts used in the specification and verification of concurrent systems.

In this thesis we study the expressiveness of spatial constraint systems in the broader perspective of modal and epistemic behaviour. We shall show that spatial constraint systems are sufficiently robust to capture inverse modalities and to derive new results for modal logics. We shall show that we can use scs's to express a fundamental epistemic behaviour such as *knowledge*. Finally we shall give an algebraic characterization of the notion of *distributed information* by means of constructors over scs's.

Résumé

Les comportement épistémiques, mobiles et spatiaux sont omniprésent dans les systèmes distribués d'aujourd'hui. La nature épistémique intrinsèque de ces types de systèmes, provient des interactions des éléments qui en font parties. La plupart des gens sont familiarisés avec les systèmes numériques où les utilisateurs partagent leurs *croyances*, *opinions* et même des *mensonges* intentionnels (des canulars). Les modèles de ces systèmes doivent prendre en compte des interactions avec les autres ainsi que leurs qualité distribuée. Les comportements spatiaux et mobiles sont exposées par des applications et des données qui se déplaçant à travers des espaces (éventuellement imbriqués) définis par, par example, des cercles d'amis, des groupes et des dossiers communs. Nous croyons donc qu'une solide compréhension de la notion d'espace et mobilité spatial ainsi que le flux d'information épistémique est pertinente dans de nombreux modèles de systèmes distribués d'aujourd'hui.

Afin d'analyser la connaissance, l'espace et la mobilité dans les systèmes distribués, nous élargissons sur la théorie mathématiquement simple et élégante des *systèmes de contraintes* (sc), utilisée pour représenter l'information et le changement d'information dans les systèmes concurrents. Dans le modèle déclaratif formel connu sous le nom de *programmation concurrent par contraintes*, les systèmes de contraintes fournissent les domaines de base et les opérations pour les fondements sémantiques de ce modèle. Les *systèmes des contraintes spatiales* (scs) sont des structures algébriques qui étendent les sc pour le raisonnement sur les comportement spatiaux et épistémiques de base tels que la *croyance* et *l'extrusion*. Les affirmations spatiales et épistémiques peuvent être considérées comme des modalités spécifiques. D'autres modalités peuvent être utilisées pour les assertions concernant le temps, les connaissances et même pour l'analyse des groupes parmi d'autres concepts utilisés dans la spécification et la vérification des systèmes concurrents.

Dans cette thèse nous étudions l'expressivité des systèmes de contraintes spatiales dans la perspective plus large du comportement modal et épistémique. Nous montrerons que les systèmes de contraintes spatiales sont suffisamment robustes pour capturer des modalités inverses et pour obtenir de nouveaux résultats pour les logiques modales. Également, nous montrerons que nous pouvons utiliser les scs pour exprimer un comportement épistémique fondamental comme *connaissance.* Enfin, nous allons donner une caractérisation algébrique de la notion de *l'information distribuée* au moyen de constructeurs sur scs.

Acknowledgments

It is the end of this journey and it is important to try to thank all the people who transformed it into a very special and an unique experience. First of all, I would like to thank my supervisors Frank Valencia and Catuscia Palamidessi for giving me the opportunity to work with them. To Frank, to whom I owe the opportunity to come here and live this amazing experience, I just have words of respect and gratitude. Frank is a brilliant researcher and a better person, the passion that he has for what he does is amazing and he is an extremely supportive person. I am in dept with him and I hope someday I have the opportunity to pay him back. Thanks also to his very nice family, Felipe and Sara for all their kindness. Thanks to Catuscia, who is a great leader and a very kind person, for all her support and aid during these years. The way she works is just incredible and shows her commitment with the research community. I thank her and Dale Miller also for opening the doors of their home to me. I always felt very welcome there.

I want to express my gratitude to Camilo Rueda. He is a very insightful researcher and he is one of my inspirations, I enjoyed very much our chats about science and music in the RER in our way back to Paris. Thanks a lot to Sophia Knight for giving me the chance to work with her at Uppsala university, it was an amazing experience for me. I want to thank her also for helping me in my postdoctoral applications, that meant a lot to me. Thanks also to Joachim Parrow and Bjorn Victor for the great feedback about my work. Thanks to my colleagues at Uppsala university, Arve, Anke and Jan for making me feel part of their team.

I want to thank to Carlos Olarte and Elaine Pimentel. Carlos is one of the person that I admire the most, he is an example of a great and a hard worker researcher. I want to thank him for taking my ideas into account, for trying to understand what I was trying to do. I want to thank him for taking the time to read carefully this dissertation and for all the corrections that helped to improve the quality of this manuscript. I also want to show my huge gratitude to Juan Francisco

Diaz, Jesus Aranda and Angela Villota for all their support during this process to become a researcher. Juan Francisco is my research father. He is not just a great teacher and researcher but the nicest person toward his students that I ever met. I always admired him for finding the time to receive his students no matter how busy he was and for finding the right words to address his students in different aspects of life, that was always impressive to me. Thanks to all my teachers at Univalle for inspiring me during my studies. In particular I want to thank to Ángel Garcia, Santiago Gonzales, Oscar Bedoya, Martha Millán, Pedro Moreno and Doris Hinestroza.

Thanks to INRIA for funding my PhD and to École Polytechnique for hosting me during my doctoral studies. Thanks also to all the administrative staff of INRIA and École Polytechnique. Thanks to Valérie Berthou, Hélèna Kutniak, Faouzi Rahali, Juana Mallmann, Evelyne Rayssac, Audrey Lemarechal, Emmanuel Fullenwarth, Alexandra Belus and Jessica Gameiro.

I would like to thank to the reviewers and the members of the jury of my thesis for taking the time to read and evaluate my manuscript, for all the insightful comments that helped to improve this manuscript.

My gratitude goes to all my colleagues, teammates and friends at LIX; Banjamin, Youcef, Salim, Vanessa, Valeria, Bibek, Adina, Marco Stronati, Marco Volpe, Nicolas, Sonia, Kostas, Ehab, Yusuke, Panayotis, Thomer, Beniamino, Gissele, Amelie, Hector, Santiago, Marco Romanelli, Gabriele, Ali, Joaquin, Giorgi, Anna and Timo.

I am grateful to all the great people that I met in France during these three years. Thanks for all the good moments we spent together. Special thanks go to Thomas, Sebastián, Marco Schmid, Michal, Bao, Arnaud, Florie, Behrang, Dominykas, Momo, Simo, Joan, Ceren, Hana, Elena, Denize, Delphine, Kajsa, Tinde, Helen, Aude, Oscar, Daniel Wamo, Daniel Ortiz, Damien, James, Willem, Catalina, Jonas, Jasmine, Luis Estévez, Simon, Marcelo, Marwan, Chahinaz, Nada, Asmae, Georg, Adrian, Francesco, Mohamed Hawari, Luiza, Zineb, Gael, Lucas, Carlos, Ramón, Anna, Christos, Yves, Hussein, Dorian, Sami, Anaelle, Silvia, Aldo, Giorgio, Barbara, Cosimo, Olga, Manuel, Eduardo and Eduwin. I just want them to know that without them my whole experience would have not be so great as it was. Thanks to the people from MEASE, in particular to Marie-Françoise Candé and Eduardo de Farias. Thanks also to my music teacher Stanislas Denussac for all the things that he taught me during the time I took lessons with him.

I would like to thank to all my friends in Colombia for all their support. In particular I would like to thank Oliver for the exchange of messages during these years, to Jhonatan, Mario, Cesar, Diego, David, Santiago, Julián, Daniel Valencia, Jaime Arias, Carlos Ramirez, Carlos Delgado, John Gómez, Federico, Luis Pino, Margarita, Carlos Mario Borrero, Carolina, Elizabeth, Paula, Maria Andrea, Andrés ("Puerto"), Lorena, Christian, Gicela, Oscar Tascón, Luis Varela, Christian Arce, Victor Riascos, Victor Tina, Victor ("Cebú"), Larry, Moises, Lucia, Alan, Carlos Holmes, Jhon Edwin. I want to thank also to my neighbours and to all my people from El Cabuyal for all the good wishes they were sending me during these three years. I would like to dedicate some few lines to the memory of my friend Junior who passed away a couple of months ago. Thank you Junior for everything, for all the good times we shared together, for all the smiles and the good vibes, may you rest in peace my friend.

Finally, I would to express my gratitude to my family for all their support, for all the love they give to me, I am in debt with them. I have no words to express how important they are for me. I want to thank for everything specially to my parents. To my mother Rubiela for all the sleepless nights, for all the effort she made to wake herself up very early in the morning to provide me the necessary things for the day, for all her love towards me. To my father Jaiber for all the support he gave to me all these years, for picking me up at very late hours when there were no more buses available to go back home, I feel very lucky to have you by my side. To my grandmother Isa for all her love and care. To my sister Karen for caring too much about me all the time. To my brother Jorge for keeping me up to date with the good music and for making me look like a white sheep in the family.

Contents

List of Figures

One

Introduction

1.1 Introduction

Constraint systems (cs's) represent a central notion of *concurrent constraint programming* (ccp) [SRP91, PSSS93]. In ccp constraint systems are used as algebraic structures for the semantics of process calculi. They also specify the domain, elementary operations and partial information upon which programs (processes) of these calculi may act. This thesis shows that spatial constraint systems are expressive enough to give semantics and derive inverse operators for modal logics as well as to characterize the epistemic notions of knowledge and distributed information.

Todays' systems are govern by entities which are disseminated across different positions with the ability to communicate their own information. In such systems we have *agents* that interact by *exchanging* their *local* information and programs. The need for understanding such systems require to take into account not only the local information owned by the agents but also the *distance* and *position* of such entities in the system.

Different abstractions have been proposed in order to model spatially distributed systems. Some of these abstractions aim to characterize the notion of space as the *local* information the agents have (*localization*). Others center their attention in the *distance* among the entities in the systems. We can also find abstractions in which the focus is given to features such as *position* and *shape*. The study of systems performing computations *dis-*

tributed in space, where the *position* and the *distance* of the entities matters, is known as *spatial computing* [DGG07]. In spatial computing, finding the right abstraction for space rise as a hard task considering the different parameters that we need to take into account. As above mentioned, often abstractions focus in some of the main features of spatial computing. A well known formalism used in order to represent the notion of space from a biological perspective is the *Brane calculi* [Car04]. In brane calculi the main attention is given to the representation of *biological membranes*. *Membranes* in Brane calculi represent the abstraction for the place in which reactions can occur. In *ambient calculi*, which is one of the most well know abstractions for spatial computing, the interest is given to the notion of *spatial mobility* [CG98]. Ambient calculi combine in the same abstraction a representation not only for *mobile devices* but also for the information they exchange. In this formalism we have mobile *agents* which interact in places called *ambients*. One of the main features in ambient calculi is that the mobility of the ambients is allowed. Ambient calculi provide for the specification of processes that can move in and out within their spatial hierarchy. It does not, however, address posting and querying epistemic information within a spatial distribution of processes. In [CC03, CC02] the authors propose an abstraction for spatial computing taking into account the notion of *spatial location* as the fundamental concept. They do this by developing modalities over name quantifiers that reflect locality.

Recently, a family of algebraic structures have been proposed in order to provide a formalism for spatial computing borrowing design ideas from the ambient calculi. This algebraic structures were proposed in order to model *spatial location* and *spatial mobility*. These algebraic structures are build upon the notion of constraint systems.

A cs can be formalized as a complete lattice $(Con, \sqsubseteq)$. The elements of Con represent the partial information and we shall think of them as being *assertions*. These elements are traditionally referred to as *constraints* since they naturally express partial information (e.g., $x > 42$). The order $\sqsubseteq$ corresponds to entailment between constraints. Then $c \sqsubseteq d$ means that c can be derived from d, or that d represents as much information as c.

Consequently, the join $\sqcup$, the bottom *true* and the top *false* of the lattice correspond respectively to entailment, conjunction, the empty information and the join of all (possibly inconsistent) information.

Every operator in ccp has a corresponding elementary construct on the constraints. Programs in ccp can be interpreted as both computational entities and logic specifications. For example, the parallel composition and the conjunction correspond to the *join* operation. The same principle applies for the notions of computational space and the epistemic notion of belief in *spatial concurrent constraint programming* (sccp) [KPPV12]. There, these notions coincide with a family of functions $[\cdot]_i : Con \to Con$ on the elements of the constraint system Con that preserve the join operation. These functions are called *space functions.* A cs equipped with space functions is called a *spatial constraint system* (scs). From a computational point of view, given $c \in Con$ the assertion (constraint) $[c]_i$ specifies that c resides within the space of agent i. From an epistemic point of view, the assertion $[c]_i$ specifies that agent i considers c to be true (i.e. that in the world of agent i assertion c is true). Both intuitions convey the idea of c being local (subjective) to agent i. The authors in [HPRV15] extended the notion of scs's in order to provide these algebraic structures with a *mobile* behaviour. They equip every space function $[\cdot]_i$ in the spatial constraint systems with a *right inverse* of $[\cdot]_i$, called *extrusion function*, satisfying some basic requirements (e.g., preservation of the join operation). By right inverse of $[\cdot]_i$ we mean a function $\uparrow_i : Con \to Con$ such that $[\uparrow_i c]_i = c$. The computational interpretation of $\uparrow_i$ is that of a process being able to extrude any c from the space $[\cdot]_i$. From an epistemically perspective, we can use $\uparrow_i$ to express *utterances* by agent i, e.g., to let others see agent i believes c. One can then think of extrusion/utterance as the *right inverse* of space/belief. This extension of scs's was called *spatial constraint systems with extrusion* (scse).

Therefore, a clear understanding of the expressive power of these structures is very important for the analysis of programs and models that we can build on them.

In this thesis we shall explore three approaches to show the expressive power of spatial constraint systems. It is important to mention that we shall tackle expressiveness in the sense of being able to model a given behaviour. First, we use scse's to *derive* inverse operators for modal logics. We also give the minimal conditions for which such right inverses exists for a specific family of scs's. Then we propose a representation for the epistemic notion of knowledge in scs's. This representation of knowledge is given by means of a derived operator over the scs satisfying the knowledge axioms. Finally, we give an algebraic characterization for the notion of distributed information. This characterization is given through the use of derived operators on scs's.

The last part of the introduction is devoted to motivate and shortly describe the main contributions of this thesis.

1.1.1 The Extrusion Problem

Given a spatial function $[\cdot]_i$, the *extrusion problem* consists in finding (constructing) a right inverse (extrusion function) of $[\cdot]_i$. As mentioned above, by building these right inverses we are equipping scs's with the necessary operations to characterize the mobile behaviour of multi-agent systems.

Modal logics [Pop94] extend classical logic to include operators expressing *modalities*. Depending on the intended meaning of the modalities, a particular modal logic can be used to reason about space, knowledge, belief or time, among others. One is often interested in *normal* modalities: Roughly speaking, a modal operator m is normal in a given modal logic system if (1) $\mathrm{m}(\phi)$ is a theorem whenever the formula ϕ is a theorem, and (2) the implication formula $\mathrm{m}(\phi \Rightarrow \psi) \Rightarrow (\mathrm{m}(\phi) \Rightarrow \mathrm{m}(\psi))$ is a theorem.

Although the notion of spatial constraint system is intended to give an algebraic account of spatial and epistemic assertions, we shall show in this dissertation that it is sufficiently robust to give an algebraic account of more general *modal logic* assertions. The main focus in this part of the thesis is the study of the above-mentioned *extrusion problem* for a meaningful family of scs's that can be used as semantic structures for modal logics. They are called *Kripke spatial constraint systems* because their elements are sets of *Kripke Structures* (KS's). These structures are a fundamental mathematical tool in logic and computer science and they can be seen as state transition graphs (i.e., transition systems) with some additional structure on their states.

We already know from the the literature of *domain theory* that we can build right inverses for $f(\cdot)$ iff $f(\cdot)$ is surjective. However, we are interested into give explicit right inverse constructions and to find which are the minimal conditions for which these exists.

1.1.2 Knowledge in Terms of Space

The notions of *belief* and *knowledge* had always been thought as related in some sense. More specifically an important question that arises in the epistemic world has been whether knowledge can be represented by means of belief. One of the proposed ideas in order to answer this question says that the notion of knowledge can be represented in terms of belief as *justified true belief* (jtb) [HSS09]. In a general way, jtb can be defined as *everything we belief must be true*. However this approach was never neither totally rejected, although the authors in [Get63] have presented some counterexamples to this representation, nor proved. In this dissertation we shall study this representation by means of modal logics. One of the first authors who studied these notions as modal logic operators was Hintikka [Hin62], creating this way the basics of modal logics as they are known these days.

In [HSS09] the authors have addressed this question following Hintikka's legacy and having as the framework for reference the epistemic and doxastic logics. The authors made use of the theory of jtb which states that the *definability* of knowledge is possible. They gave some proofs about when the epistemic $S5$ logic can or cannot be defined in terms of belief. The logic of belief they use is the $KD45$ belief. They also consider different definitions of definability, mainly when a definition is given in an *implicit* or *explicit* way. We shall instead give an algebraic treatment to the $S4$ knowledge logic by means of spatial constraint systems.

We show that scs's as introduced first in [KPPV12] besides capturing the notion of belief, can also be used to derive the epistemic notion of knowledge. In [KPPV12] the authors give a representation not only of the notion of belief but also they introduce what is known as *knowledge constraint system*. The idea there is that we can impose the knowledge axioms over the scs, restricting this way the space functions to satisfy further conditions than the ones required by scs's. Here we shall show that we do not need to impose these axioms over the scs to characterize the notion of knowledge. Instead, we shall show that those axioms can be satisfied by means of a derived operator from the underlying scs.

1.1.3 Distributed Spaces

Distributed and epistemic scenarios can be found in many applications of interest to the computer science community. The range of applications goes from security systems to social networks. In these kind of systems the *epistemic* behaviour plays an important role and arises in a natural way as a result of the interaction among the agents of which they are comprised.

The analysis of programs is one of the most studied topics in computer science. A huge variety of techniques has been created to tackle this issue. Ranging from *model checking* [CGP99, McM03], a well known and widely applicable verification technique, to *bisimulation* [Mil80, Par81, MS92] and its derivations which we can thought of as a more theoretical way to analyze programs. Traditionally, the analysis of programs has focused on the single-agent case. However given the rise of distributed and multi-agent computing, where having groups of agents sharing their local information is typical, the analysis of multi-agent scenarios is attracting the attention of researchers in computer science.

In distributed applications one might be interested in knowing whether the models used to characterize the systems present a (un)desired behaviour. One might also wonder what could happen if the agents in the systems were to communicate the information/knowledge they have. This potential way of communicating is known in the literature of epistemic logic as *distributed knowledge* [FHMV95].

As an example, consider the situation in which a given agent A knows that *every time Pete is wearing his favourite team shirt then he is going* *to the stadium*. Also consider that agent B, who saw Pete in the morning, knows that *Pete is wearing his favourite team shirt*. Therefore if agents A and B were to exchange the knowledge they have, they will conclude that *Pete is going to the stadium* even if individually they do not know it.

In [FHMV95] the authors propose a modal language with modalities for the analysis of a group of agents in a system. The modalities in the language represent situations in which *every* agent in the group knows a fact, they all know that they know it, they all know that they all know that they know it, and so on ad infinitum (known as *common knowledge*) and situations in which a fact is known by *combining* the information owned individually by the agents of a group (*distributed knowledge*).

We shall give an algebraic characterization of the notion of distributed information. We shall define the notion of *distributed space* as an operator aiming to capture the behaviour of when a group of agents has the knowledge of a piece of information distributed among the individual knowledge of its agents. Therefore, given a group of agents G, the *distributed information* of a subgroup $I \subseteq G$ can be intuitively defined as the collective/aggregate information from all its members.

Contributions and Organization

In the following list we shall summarize the main contributions of this dissertation. The list also reflects how this dissertation is organized.

- In Part I we give some preliminaries on constraint systems and modal logics.
- In Part II we study the extrusion problem for scs's. We derive a complete characterization for the existence of right inverses of space functions: The weakest restriction on the elements of the constraint systems (i.e., Kripke Structures) that guarantees the existence of right inverses. Also we give a characterization and derivations of extrusion functions that are normal (and thus they corresponding to normal

inverse modalities). In particular we identify the *greatest* normal extrusion function and a family of *minimal* normal extrusion functions.

- In Part III we focus our attention in the problem of representing knowledge in terms of scs's. We contribute by giving an algebraic representation of the epistemic notion of *knowledge* by means of a derived spatial operation from the underlaying scs.

- In Part IV we go through the notion of distributed information from an algebraic perspective. We provide a canonical representation of the notion of *distributed information* in terms of scs's.

For the convenience of the reader this dissertation includes an index table.

Publications

This manuscript is mainly based on the following articles I have co-authored during my thesis.

- Michell Guzmán, Stefan Haar, Salim Perchy, Camilo Rueda, and Frank D. Valencia. Belief, knowledge, lies and other utterances in an algebra for space and extrusion. *Journal of Logical and Algebraic Methods in Programming, JLAMP*, 86:107âĂŞ133, 2017.

- Michell Guzmán, Frank D. Valencia: On the Expressiveness of Spatial Constraint Systems. *ICLP (Technical Communications)* 2016: 16:1-16:12, 2016.

- Michell Guzmán, Salim Perchy, Camilo Rueda, and Frank D. Valencia. Deriving inverse operators for modal logics. *In Proceedings of the 13th International Colloquium on Theoretical Aspects of Computing, ICTAC 2016*, pages 214âĂŞ232, 2016.

During my PhD, I also have co-authored the following article based on my previous work. The work was not included in this dissertation because it deviates from the main topic of this thesis, i.e., the expressiveness of spatial constraint systems.

- Jaime Arias, Michell Guzmán, Carlos Olarte: A Symbolic Model for Timed Concurrent Constraint Programming. *Electronic Notes on Theoretical Computer Science*, 312: 161-177, 2015.

Recently, we have submitted the following work to the journal of theoretical computer sciences.

- Michell Guzmán, Salim Perchy, Camilo Rueda, and Frank Valencia. Deriving Extrusion on Constraint Systems from Concurrent Constraint Programming Process Calculi. Submitted to the *Journal of Theoretical Computer Science* (TCS), April 2017.

Part I

Constraint Systems and Modal Logics

Two

Introduction : Constraint Systems

In this chapter we introduce the notion of *constraint systems* (cs's) that is central for this dissertation. Constraint systems is one of the main notions of the *concurrent constraint programming* (ccp), a well known formalism in concurrency theory. Most of the material in this chapter was taken from [BDPP95, KPPV12, HPRV15, ORV13]. For more details on constraint systems and ccp we strongly recommend you to see [SRP91, PSSS93].

This chapter is organized as follows. First, we shall start by giving some definitions for *posets*, *maps* and *lattices* which were taken from [Vic96, DP02]. Then we shall move to the basic notion of *constraint systems*. We shall also introduce in this chapter the notions of *spatial constraint systems (scs's)* and *spatial constraint systems with extrusion (scse's)*, which can be seen as extensions of the basic cs's in order to account for *space* and *mobility*, respectively. We shall describe the differences between scs's and its predecessor cs's.

2.1 Posets and Maps

Definition 2.1.1 (Preorder). *A* preorder *is a set P which is equipped with a binary relation $\sqsubseteq$ that is* reflexive *and* transitive. □

2.1.1 Posets

A *poset* or a *partially ordered set* can be seen as a set of elements together with a binary relation stating the order of the elements in the set.

Definition 2.1.2 (Poset). *A* partially ordered set *(poset) is a set P equipped with a binary relation $\sqsubseteq$ satisfying, for every $x, y, z \in P$ the following properties:*

1. *(*reflexivity*) $x \sqsubseteq x$ for all $x \in P$*
2. *(*transitivity*) if $x \sqsubseteq y$ and $y \sqsubseteq z$ then $x \sqsubseteq z$*
3. *(*antisymmetry*) if $x \sqsubseteq y$ and $y \sqsubseteq x$ then $x = y$*

□

Definition 2.1.3 (Directed Set). *Consider the poset $(P, \sqsubseteq)$ and $D \subseteq P$. We say that D is an upward directed set (resp. downward directed set) if for every $a, b \in D$ there exists $c \in D$ such that $a \sqsubseteq c$ and $b \sqsubseteq c$ (resp. $c \sqsubseteq a$ and $c \sqsubseteq b$).* □

2.1.2 Maps

We shall now introduce some formal definitions for maps on posets.

Definition 2.1.4 (Maps). *Let $(P, \sqsubseteq)$ and $(Q, \sqsubseteq)$ be posets. A map $f : P \to Q$ is*

1. ***order-preserving** if $a \sqsubseteq b$ in P then $f(a) \sqsubseteq f(b)$ in Q,*
2. ***order-reflecting** if $f(a) \sqsubseteq f(b)$ in Q then $a \sqsubseteq b$ in P,*
3. ***order-embedding** if it is both, an order-preserving and order-reflecting map,*
4. ***order-isomorphism** if it is an order-embedding and surjective map.*

In addition, we shall say that a map is a self-map *if $P = Q$.* □

2.2 Lattices

2.2.1 Lattices

We define the *meet* (greatest lower bound or glb) of a set as an element which is below every element in that set, formally

Definition 2.2.1 (Meet (glb))**.** *Let P be a poset, $X \subseteq P$ and $y \in P$. Then y is a glb for X iff*

- *(*lower bound*) If $x \in X$ then $y \sqsubseteq x$, and*
- *If z is any other lower bound for X then $z \sqsubseteq y$*

We shall write $y = \bigsqcap X$ whenever y is the glb of X. □

The next proposition states that for every subset of a poset we can have *at most* one meet.

Proposition 2.2.1. *Let P be a poset and $X \subseteq P$. Then X can have at most one glb.*

Now we introduce the definition of *join* (least upper bound or lub) of a set as an element which is above every element in that set, that is

Definition 2.2.2 (Join (lub))**.** *Let P be a poset, $X \subseteq P$ and $y \in P$. Then y is a lub for X iff*

- *(*upper bound*) If $x \in X$ then $y \sqsupseteq x$, and*
- *If z is any other upper bound for X then $z \sqsupseteq y$*

We shall write $y = \bigsqcup X$ whenever y is the lub of X. □

By the duality principle we shall use P^{OP} to denote the same elements in P with the order reversed.

Proposition 2.2.2. *Let P be a poset, $X \subseteq P$ and $y \in P$. Then y is a lub for X iff y is a glb for X in P^{OP}*

Now we formally define when a function between two posets preserves meets.

Definition 2.2.3. *Let P and Q be two posets, and f be a function $f : P \to Q$. Then we say that f preserves* glbs *iff whenever $X \subseteq P$ has a glb y, then $f(y)$ is a glb for $\{f(x) \mid x \in X\}$.* □

Given that posets only can guarantee the *order* of the elements in a given set, then we need to make use of the notion of *lattices* which, in addition to preserve the order, they have *all* finite meets (glbs) and joins (lubs).

Definition 2.2.4 (Lattice)**.** *Consider the poset P. Then P is called*

- *a* lattice *if for all $x, y \in P$ then $x \sqcup y$ and $x \sqcap y$ exists.*
- *a* complete lattice *iff for every subset $X \subseteq P$, we have $\bigsqcap X$ and $\bigsqcup X$.*

□

We shall call a meet (join) preserving function between two lattices a *lattice homomorphism.* An important property on maps that will be very useful through this dissertation is that of being able to preserve all lub (or glb). This property on maps is called *continuity* and can be defined as follows :

Definition 2.2.5 (Continuous Maps)**.** *Consider the lattices L and L', a map between the lattices $f : L \to L'$ and and a set $S \in L$. We say that*

1. *f is* upward-continuous *if $f(\bigsqcup S) = \bigsqcup f(S)$ and S is directed. If S is not directed then f is* join-complete.
2. *f is* downward-continuous *if $f(\bigsqcap S) = \bigsqcap f(S)$ and S is downward directed. If S is not downward directed then f is* meet-complete.

□

2.2.2 Algebraic Lattice

Another important definition that we shall introduce is that of *algebraic lattices.* Intuitively, we say that a lattice is algebraic whenever every element of the lattice can be approximated by the finite elements below it. These elements are knows as *compact elements.*

Definition 2.2.6 (Compact Elements)**.** *Let L be a lattice and $k \in L$. We say that k is* compact *if for every subset $S \subseteq L$*

$$k \sqsubseteq \bigsqcup S \textit{ implies } k \sqsubseteq \bigsqcup S' \text{ for some finite } S' \subseteq S$$

The set of compact elements in L is denoted by $K(L)$

Definition 2.2.7 (Algebraic Lattices)**.** *Let L be a complete lattice. We say that L is an* algebraic *lattice if for each element $x \in L$ we have*

$$c = \bigsqcup \{y \in K(L) \mid y \sqsubseteq x\}$$

where $K(L)$ represents the set of compact elements in L. □

2.2.3 Distributive lattices

Many applications in order theory require lattices in which meets (glb) distributes over the joins (lub). These lattices are know in the literature as *distributive lattices.*

Definition 2.2.8 (Distributive Lattices)**.** *A lattice L is distributive iff for every x, y and $z \in L$ we have that $x \sqcap (y \sqcup z) = (x \sqcap y) \sqcup (x \sqcap z)$.* □

2.3 Frames

In this section we shall talk about *frames* and *Heyting algebras.* We recommend to see [Vic96] for more details and proofs about this two notions.

2.3.1 Frames

We can intuitively define a *frame* as a distributive lattice in which every arbitrary subset has a join and every finite subset has a meet. Formally.

Definition 2.3.1 (Frame). *Let A be a poset, $Y \subseteq A$ and $x \in Y$. We say that A is a* frame *iff*

1. *Every subset has a* join
2. *Every finite subset has a* meet
3. *Binary meets* distribute *over joins:*

$$x \sqcap \bigsqcup Y = \bigsqcup \{x \sqcap y : y \in Y\} \quad (\textbf{\textit{frame distributivity}})$$

□

We shall call a function between two frames preserving arbitrary joins and finite meets a *frame homomorphism.*

2.3.2 Heyting algebras

In this dissertation we shall often mention the notion of *Heyting implication.* We can intuitively define a *Heyting implication* $c \to d$ as the weakest constraint e you should join c with in order to be able to entail d. Heyting implications are the standard implication operation in *Heyting algebras.* Formally.

Definition 2.3.2 (Heyting algebra). *Consider the lattice $A = (Con, \sqsubseteq)$. We say that A is a* Heyting algebra *iff for every $c, d \in Con$ we can build one element $c \to d$ such that*

$$c \to d \stackrel{\texttt{def}}{=} \bigsqcap \{e \in Con \mid c \ \sqcup \ e \sqsupseteq d\}.$$

The element $c \to d$ is known as a Heyting implication*. We shall define a* Heyting negation *as $\sim c \stackrel{\texttt{def}}{=} c \to false$. Recall that false is the greatest element in the constraint system representing the join of possible inconsistent information.* □

We shall call a function f between Heyting algebras preserving finite joins, finite meets and the operation $\to$ (i.e. $f(a \to b) = f(a) \to f(b)$) a *Heyting algebra homomorphism.* Additionally, we shall say that a Heyting algebra is a *complete Heyting algebra (cHa)* iff it is a *complete* lattice.

The following proposition states the relation between frames and cHa.

Proposition 2.3.1. *Consider the lattice A. Then we say that A is a frame iff it is a cHa.*

2.4 Constraint Systems

The concurrent constraint programming model of computation [SRP91] is parametric in a *constraint system* (cs) specifying the structure and interdependencies of the partial information that computational agents can ask of and post in a *shared store*.

Constraint systems can be seen as *complete algebraic lattices* [BDPP95][1]. As mentioned in Section 1.1, the elements of the lattice, the *constraints*, represent (partial) information. A constraint c can be viewed as an *assertion* (or a *proposition*). The lattice order $\sqsubseteq$ is meant to capture entailment/derivation of information: $c \sqsubseteq d$, alternatively written $d \sqsupseteq c$, means that the assertion d represents as much information as c. Thus we may think of $c \sqsubseteq d$ as saying that d *entails* c or that c can be *derived* from d. The *least upper bound (lub)* operator $\sqcup$ represents join of information; $c \sqcup d$, the least element in the underlying lattice above c and d. Thus $c \sqcup d$ can be seen as an assertion stating that both c and d hold. The top element represents the lub of all, possibly inconsistent, information, hence it is referred to as *false*. The bottom element *true* represents the empty information.

Definition 2.4.1 (Constraint Systems [BDPP95])**.** *A constraint system (cs)* $\mathbf{C}$ *is a complete algebraic lattice* $(Con, \sqsubseteq)$*. The elements of Con are called* constraints*. The symbols* $\sqcup$*, true and false will be used to denote the least upper bound (lub) operation, the bottom, and the top element of* $\mathbf{C}$*, respectively.* □

Now we give an example of a constraint system, in which the intuitive idea is to characterize a *propositional logic.*

Example 2.4.1 (Boolean Constraint System [ABP+11])**.** *Let* Φ *be a set of* primitive propositions. *A boolean (or truth) assignment* π *over* Φ *is a total map from* Φ *to the set* $\{0, 1\}$*. We use* $\mathcal{A}(\Phi)$ *to denote the set of all such boolean assignments. We can now define the boolean cs* $\mathbf{B}(\Phi)$ *as*

[1] An alternative syntactic characterization of cs's, akin to Scott information systems, is given in [SRP91].

$(\mathcal{P}(\mathcal{A}(\Phi)), \supseteq)$: *The powerset of assignments ordered by* $\supseteq$. *Thus constraints in Con are sets of assignments,* $\sqsubseteq$ *is* $\supseteq$, *false is* $\emptyset$, *true is* $\mathcal{A}(\Phi)$, *the join operator* $\sqcup$ *is* $\cap$, *and the meet operator* $\sqcap$ *is* $\cup$. *A constraint* c *in* $\mathbf{B}(\Phi)$ *is compact iff* $\mathcal{A}(\Phi) \setminus c$ *is a finite set.* □

We shall use the following notions and notations from order theory.

Notation 2.4.1 (Lattices and Limit Preservation). *Let* $\mathbf{C}$ *be a partially ordered set (poset)* $(Con, \sqsubseteq)$. *We shall use* $\bigsqcup S$ *to denote the least upper bound (lub) (or* supremum *or* join*) of the elements in* S, *and* $\bigsqcap S$ *is the greatest lower bound (glb) (*infimum *or* meet*) of the elements in* S. *We say that* $\mathbf{C}$ *is a* complete lattice *iff each subset of Con has a supremum and a infimum in Con. A non-empty set* $S \subseteq Con$ *is* directed *iff every* finite *subset of* S *has an upper bound bound in* S. *Also* $c \in Con$ *is* compact *iff for any directed subset* D *of Con,* $c \sqsubseteq \bigsqcup D$ *implies* $c \sqsubseteq d$ *for some* $d \in D$. *A complete lattice* $\mathbf{C}$ *is said to be* algebraic *iff for each* $c \in Con$, *the set of compact elements below it forms a directed set and the lub of this directed set is* c. *A self-map on Con is a function* $f : Con \to Con$. *Let* $(Con, \sqsubseteq)$ *be a complete lattice. The self-map* f *on Con* preserves *the supremum of a set* $S \subseteq Con$ *iff* $f(\bigsqcup S) = \bigsqcup\{f(c) \mid c \in S\}$. *The preservation of the infimum of a set is defined analogously. We say* f *preserves finite/infinite suprema iff it preserves the supremum of arbitrary finite/infinite sets. Preservation of finite/infinite infima is defined similarly.*

2.5 Spatial Constraint Systems

The authors in [KPPV12] have extended the basic notion of cs to account for distributed and multi-agent scenarios where agents have their own space in which they can either store their local information or perform their computations. As early introduced in Section 1.1, in scs's each agent i has a *space* function $[\cdot]_i : Con \to Con$ having as domain and co-domain a set of constraints. Recall that constraints can be viewed as assertions. We can then think of $[c]_i$ as an assertion stating that c is a piece of information residing *within a space attributed to agent* i. An alternative *epistemic logic* interpretation of $[c]_i$ is an assertion stating that agent i *believes* c or that c holds within the space of agent i (but it may not hold elsewhere). Following both interpretations we can say that c is *local* to agent i. Similarly, $[[c]_j]_i$ can be seen as a hierarchical spatial specification stating that c holds within the local space the agent i attributes to agent j. Nesting of spaces can be of any depth. We can think of a constraint of the form $[c]_i \sqcup [d]_j$ as an assertion specifying that c and d hold within two *parallel* or *neighbouring* spaces that belong to agents i and j, respectively. From a *computational* (or *concurrency theory*) point of view, we think of $\sqcup$ as parallel composition. Also, as mentioned before, from a logic point of view the join of information $\sqcup$ corresponds to conjunction.

Definition 2.5.1 (Spatial Constraint System [KPPV12]). *An n-agent* spatial constraint system (n-scs) $\mathbf{C}$ *is a cs* $(Con, \sqsubseteq)$ *equipped with n self-maps* $[\cdot]_1, \ldots, [\cdot]_n$ *over its set of constraints Con such that:*

S.1 $[true]_i = true$, *and*

S.2 $[c \sqcup d]_i = [c]_i \sqcup [d]_i$ *for each* $c, d \in Con$.

□

Axiom S.1 requires space functions to be *strict* maps (i.e bottom preserving). Intuitively, it states that having an empty local space amounts to nothing. Axiom S.2 states that the information in a given space can be distributed. Notice that requiring S.1 and S.2 is equivalent to requiring that each $[\cdot]_i$ preserves *finite suprema* (lub).

Remark 2.5.1 (Monotone Spaces)**.** *Notice that S.2 implies that space function are* order-preserving *(or monotone): i.e., if* $c \sqsubseteq d$ *then* $[c]_i \sqsubseteq [d]_i$. *Intuitively, if* c *can be derived from* d *then any agent* i *should be able to derive* c *from* d *within its own space.*

Proof. Assume $c \sqsubseteq d$, thus $d = c \sqcup d$. Then $[d]_i = [c \sqcup d]_i$. Using S.2 we have $[d]_i = [c]_i \sqcup [d]_i$, hence $[c]_i \sqsubseteq [d]_i$. ☐

We already know that we can have nested spaces which can be of any depth. Then, with the idea of having *globally* available certain information in any nesting of spaces we shall introduce the notion of *global information.*

Definition 2.5.2 (Global Information)**.** *Consider* $\mathbf{C}$ *to be an* n*-scs with space functions* $[\cdot]_1, \ldots, [\cdot]_n$ *and* G *as a non-empty subset of* $\{1, \ldots, n\}$. Group-spaces $[\cdot]_G$ *and* global *information* $[\![\cdot]\!]_G$ *of* G *in* $\mathbf{C}$ *are defined as:*

$$[c]_G \stackrel{\text{def}}{=} \bigsqcup_{i \in G} [c]_i \quad \textit{and} \quad [\![c]\!]_G \stackrel{\text{def}}{=} \bigsqcup_{j=0}^{\infty} [c]_G^j \tag{2.5.1}$$

where $[c]_G^0 \stackrel{\text{def}}{=} c$ *and* $[c]_G^{k+1} \stackrel{\text{def}}{=} [[c]_G^k]_G$. ☐

The constraint $[c]_G$ intuitively means that c holds in the spaces of agents in G. Given that the constraint $[\![c]\!]_G$ entails $[[\ldots[c]_{i_m}\ldots]_{i_2}]_{i_1}$ for any $i_1, i_2, \ldots, i_m \in G$, it satisfies the intuition that the constraint c holds *globally* w.r.t. G. This means that c holds in each nested space involving the agents in G. A particular case is given if we take G to be the set of all agents. In this case, $[\![c]\!]_G$ means that the information c holds *everywhere.*

From a logic (epistemic logic) point of view, the global information operator can be seen as the well known notion of *common-knownledge* [FHMV95]. Thus $[\![c]\!]_G$ can be thought of agents in G having common-knowledge of the information c.

2.6 Spatial Constraint Systems with Extrusion

In the last section we saw how constraint systems were extended in order to tackle distributed and multi-agent scenarios where agents have their own spaces in which they can store their information and run their programs. But the information lying inside agents' spaces was *static*, i.e. the exchange of information among the agents in the system was not possible. We shall call that exchange of information (data and programs) as *mobility*. The authors in [HPRV15] have extended scs's in order to characterize the mobile behaviour in these algebraic structures. This extension was called *spatial constraint systems with extrusion* (scse).

The basic idea behind scse's is to equip each agent i with an *extrusion* function $\uparrow_i : Con \to Con$. Intuitively, within a space context $[\cdot]_i$, the assertion $\uparrow_i c$ specifies that c must be posted outside of (or extruded from) agent i's space. This is captured by requiring the *extrusion* functions to satisfy the extrusion axiom, i.e. $[\ \uparrow_i c\ ]_i = c$. As introduced in Section 1.1, we view *extrusion/utterance* as the right inverse of *space/belief* (and thus space/belief as the left inverse of extrusion/utterance).

Definition 2.6.1 (Extrusion). *Given an n-scs $(Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$, we say that $\uparrow_i$ is an extrusion function for the space $[\cdot]_i$ iff $\uparrow_i$ is a right inverse of $[\cdot]_i$, i.e., iff*

$$[\ \uparrow_i c\ ]_i = c. \tag{E.1}$$

□

From the above definitions it follows that $[c \sqcup \uparrow_i d]_i = [c]_i \sqcup d$. From a spatial point of view, agent i *extrudes* d from its local space. From an epistemic view this can be seen as an agent i that believes c and *utters* d to the outside world. If d is inconsistent with c, i.e., $c \sqcup d = \mathit{false}$, we can see the utterance as an intentional *lie* by agent i: The agent i utters an assertion inconsistent with her own beliefs.

Example 2.6.1. *Let* $e = [c \sqcup \uparrow_i [a]_j\ ]_i \sqcup [d]_j$. *The constraint* e *specifies that agent* i *has* c *and wishes to transmit, via extrusion,* a *addressed to agent* j. *Agent* j *has* d *in their space. Indeed, with the help of E.1 and S.2, we can derive* $e \sqsupseteq [d \sqcup a]_j$ *thus stating that* e *entails that* a *will be in the space of* j. □

We shall now introduce a very important property of complete lattices and space functions. Consider a space function $[\cdot]_i$ of a given spatial constraint system $\mathcal{C}$. Given that $[\cdot]_i$ satisfy axioms S.1, S.2, continuity and that $\mathcal{C}$ is a complete lattice then it is easy to see that $[\cdot]_i$ preserves *arbitrary* lubs. A space function that preserves the supremum of any arbitrary set of Con is said to be *join-complete*. If the space function preserves the infimum of any arbitrary set of Con is said to be *meet-complete*.

Proposition 2.6.1. *Let* $[\cdot]_i$ *be a space function of a scs* $\mathcal{C}$. *If* $[\cdot]_i$ *is continuous then* $[\cdot]_i$ *is join-complete.*

Proof. The proof follows from the fact that every function from a complete lattice preserves arbitrary suprema iff it preserves both directed suprema (continuous) and finite suprema (binary lub) [GHK+03]. □

Three

Introduction : Modal Logics

In this chapter we shall recall the notion of *modal logic.* Modal logics extend propositional logics with modal operators which can be used to represent different scenarios such as *temporal* and *epistemic* (depending on the interpretation we give to them). Modal logics are mainly used in computer science in the *specification* of programs, which is one of the main steps in the theory of formal verification. The notions introduced here will be relevant for this dissertation because the main results of this thesis have modal logics as elemental notions. Most of the material in this chapter is based on [BDRV01, BvBW06]. This chapter is organized as follows. First, we shall introduce the basics of modal logics. We shall start by defining how we can build formulae in modal logics (the *language*). Then we shall describe the way in which modal formulas are interpreted (the *semantics*). Finally, we shall define the notion of *normality* in modal logics. For more details or further references see [BDRV01, BvBW06, Gol03].

3.1 Modal Logics

In order to be able to deal with modal logics we would need to define how to *build* modal formulae and how to *interpret* them. In this section we shall give the basic definitions for modal languages that will help us to show how to interpret modal formulae.

3.1.1 Modal Language

A modal language $\mathcal{L}_n(\Phi)$ can be seen as the language that results by adding modalities m to the language formed by sets of primitive propositions and logical connectives such as *or, and, negation* and *implication*. Formally, a modal language can be defined as follows.

Definition 3.1.1 (Modal Language)**.** *Let Φ be a set of primitive propositions. The modal language $\mathcal{L}_n(\Phi)$ is given by the following grammar:*

$$\phi, \psi, \ldots \ := \ p \mid \phi \wedge \psi \mid \neg\phi \mid \Box_i \phi$$

where $p \in \Phi$ and $i \in \{1, \ldots, n\}$. We shall use the abbreviations $\phi \vee \psi$ for $\neg(\neg\phi \wedge \neg\psi)$, $\phi \Rightarrow \psi$ for $\neg\phi \vee \psi$, $\phi \Leftrightarrow \psi$ for $(\phi \Rightarrow \psi) \wedge (\psi \Rightarrow \phi)$, the constant false ff for $p \wedge \neg p$, and the constant tt for $\neg$ff. □

3.1.2 Kripke Model

In Definition 3.1.1 we have formally defined the way in which modal formulae should be built. Now we shall introduce some basic definitions that will allow us to give semantics to modal formulae.

Kripke structures (KS) are a fundamental mathematical tool in logic and computer science. They can be seen as transition systems and they are used to give semantics to modal logics. A KS M provides a relational structure with a set of states and one or more accessibility relations $\xrightarrow{i}_M$ between them: $s \xrightarrow{i}_M t$ can be seen as a transition, labelled with i, from s to t in M.

Definition 3.1.2 (Kripke Structures)**.** *An n-agent Kripke Structure (KS) M over a set of atomic propositions Φ is a tuple $(S, \pi, \mathcal{R}_1, \ldots, \mathcal{R}_n)$ where*

- *S is a nonempty set of states,*
- *$\pi : S \to (\Phi \to \{0, 1\})$ is an interpretation associating with each state a truth assignment to the primitive propositions in Φ, and*
- *$\mathcal{R}_i$ is a binary relation on S.*

A pointed KS *is a pair* (M, s) *where* M *is a KS and* s*, called the* actual world*, is a state of* M*.* □

Finally, with the following definition we introduce how to give semantics to modal logics by using Kripke structures.

Definition 3.1.3. *Let* $M = (S, \pi, R_1, \ldots, R_n)$ *be a Kripke structure and* ϕ *be a modal formula. Then we define the relation* $\models$ *inductively, over* M, ϕ *and a given state* $s \in S$ *as follows:*

$$
\begin{array}{lr}
(M,s) \models p & \textit{iff} \;\; \pi_M(s)(p) = 1 \\
(M,s) \models \phi \wedge \psi & \textit{iff} \;\; (M,s) \models \phi \textit{ and } (M,s) \models \psi \\
(M,s) \models \neg\phi & \textit{iff} \;\; (M,s) \not\models \phi \\
(M,s) \models \Box_i\phi & \textit{iff} \;\; (M,t) \models \phi \textit{ for all } t, \textit{ s.t. } (s,t) \in R_i
\end{array}
$$

□

From now on we shall use the following notation.

Notation 3.1.1. *Each* $\mathcal{R}_i$ *is referred to as the* accessibility *relation for agent* i*. We shall use* $\xrightarrow{i}_M$ *to refer to the accessibility relation of agent* i *in* M*. We write* $s \xrightarrow{i}_M t$ *to denote* $(s,t) \in \mathcal{R}_i$*. We use* $\blacktriangleright_i(M,s) = \{(M,t) \mid s \xrightarrow{i}_M t\}$ *to denote the pointed KS reachable from the pointed KS* (M,s)*. Recall that the interpretation function* π *tells us what primitive propositions are true at a given world:* p *holds at state* s *iff* $\pi(s)(p) = 1$*. We shall use* S_M *and* π_M *to denote the set of states and interpretation function of* M*.*

3.2 Normal Modal Logics

We have already introduced how to build modal formulae and how to interpret them over Kripke structures. In this section we shall introduce the notion of normality in modal logics. We start by giving some basic definitions before we formally define normal modal logics.

Two important properties when dealing with modal logics are those of *modus ponens* and *generalization*. These two properties are the ones characterizing the type of modal logics we shall work with in this thesis. The modal logics satisfying modus ponens and generalization are called *normal modal logics*. The property of modus ponens can be formally defined as follows.

Definition 3.2.1 (Modus Ponens)**.** *We say that a set of formulas S is closed under* modus ponens *if whenever $\phi \in S$ and $\phi \Rightarrow \psi \in S$ then $\psi \in S$.* $\square$

The notion of generalization can then be defined as follows.

Definition 3.2.2 (Generalization)**.** *We say that a given set of formulas S is closed under* generalization *if whenever $\phi \in S$ then $\Box_i \phi \in S$, for all $i \in I$, where I is the set of agents.* $\square$

3.2.1 Normal Modal Logics

In modal logics one is interested in *normal modal operators*. We already know that modal logic formulae are those of propositional logic extended with modal operators. Roughly speaking, a modal logic operator m is called *normal* if and only if the following conditions are met :

1. The formula $\mathrm{m}(\phi)$ is a theorem (i.e., true in all models for the underlying modal language) whenever the formula ϕ is a theorem, and
2. the implication formula $\mathrm{m}(\phi \Rightarrow \psi) \Rightarrow (\mathrm{m}(\phi) \Rightarrow \mathrm{m}(\psi))$ is a theorem.

Thus, as mentioned above, a normal modal logic is a modal logic in which the modalities satisfy the modus ponens and the generalization conditions. Formally, it can be defined as follows.

Definition 3.2.3 (Normal Modal Logics)**.** *Consider the modal language $\mathcal{L}_n(\Phi)$ formed by the set of modal formulas Φ. We say that $\mathcal{L}_n(\Phi)$ is a* normal modal logic *if it contains:*

- *all the tautologies from propositional logic,*
- *the implication formula* $\mathrm{m}(\phi \Rightarrow \psi) \Rightarrow (\mathrm{m}(\phi) \Rightarrow \mathrm{m}(\psi))$ *for all modalities $m \in \mathcal{L}_n(\Phi)$ with $\phi, \psi \in \Phi$.*

- *and it is* closed *under modus ponens and generalization.*

□

Part II

The Extrusion Problem

Four

Introduction

The main focus in this part of the dissertation is the study of the *extrusion problem* for a meaningful family of scs's.

Spatial constraint systems with extrusion were proposed as an alternative to model spatial and mobile behaviour of multi-agent systems. As mentioned in Section 1.1.1 in scse space functions $[\cdot]_i$ were equipped with right inverses defined as $\uparrow_i : Con \to Con$ satisfying that $[\uparrow_i c]_i = c$. We already know that surjectivity of $[\cdot]_i$ is the property that guarantees the existence of these right inverses. Nevertheless this property does not say anything about how to build these functions (i.e. it does not provide provide explicit constructions for them).

In this part of the dissertation we shall study the *extrusion problem* for a meaningful family of scs's called *Kripke spatial constraint systems* (Kripke scs). Kripke scs are commonly used as semantic structures for modal logics. The elements in Kripke scs are *Kripke structures*(ks's) which can be seen as transition systems that endow with some additional structure on their states. The main contribution in this chapter is the characterization and derivation of extrusion functions that are normal. The importance of normal functions is that they correspond to normal modal logics which are widely used in computer science. We shall therefore provide the minimal conditions for which extrusion functions exist in Kripke scs as well as a taxonomy of normal right inverses over these constraint systems.

Our contributions in this part can be summarised as follows:

1. We give an algebraic characterization of the notion of *normality* in modal logic by building upon work on Geometric Logic [Vic96] and show that this abstract notion corresponds exactly to preservation of the join operation.

2. We show that extrusion functions (i.e., right inverses of space) of Kripke scs's correspond to (right) inverse modalities in modal logic. Inverses modalities arise in the form of past operators in temporal logic [RS97], utterances in epistemic logic [HPRV15], backward moves in modal logic for concurrency [PU11], among others.

3. We derive a complete characterization for the existence of right inverses of space functions: The weakest restriction on the elements of the constraint systems (i.e., KS's) that guarantees the existence of right inverses.

4. We give a characterization and derivations of extrusion functions that are normal (and thus they corresponding to normal inverse modalities). In particular we identify the *greatest* normal extrusion function and a family of *minimal* normal extrusion functions.

5. Finally we discuss the application of our results in the context of specific modal systems and related concepts such as the minimal modal logic K_n [FHMV95], a modal logic of linear-time [PM92], and bisimilarity.

Organization.

This part of the dissertation is structured as follows. In Section 4.1 we recall the notions of constraint system (cs) and spatial cs (scs). In Section 4.2 we give an algebraic characterization of the notion of normality in modal logic and show that this abstract notion is equivalent to preservation of finite joins. In Section 4.3 we identify necessary and sufficient conditions for the existence of right inverses of space functions. In Section 4.4 we turn our attention to the derivation and classification of normal right inverses. Finally, in Section 4.5 we discuss and apply our results in the context of the modal system K_n, LTL, and bisimilarity.

4.1 Background: Spatial Constraint Systems

The formal definitions for the notions of basic constraint systems and spatial constraint systems (with and without extrusion) conforming the background for this section were already introduced in Sections 2.4, 2.5 and 2.6. We shall now present the intuition behind the extrusion problem.

The Extrusion/Right Inverse Problem A legitimate question is: Given space $[\cdot]_i$ can we derive an extrusion function $\uparrow_i$ for it ? From set theory we know that there is an extrusion function (i.e., a right inverse) $\uparrow_i$ for $[\cdot]_i$ iff $[\cdot]_i$ is *surjective*. Recall that the *pre-image* of $y \in Y$ under $f : X \to Y$ is the set $f^{-1}(y) = \{x \in X \mid y = f(x)\}$. Thus the extrusion $\uparrow_i$ can be defined as a function, called *choice* function, that maps each element c to some element from the pre-image of c under $[\cdot]_i$.

The existence of the above-mentioned choice function assumes the Axiom of Choice. The next proposition from [HPRV15] gives some constructive extrusion functions. It also identifies a distinctive property of space functions for which a right inverse exists.

Proposition 4.1.1. *Let $[\cdot]_i$ be a space function of scs. Then*

1. *If $[false]_i \neq false$ then $[\cdot]_i$ does not have any right inverse.*
2. *If $[\cdot]_i$ is surjective and preserves arbitrary suprema then*
$$\uparrow_i : c \mapsto \bigsqcup [c]_i^{-1}$$
is a right inverse of $[\cdot]_i$ and preserve arbitrary infima.
3. *If $[\cdot]_i$ is surjective and preserves arbitrary infima then*
$$\uparrow_i : c \mapsto \bigsqcap [c]_i^{-1}$$
is a right inverse of $[\cdot]_i$ and preserve arbitrary suprema.

We have presented spatial constraint systems as algebraic structures for spatial and epistemic behaviour as that was their intended meaning. Nevertheless, we shall see that they can also provide an algebraic structure to reason about Kripke models with applications to modal logics.

In Section 4.3 we shall study the existence, constructions and properties of right inverses for a meaningful family of scs's; the Kripke scs's. The importance of such a study is the connections we shall establish between right inverses and reverse modalities which are present in temporal, epistemic and other modal logics. Property (1) in Proposition 4.1.1 can be used as a test for the existence of a right inverse. The space functions of Kripke scs's preserve arbitrary suprema, thus Property (2) will be useful. They do not preserve in general arbitrary (or even finite) infima so we will not apply Property (3).

It is worth noticing that the derived extrusion $\uparrow_i$ in Property (3) preserves arbitrary suprema. This implies that $\uparrow_i$ is *normal* in a sense we shall make precise next. Normal self-maps give an abstract characterization of normal modal operators, a fundamental concept in modal logic. We will be therefore interested in deriving normal inverses.

4.2 Constraint Frames and Normal Self-Maps

Spatial constraint systems are algebraic structures for spatial and mobile behavior. By building upon ideas from Geometric Logic and Heyting Algebras [Vic96] we can also make them suitable as semantics structures for modal logic. In this section we give an algebraic characterization of the concept of normal modality as those maps preserving finite suprema.

In Definition 2.3.2 we have defined a general form of implication by adapting the corresponding notion from Heyting Algebras to constraint systems. Recall that intuitively, a *Heyting implication* $c \rightarrow d$ in our setting corresponds to the *weakest constraint* one needs to join c with to derive d: The greatest lower bound $\bigsqcap\{e \mid e \sqcup c \sqsupseteq d\}$. Similarly, the negation of a constraint c, written $\sim c$, can be seen as the *weakest constraint inconsistent* with c, i.e., the greatest lower bound $\bigsqcap\{e \mid e \sqcup c \sqsupseteq \mathit{false}\} = c \rightarrow \mathit{false}$.

Definition 4.2.1 (Constraint Frames)**.** *A constraint system* $(Con, \sqsubseteq)$ *is said to be a* constraint frame *iff its joins distribute over arbitrary meets: For every* $c \in Con$ *and* $S \subseteq Con$ *we have* $c \sqcup \bigsqcap S = \bigsqcap\{c \sqcup e \mid e \in S\}$*.* □

The following basic properties of Heyting implication are immediate consequences of the above definitions and Definition 2.3.2.

Proposition 4.2.1. *Let* $(Con, \sqsubseteq)$ *be a constraint frame. For every* $c, d, e \in Con$*:*

1. $c \sqcup (c \to d) = c \sqcup d$
2. $c \sqsupseteq (d \to e)$ *iff* $c \sqcup d \sqsupseteq e$
3. $c \to d = true$ *iff* $c \sqsupseteq d$

Proof. The proof proceeds as follows:

1. $c \sqcup (c \to d) = c \sqcup d$

   This is proved in Lemma 1 (Modus Ponens) of [HPRV15].

2. $c \sqsupseteq (d \to e)$ iff $c \sqcup d \sqsupseteq e$

   - Assume $c \sqsupseteq (d \to e)$. Then we have $c \sqcup d \sqsupseteq (d \to e) \sqcup d$. Therefore, from (1) (modus ponens) we obtain $c \sqcup d \sqsupseteq d \sqcup e$ and $c \sqcup d \sqsupseteq e$ as wanted.
   - Suppose $c \sqcup d \sqsupseteq e$. Because $d \to e = \bigsqcap S$ where $S = \{a \mid a \sqcup d \sqsupseteq e\}$, and $c \in S$, then $c \sqsupseteq (d \to e)$.

3. $c \to d = true$ iff $c \sqsupseteq d$. By (2), assume $e \sqsupseteq (c \to d)$ iff $e \sqcup c \sqsupseteq d$. Let us take $e = true$. Then, $true \sqsupseteq (c \to d)$ iff $true \sqcup c \sqsupseteq d$. Therefore, $true = (c \to d)$ iff $c \sqsupseteq d$ as wanted.

□

From a computational point of view, we can think of $c \to d$ as a process that triggers d if c is present in the space the process is placed. We illustrate this next.

Example 4.2.1. *A simple example can be obtained by taking $d = a \to b$ in Example 2.6.1, i.e., $e = [c \sqcup \uparrow_i [a]_j\ ]_i \sqcup [a \to b]_j$. We can use Proposition 4.2.1, S.2, and E.1 to obtain $e \sqsupseteq [a \sqcup b]_j$. Thus i can send a to j and cause b to be triggered in the space of j.*

A more complex example involves extrusion from i to j and back from j to i by letting a to be the conditional $d \to \uparrow_j [d]_i$ in the constraint $e = [c \sqcup \uparrow_i [a]_j\ ]_i \sqcup [d]_j$ defined in Example 2.6.1. Intuitively the agent i sends a conditional process a to the space of j. Once in this space, d is entailed and thus d is sent to the space of i via extrusion. Indeed one can verify that $e \sqsupseteq [a \sqcup d]_i$. □

4.2.1 Normal Self-Maps

In computer sciences one is often interested in *normal modal* operators. In Section 3.2.1 we have formally introduced normal modal logics. Since constraints can be viewed as logic assertions, we can think of modal operators as self-maps on constraints. Thus, using Heyting implication, we can express the normality condition in constraint frames as follows.

Definition 4.2.2 (Normal Maps)**.** *Let $(Con, \sqsubseteq)$ be a constraint frame. A self-map m on Con is said to be* normal *if*

1. *$m(true) = true$ and*
2. *$m(c \to d) \to (m(c) \to m(d)) = true$ for each $c, d \in Con$.*

□

Recall that Condition (1) in Definition 4.2.2 represents Condition (1) in Section 3.2.1 because we know that if $m(\phi)$ is a theorem then it must be *true* in all models for the underlying modal languages. We now prove that the normality requirement is equivalent to the requirement of preserving finite suprema. The next theorem states that Condition (2) in Definition 4.2.2 is equivalent to the seemingly simpler condition: $m(c \sqcup d) = m(c) \sqcup m(d)$.

Theorem 4.2.1 (Normality & Finite Suprema)**.** *Let* **C** *be a constraint frame* $(Con, \sqsubseteq)$ *and let* f *be a self-map on* Con*. Then* f *is normal if and only if* f *preserves finite suprema.*

Proof. It suffices to show that for any bottom preserving self-map f, $\forall c, d \in Con : f(c \to d) \to (f(c) \to f(d)) = true$ iff $\forall c, d \in Con : f(c \sqcup d) = f(c) \sqcup f(d)$. (Both conditions require f to be bottom preserving, i.e., $f(true) = true$, and preservation of non-empty finite suprema is equivalent to the preservation of binary suprema.)

- Assume that $\forall c, d \in Con : f(c \to d) \to (f(c) \to f(d)) = true$. Take two arbitrary $c, d \in Con$. We first prove $f(c \sqcup d) \sqsupseteq f(c) \sqcup f(d)$. From the assumption and Proposition 4.2.1(3) we obtain

$$f((c \sqcup d) \to d) \sqsupseteq f(c \sqcup d) \to f(d). \qquad (4.2.1)$$

From Proposition 4.2.1(3) $(c \sqcup d) \to d = true$. Since $f(true) = true$ we have $f((c \sqcup d) \to d) = true$. We must then have, from Equation 4.2.1, $f(c \sqcup d) \to f(d) = true$ as well. Using Proposition 4.2.1(3) we obtain $f(c \sqcup d) \sqsupseteq f(d)$. In a similar fashion, by exchanging c and d in Equation 4.2.1, we can obtain $f(d \sqcup c) \sqsupseteq f(c)$. We can then conclude $f(c \sqcup d) \sqsupseteq f(c) \sqcup f(d)$ as wanted.

We now prove $f(c) \sqcup f(d) \sqsupseteq f(c \sqcup d)$. From the assumption and Proposition 4.2.1(3) we have

$$f(c \to (d \to c \sqcup d)) \sqsupseteq f(c) \to f(d \to c \sqcup d). \qquad (4.2.2)$$

Using Proposition 4.2.1 one can verify that $c \to (d \to c \sqcup d) = true$. Since $f(true) = true$ then $f(c \to (d \to c \sqcup d)) = true$. From Equation 4.2.2, we must then have $f(c) \to f(d \to c \sqcup d) = true$ and by using Proposition 4.2.1(3) we conclude $f(c) \sqsupseteq f(d \to c \sqcup d)$. From the assumption and Proposition 4.2.1(3) $f(d \to c \sqcup d) \sqsupseteq f(d) \to f(c \sqcup d)$. We then have $f(c) \sqsupseteq f(d \to c \sqcup d) \sqsupseteq f(d) \to f(c \sqcup d)$. Thus $f(c) \sqsupseteq f(d) \to f(c \sqcup d)$ and then using Proposition 4.2.1(2) we obtain $f(c) \sqcup f(d) \sqsupseteq f(c \sqcup d)$ as wanted.

- Assume that $\forall c, d \in \mathit{Con} : f(c \sqcup d) = f(c) \sqcup f(d)$. Take two arbitrary $c, d \in \mathit{Con}$. We shall prove $f(c \to d) \to (f(c) \to f(d)) = \mathit{true}$. From Proposition 4.2.1(2-3) it suffices to prove $f(c \to d) \sqcup f(c) \sqsupseteq f(d)$. Using the assumption and Proposition 4.2.1(1) we obtain $f(c \to d) \sqcup f(c) = f(c \sqcup (c \to d)) = f(c \sqcup d) = f(c) \sqcup f(d) \sqsupseteq f(d)$ as wanted.

□

It then follows from the above theorem that space functions from constraint frames are indeed normal self-maps, since they preserve finite suprema. Another immediate consequence of the above theorem is that every normal self-map is also monotone.

Corollary 4.2.1. *Let* **C** *be a constraint frame* $(\mathit{Con}, \sqsubseteq)$ *and let* f *be a normal self-map on Con. If* $c \sqsubseteq d$ *then* $f(c) \sqsubseteq f(d)$.

In this section we have characterized the notion of normal self-maps as those that preserve finite joins. This characterization will be useful in the next section when we turn our attention to normal right-inverse self-maps.

4.3 Extrusion Problem for Kripke Constraint Systems

In this section will study the extrusion/right inverse problem for a meaningful family of spatial constraint systems (scs's); the Kripke scs. In particular we shall derive and give a *complete* characterization of *normal* extrusion functions as well as identify the *weakest* condition on the elements of the scs under which extrusion functions may exist. To illustrate the importance of this study it is convenient to give some intuition first.

Kripke structures (KS) can be seen as transition systems and they are often used to give semantics to modal logics. A KS M provides a relational structure with a set of states and one or more accessibility relations $\xrightarrow{i}_M$ between them: $s \xrightarrow{i}_M t$ can be seen as a transition, labelled with i, from s to t in M. Broadly speaking, a model-based Kripke semantics equates each modal formula ϕ to a certain set $[\![\phi]\!]$ of pairs (M, s), called pointed KS's, where s is a state of the KS M. In particular, in modal logics with one or more modal (box) operators $\Box_i$, the formula $\Box_i\phi$ is equated to $[\![\Box_i\phi]\!] = \{(M, s) \mid \forall t : s \xrightarrow{i}_M t, (M, t) \in [\![\phi]\!]\}$.

Analogously, in a Kripke scs each constraint c is equated to a set of pairs (M, s) of pointed KS. Furthermore, for each space $[\cdot]_i$ we have $[c]_i = \{(M, s) \mid \forall t : s \xrightarrow{i}_M t, (M, t) \in c\}$. This means that formulae can be interpreted as constraints and in particular $\Box_i$ can be interpreted by $[\cdot]_i$ as $[\![\Box_i\phi]\!] = [\ [\![\phi]\!]\ ]_i$.

Inverse modalities $\Box_i^{-1}$, also known as reverse modalities, are used in many modal logics. In tense logics they represent past operators [Pri67] and in epistemic logic they represent utterances [HPRV15]. The basic property of a (right) inverse modality is given by the axiom $\Box_i(\Box_i^{-1}\phi) \Leftrightarrow \phi$. In fact, given a modal logic one may wish to see if it can be extended with reverse modalities (e.g., is there a reverse modality for the *always* operator of temporal logic?).

Notice that if we have an extrusion function $\uparrow_i$ for $[\cdot]_i$ we can provide the semantics for inverse modalities $\Box_i^{-1}$ by letting $[\![\Box_i^{-1}\phi]\!] = \uparrow_i(\ [\![\phi]\!]\ )$ (thus $[\![\Box_i(\Box_i^{-1}\phi)]\!] = [\![\phi]\!]$). Then deriving extrusion functions and establishing the weakest conditions under which they exists is a relevant issue. Furthermore, the algebraic structure of Kripke scs may help us establish desirable properties of the reverse modality such as that of being normal (Definition 4.2.2).

4.3.1 KS and Kripke SCS

In Section 3.1 we have formally defined Kripke structures and gave some useful notation that we shall use through the whole manuscript.

We now define the Kripke scs w.r.t. a set $\mathcal{S}_n(\Phi)$ of pointed KS.

Definition 4.3.1 (Kripke Spatial Constraint Systems [KPPV12])**.** *Let $\mathcal{S}_n(\Phi)$ be a (non-empty) set of n-agent Kripke structures over a set of primitive propositions Φ. Let Δ be the set of all pointed Kripke structures (M, s) such that $M \in \mathcal{S}_n(\Phi)$.*

A Kripke n-scs for $\mathcal{S}_n(\Phi)$ *is a scs* $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *where* $Con = \mathcal{P}(\Delta)$*, for every* $c, d \in Con : c \sqsubseteq d$ *iff* $d \subseteq c$*, and*

$$[\ c\ ]_i \stackrel{\text{def}}{=} \{(M, s) \in \Delta \mid \blacktriangleright_i(M, s) \subseteq c\} \tag{4.3.1}$$

□

In the Kripke-based semantics of modal logic (Section 4.5) every pointed KS is a model of the constant *true* and no pointed KS is a model of the constant *false*. This is consistent with the reversed inclusion $\sqsubseteq = \supseteq$ order of Kripke constraint systems. We clarify this in the following remark.

Remark 4.3.1. *The structure* $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *is a complete algebraic lattice given by a powerset ordered by reversed inclusion* $\supseteq$*. The join* $\sqcup$ *is set intersection, the meet* $\sqcap$ *is set union, the top element false is the empty set* $\emptyset$*, and bottom true is the set* Δ *of all pointed Kripke structures* (M, s) *with* $M \in \mathcal{S}_n(\Phi)$*. Notice that* $\mathbf{K}(\mathcal{S}_n(\Phi))$ *is a frame since meets are unions and joins are intersections so the distributive requirement is satisfied. Furthermore, each* $[\cdot]_i$ *preserves arbitrary suprema (intersection) and thus, from Theorem 4.2.1 it is a normal self-map.*

Proposition 4.3.1. *[GHP+17] Let* $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *as in Definition 4.3.1.*

1. $\mathbf{K}(\mathcal{S}_n(\Phi))$ *is an n-agent spatial constraint frame.*
2. *Each* $[\cdot]_i$ *preserves arbitrary suprema.*

4.3.2 Complete Characterization of the Existence of Right Inverses

We now address the question of whether a given Kripke constraint system can be extended with extrusion functions. We shall identify a sufficient and necessary condition on accessibility relations for the existence of an extrusion function $\uparrow_i$ given the space $[\cdot]_i$. We also give explicit right inverse constructions.

Notation 4.3.1. *For notational convenience, we take the set Φ of primitive propositions and n to be fixed from now on and omit them from the notation. E.g., we write $\mathcal{M}$ instead of $\mathcal{M}_n(\Phi)$.*

The following notions play a key role in our complete characterization, in terms of classes of KS, of the existence of right inverses for Kripke space functions.

Definition 4.3.2 (Determinacy)**.** *Let S and $\mathcal{R}$ be the set of states and an accessibility relation of a KS M, respectively. Given $s, t \in S$, we say that s* determines *t w.r.t. $\mathcal{R}$ if $(s, t) \in \mathcal{R}$. We say that s* uniquely determines *t w.r.t. $\mathcal{R}$ if s is the* only state *in S that determines t w.r.t. $\mathcal{R}$. A state $s \in S$ is said to be* determinant *w.r.t. $\mathcal{R}$ if it uniquely determines some state in S w.r.t. $\mathcal{R}$. Furthermore, $\mathcal{R}$ is* determinant-complete *if every state in S is determinant w.r.t. $\mathcal{R}$.*

(We shall often omit "w.r.t. $\mathcal{R}$" when no confusion can arise.) □

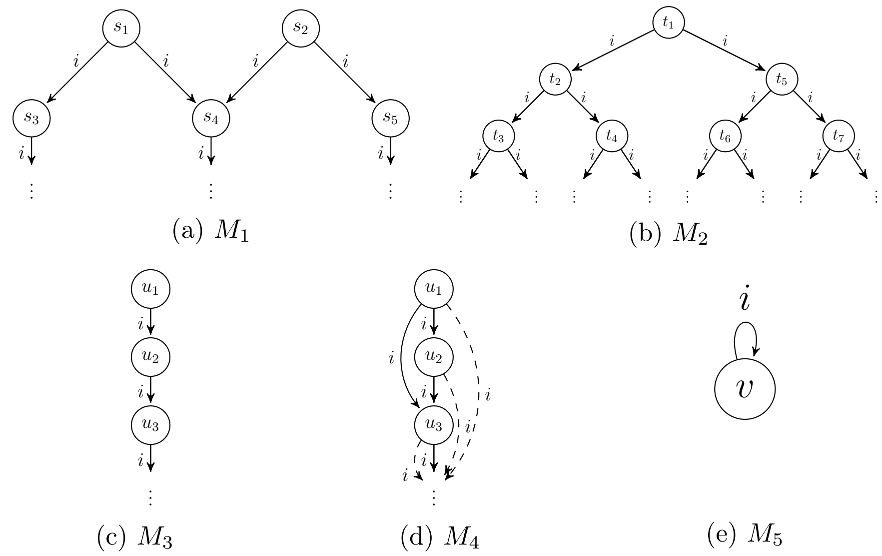


Figure 4.1 – Accessibility relations for an agent i. In each sub-figure we omit the corresponding KS M_k.

Let us have a look at some the following examples of determinacy.

Example 4.3.1. *Figure 4.1 illustrates some typical determinant-complete accessibility relations for agent i. Notice that any determinant-complete relation $\xrightarrow{i}_M$ is necessarily* serial *(or left-total): i.e., For every $s \in S_M$, there exists $t \in S_M$ such that $s \xrightarrow{i}_M t$. Tree-like accessibility relations where all paths are infinite are determinant-complete (Figure 4.1.(b) and Figure 4.1.(c)). Also some non-tree like structures such as Figure 4.1.(a) and Figure 4.1.(e). Figure 4.1.(d) shows a non determinate-complete accessibility relation by taking the transitive closure of Figure 4.1.(c).* □

The following proposition gives an alternative definition of determinant states. First, we need the following notation.

Notation 4.3.2. *Recall that $\blacktriangleright_i(M,s) = \{(M,t) \mid s \xrightarrow{i}_M t\}$ where $\xrightarrow{i}_M$ denotes the accessibility relation of agent i in the KS M. We extend this definition to sets of states as follows $\blacktriangleright_i(M,S) = \bigcup_{s\in S} \blacktriangleright_i(M,s)$. Furthermore, we shall write $s \overset{i}{\twoheadrightarrow}_M t$ to mean that s* uniquely determines *t w.r.t. $\xrightarrow{i}_M$. Analogously, we define $\triangleright_i(M,s) \stackrel{\text{def}}{=} \{(M,t) \mid s \overset{i}{\twoheadrightarrow}_M t\}$.*

Proposition 4.3.2. *Let $s \in S_M$. The state s is determinant w.r.t. $\xrightarrow{i}_M$ if and only if for every $S' \subseteq S_M$: If $\blacktriangleright_i(M,s) \subseteq \blacktriangleright_i(M,S')$ then $s \in S'$.*

Proof. We prove the two implications separately as follows:

- $\Rightarrow$ Assume that $s \in S_M$ is determinant w.r.t. $\xrightarrow{i}_M$. Let us proceed by contradiction and assume that there exists a $S' \subseteq S_M$ such that $\blacktriangleright_i(M,s) \subseteq \blacktriangleright_i(M,S')$ and $s \notin S'$. Since s is determinant w.r.t. $\xrightarrow{i}_M$ there must exists a t such that $s \xrightarrow{i}_M t$ and $s' \not\xrightarrow{i}_M t$ for any $s' \in S'$. By definition of $\blacktriangleright_i$, $(M,t) \in \blacktriangleright_i(M,s)$ and $(M,t) \notin \blacktriangleright_i(M,S')$ which is a contradiction since $\blacktriangleright_i(M,s) \subseteq \blacktriangleright_i(M,S')$.

- $\Leftarrow$ Assume for every $S' \subseteq S_M$, if $\blacktriangleright_i(M,s) \subseteq \blacktriangleright_i(M,S')$ then $s \in S'$. To reach a contradiction, let us suppose that s is not determinant w.r.t. $\xrightarrow{i}_M$. By our assumption that s is not determinant w.r.t. $\xrightarrow{i}_M$ it follows that for any t_i such that $s \xrightarrow{i}_M t_i$ there must be $s_i \neq s$ such that $s_i \xrightarrow{i}_M t_i$. Now, take $S'' = \{s_1, s_2, \ldots\}$. Notice that $\blacktriangleright_i(M,s) \subseteq \blacktriangleright_i(M,S'')$. By applying the assumption, we deduce $s \in S''$, a contradiction.

$\square$

The following theorem provides a complete characterization, in terms of classes of KS, of the existence of right inverses for space functions.

Theorem 4.3.1 (Completeness). *Let $[\cdot]_i$ be a spatial function of a Kripke scs $\mathbf{K}(\mathcal{S})$. Then $[\cdot]_i$ has a right inverse if and only if for every $M \in \mathcal{S}$ the accessibility relation $\xrightarrow{i}_M$ is determinant-complete.*

Proof. We prove the two implications separately as follows:

- $\Leftarrow$ Suppose that for every $M \in \mathcal{S}$, $\xrightarrow{i}_M$ is determinant-complete. By the Axiom of Choice, $[\cdot]_i$ has a right inverse if $[\cdot]_i$ is surjective. Thus, it suffices to show that for every set of pointed KS d, there exists a set of pointed KS c such that $[c]_i = d$. Take an arbitrary d and let $c = \blacktriangleright_i(M',S')$ where $S' = \{s \mid (M,s) \in d\}$. From Definition 4.3.1 we conclude $d \subseteq [c]_i$. It remains to prove $d \supseteq [c]_i$. Suppose $d \not\supseteq [c]_i$. Since $d \subseteq [c]_i$ we have $d \subset [c]_i$. Then there must be a (M',s'), with $M' \in \mathcal{S}$, such that $(M',s') \notin d$ and $(M',s') \in [c]_i$. But if $(M',s') \in [c]_i$ then from Definition 4.3.1 we conclude that $\blacktriangleright_i(M',s') \subseteq c = \blacktriangleright_i(M',S')$. Furthermore $(M',s') \notin d$ implies $s' \notin S'$. It then follows from Proposition 4.3.2 that s' is not determinant w.r.t. $\xrightarrow{i}_{M'}$. This leads us to a contradiction since $\xrightarrow{i}_{M'}$ is supposed to be determinant-complete.

- $\Rightarrow$ Suppose $[\cdot]_i$ has a right inverse. By the Axiom of Choice, $[\cdot]_i$ is surjective. We claim that $\xrightarrow{i}_M$ is determinant-complete for every $M \in \mathcal{S}$. To show this claim let us assume there is $M' \in \mathcal{S}$ such that $\xrightarrow{i}_M$ is not determinant-complete. From Proposition 4.3.2 we should have $s \in S$ and $S' \subseteq S$ such that $\blacktriangleright_i(M',s) \subseteq \blacktriangleright_i(M',S')$ and $s \notin S'$. Since $[c']_i$ is surjective, there must be a set of pointed KS c' such that $\{(M',s') \mid s' \in S'\} = [c']_i$. We can then verify, using Definition 4.3.1, that $\blacktriangleright_i(M,S') \subseteq c'$. Since $\blacktriangleright_i(M',s) \subseteq \blacktriangleright_i(M',S')$ then

$\blacktriangleright_i(M', s) \subseteq c'$. It then follows from Definition 4.3.1 that $(M', s) \in [c']_i$. But $[c']_i = \{(M', s') \mid s' \in S'\}$ then $s \in S'$, a contradiction.

□

Henceforth we use $\mathcal{M}^{\mathtt{D}}$ to denote the class of KS's whose accessibility relations are determinant-complete. It follows from Theorem 4.3.1 that $\mathcal{S} = \mathcal{M}^{\mathtt{D}}$ is the largest class for which space functions of a Kripke $\mathbf{K}(\mathcal{S})$ have right inverses.

4.3.3 Deriving Greatest Right-Inverse

Let $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ be the Kripke scs. The Axiom of Choice and Theorem 4.3.1 tell us that each $[\cdot]_i$ has a right inverse (extrusion function) if and only if $\mathcal{S} \subseteq \mathcal{M}^{\mathtt{D}}$. We are interested, however, in explicit constructions of the right inverses.

Remark 4.3.2. *Recall from Remark 4.3.1 that any Kripke scs* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *is ordered by reversed inclusion (i.e., $c \sqsubseteq d$ iff $d \subseteq c$). Thus, for example, saying that some f is the least function w.r.t. $\subseteq$ satisfying certain conditions is equivalent to saying that f is the greatest function w.r.t. $\sqsubseteq$ satisfying the same conditions. As usual given two self-maps f and g over Con we define $f \sqsubseteq g$ iff $f(c) \sqsubseteq g(c)$ for every $c \in Con$.*

Since any Kripke scs space function preserve arbitrary suprema (Proposition 4.3.1), we can apply Property (2) in Proposition 4.1.1 to obtain the following canonical greatest right-inverse construction. Recall that $[c]_i^{-1} = \{d \mid c = [d]_i\}$ denotes the pre-image of c under $[\cdot]_i$.

Definition 4.3.3 (Max Right Inverse). *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *be a Kripke scs over $\mathcal{S} \subseteq \mathcal{M}^{\mathtt{D}}$. We define $\uparrow_i^{\mathtt{M}}$ as the following self-map on Con: $\uparrow_i^{\mathtt{M}} : c \mapsto \bigsqcup [c]_i^{-1}$.* □

It follows from Property (2) in Proposition 4.1.1 that $\uparrow_i^{\mathtt{M}}$ is a right inverse of $[\cdot]_i$, and furthermore, from its definition it is clear that $\uparrow_i^{\mathtt{M}}$ is the greatest right inverse of $[\cdot]_i$ w.r.t. $\sqsubseteq$.

In this section we singled out determinacy-completeness as a necessary and sufficient condition on KS's to guarantee the existence of right inverses (Theorem 4.3.1). As mentioned before $\mathcal{M}^{\mathtt{D}}$ is used to denote the class of KS's whose accessibility relations are determinant-complete; $\mathcal{M}^{\mathtt{D}}$ is thus the largest class for which space functions of a Kripke scs have right inverses. We shall assume Kripke scs's $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ with $\mathcal{S} \subseteq \mathcal{M}^{\mathtt{D}}$.

We also derived $\uparrow_i^{\mathtt{M}}$ as the greatest right inverse of $[\cdot]_i$ w.r.t. to the underlying Kripke scs order. As shown next, however, $\uparrow_i^{\mathtt{M}}$ is not necessarily *normal* in the sense of Definition 4.2.2. In the next section we shall derive and classify normal right inverses.

4.4 Deriving Normal Right-Inverses

This section is devoted to provide a complete taxonomy, based on the underlying scs order, of right inverse constructions that are normal. As discussed later in this dissertation, the Kripke semantics of several inverse modalities in the literature corresponds to normal right inverses of space functions. The notion of indeterminacy and multiply determinacy introduced in Definition 4.4.1 will play a central role.

Let us first extend the terminology in Definition 4.3.2.

Definition 4.4.1 (Indeterminacy and Multiple Determinacy)**.** *Let S and $\mathcal{R}$ be the set of states and an accessibility relation of a KS M, respectively. Given $t \in S$, we say that t is* determined *w.r.t. $\mathcal{R}$ if there is $s \in S$ such that s determines t w.r.t. $\mathcal{R}$, else we say that t is* indetermined *(or* initial*) w.r.t. $\mathcal{R}$. Similarly, we say that t is* multiply, or ambiguously, determined *if it is determined by at least two different states in S w.r.t. $\mathcal{R}$.*

(As in Definition 4.3.2, we shall often omit "w.r.t. $\mathcal{R}$" when no confusion can occur.) □

The following statement and Theorem 4.2.1 lead us to conclude that the presence of indetermined/initial states or multiple-determined states causes $\uparrow_i^{\mathtt{M}}$ not to be normal.

Proposition 4.4.1. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *and* $\uparrow_i^{\mathtt{M}}$ *as in Definition 4.3.3. Let* $\mathtt{nd}(\mathcal{S}) = \{(M,t) \mid M \in \mathcal{S} \;\&\; t$ *is indetermined w.r.t.* $\xrightarrow{i}_M\}$ *and* $\mathtt{md}(\mathcal{S}) = \{(M,t) \mid M \in \mathcal{S} \;\&\; t$ *is multiply determined w.r.t.* $\xrightarrow{i}_M\}$*:*

- *If* $\mathtt{nd}(\mathcal{S}) \neq \emptyset$ *then* $\uparrow_i^{\mathtt{M}}(true) \neq true$*.*
- *If* $\mathtt{md}(\mathcal{S}) \neq \emptyset$ *then* $\uparrow_i^{\mathtt{M}}(c \sqcup d) \neq \uparrow_i^{\mathtt{M}}(c) \sqcup \uparrow_i^{\mathtt{M}}(d)$ *for some* $c, d \in Con$*.*

Proof.

- If $\mathtt{nd}(\mathcal{S}) \neq \emptyset$ then $\uparrow_i^{\mathtt{M}}(true) \neq true$.

  Assume $\mathtt{nd}(\mathcal{S}) \neq \emptyset$. Take c to be the complement of $\mathtt{nd}(\mathcal{S})$ in Con. Using Equation 4.3.1 we have (1) $[c]_i = true$ and from the assumption (2) $c \sqsupset true$ (recall that $true$ is the set of all pointed KS, see Remark 4.3.1). From (1) and (2) we have $\uparrow_i^{\mathtt{M}}(true) = \bigsqcup [c]_i^{-1} \neq true$.

- If $\mathtt{md}(\mathcal{S}) \neq \emptyset$ then $\uparrow_i^{\mathtt{M}}(c \sqcup d) \neq \uparrow_i^{\mathtt{M}}(c) \sqcup \uparrow_i^{\mathtt{M}}(d)$ for some $c, d \in Con$.

  Suppose $\mathtt{md}(\mathcal{S}) \neq \emptyset$. Take $t \in \mathtt{md}(\mathcal{S})$, then there exist s, s' s.t. $s \xrightarrow{i}_M t$ and $s' \xrightarrow{i}_M t$. Let $c = \{(M,s)\}$ and $d = \{(M,s')\}$. Recall that $\uparrow_i^{\mathtt{M}}$ is a right inverse of $[\cdot]_i$. Because of this any constraint $e = [\uparrow_i^{\mathtt{M}}(e)]_i = \{(M,s'') \mid \blacktriangleright_i(M,s'') \subseteq \uparrow_i^{\mathtt{M}}(e)\}$, therefore $(M,t) \in \uparrow_i^{\mathtt{M}}(c) \cap \uparrow_i^{\mathtt{M}}(d)$. Now, $\uparrow_i^{\mathtt{M}}(c \cap d) = \uparrow_i^{\mathtt{M}}(false) = false$ because of Proposition 4.1.1 and $\uparrow_i^{\mathtt{M}}(false) = \bigsqcup [false]_i^{-1}$ (recall that $false$ is the empty set, see Remark 4.3.1). We therefore conclude $\uparrow_i^{\mathtt{M}}(c \sqcup d) = \uparrow_i^{\mathtt{M}}(c \cap d) \neq \uparrow_i^{\mathtt{M}}(c) \cap \uparrow_i^{\mathtt{M}}(d) = \uparrow_i^{\mathtt{M}}(c) \sqcup \uparrow_i^{\mathtt{M}}(d)$

□

The following central lemma provides distinctive properties of *any* normal right inverse.

Lemma 4.4.1. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *be the Kripke scs over* $\mathcal{S} \subseteq \mathcal{M}^{\mathcal{D}}$*. Suppose that* f *is a normal right inverse of* $[\cdot]_i$*. Then for every* $M \in \mathcal{S}$*,* $c \in Con$*:*

1. $\blacktriangleright_i(M,s) \subseteq f(c)$ *if* $(M,s) \in c$*,*
2. $\{(M,t)\} \subseteq f(c)$ *if* t *is multiply determined w.r.t.* $\xrightarrow{i}_M$*, and*

3. *true* $\subseteq f(\mathit{true})$.

Proof. The proof proceeds as follows:

1. $\blacktriangleright_i(M,s) \subseteq f(c)$ if $(M,s) \in c$. Assume $(M,s) \in c$. By definition of $\blacktriangleright_i$ we have $\blacktriangleright_i(M,s) = \{(M,t) \mid s \xrightarrow{i}_M t\}$. Then for every (M,t) belonging to $\blacktriangleright_i(M,s)$ t is determined w.r.t. $\xrightarrow{i}_M$. Given that f is a right inverse of $[\cdot]_i$, it must satisfy $[f(c)]_i = c$. By Definition 4.3.1 we have $[c]_i = \{(M,s) \mid \blacktriangleright_i(M,s) \subseteq c\}$, then $[f(c)]_i = \{(M,s) \mid \blacktriangleright_i(M,s) \subseteq f(c)\}$. Therefore, since $[f(c)]_i = c = \{(M,s) \mid \blacktriangleright_i(M,s) \subseteq f(c)\}$ then $\blacktriangleright_i(M,s) \subseteq f(c)$.

2. $\{(M,t)\} \subseteq f(c)$ if t is multiply determined w.r.t. $\xrightarrow{i}_M$. Assume t is multiply determined w.r.t. $\xrightarrow{i}_M$. Therefore we know that there exist s, s' with $s \neq s'$ such that $s \xrightarrow{i}_M t$ and $s' \xrightarrow{i}_M t$. From Corollary 4.2.1 we know if $c \subseteq d$ then $f(c) \subseteq f(d)$. Then it follows that $\mathit{false} \subseteq d$ and $f(\mathit{false}) \subseteq f(d)$. Thus, it suffices to prove that $(M,t) \in f(\mathit{false})$. From the assumption that f is a normal right inverse then by Theorem 4.2.1 it preserves suprema $f(\{(M,s)\} \cap \{(M,s')\}) = f(\{(M,s)\}) \cap f(\{(M,s')\})$. Since $s \neq s'$ then $\{(M,s)\} \cap \{(M,s')\} = \emptyset$. From Lemma 4.4.1 (1) $s \xrightarrow{i}_M t$ and $s' \xrightarrow{i}_M t$. Then we know $\{(M,t)\} \subseteq f(\{(M,s)\}) \cap f(\{(M,s')\})$. Therefore by Corollary 4.2.1, $\{(M,t)\} \subseteq f(\{(M,s)\} \cap \{(M,s')\})$ which is the same as $\{(M,t)\} \subseteq f(\mathit{false})$. Then, we know $f(\mathit{false}) \subseteq f(c)$ therefore $\{(M,t)\} \subseteq f(c)$ as wanted.

3. *true* $\subseteq f(\mathit{true})$. Straightforward from the assumption that f is a normal-right inverse.

□

It follows from the above lemma that every normal right inverse f, $f(c)$ must necessarily include every (M,t) such that t is multiply determined w.r.t. $\xrightarrow{i}_M$ as well as every (M,t) such that t is uniquely determined w.r.t. $\xrightarrow{i}_M$ by some s such that $(M,s) \in c$.

4.4.1 Deriving Greatest Normal Right Inverse

Lemma 4.4.1 above tells us which sets should necessarily be included in every $f(c)$ if f is to be both normal and a right inverse of $[\cdot]_i$. It turns out that it is sufficient to include exactly those sets to obtain a normal right inverse of $[\cdot]_i$. In other words Lemma 4.4.1 gives us a *complete* set of conditions for normal right inverses. In fact, the least self-map f w.r.t. $\subseteq$, i.e., the greatest one w.r.t. the lattice order $\sqsubseteq$, satisfying Conditions 1,2 and 3 in Lemma 4.4.1 is indeed a normal right inverse. We shall call such a function the *max normal right inverse* $\uparrow_i^{\mathtt{MN}}$.

Definition 4.4.2 (Max Normal-Right Inverse)**.** *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq)$ *be a Kripke scs over* $\mathcal{S} \subseteq \mathcal{M}^{\mathcal{D}}$*. We define the* max normal right inverse *for agent* i*,* $\uparrow_i^{\mathtt{MN}}$ *as the following self-map on Con:*

$$\uparrow_i^{\mathtt{MN}}(c) \stackrel{\mathtt{def}}{=} \begin{cases} true & \text{if } c = true \\ \{(M,t) \mid t \text{ is determined w.r.t. } \xrightarrow{i}_M \& \forall s : s \xrightarrow{i}\!\!\triangleright_M t, (M,s) \in c\} \end{cases} \tag{4.4.1}$$

(Recall that $s \xrightarrow{i}\!\!\triangleright_M t$ *means that* s uniquely determines t *w.r.t.* $\xrightarrow{i}_M$*.)* □

The following theorem states that $\uparrow_i^{\mathtt{MN}}(c)$ is indeed the greatest normal right inverse of $[\cdot]_i$ w.r.t. $\sqsubseteq$.

Theorem 4.4.1. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *and* $\uparrow_i^{\mathtt{MN}}$ *as in Definition 4.4.2.*

- *The self-map* $\uparrow_i^{\mathtt{MN}}$ *is a normal right inverse of* $[\cdot]_i$*,*
- *For every normal right inverse* f *of* $[\cdot]_i$*, we have* $f \sqsubseteq \uparrow_i^{\mathtt{MN}}$*.*

Proof.
- The self-map $\uparrow_i^{\mathtt{MN}}$ is a normal right inverse of $[\cdot]_i$.

  To prove that $\uparrow_i^{\mathtt{MN}}$ is a right inverse, we shall prove that $[\uparrow_i^{\mathtt{MN}}(c)]_i = c$. From Theorem 4.2.1 it suffices to show that $\uparrow_i^{\mathtt{MN}}(true) = true$ and $\uparrow_i^{\mathtt{MN}}(c \sqcup d) = \uparrow_i^{\mathtt{MN}}(c) \sqcup \uparrow_i^{\mathtt{MN}}(d)$ (equivalent to proving $\uparrow_i^{\mathtt{MN}}(c \cap d) = \uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d)$).

- $[\uparrow_i^{\mathtt{MN}}(c)]_i = c$. Suppose $(M,s) \in [\uparrow_i^{\mathtt{MN}}(c)]_i$. Then $\blacktriangleright_i(M,s) \subseteq \uparrow_i^{\mathtt{MN}}(c)$ from Definition 4.3.1. Since s is determinant w.r.t. $\xrightarrow{i}_M$, there exists t s.t. $s \overset{i}{\twoheadrightarrow}_M t$ and $(M,t) \in \uparrow_i^{\mathtt{MN}}(c)$. Therefore by definition of $\uparrow_i^{\mathtt{MN}}$, $(M,s) \in c$. Now suppose $(M,s) \in c$, then for all states in $\blacktriangleright_i(M,s)$, they are either uniquely determined by s or multiply determined w.r.t. $\xrightarrow{i}_M$, therefore $\blacktriangleright_i(M,s) \subseteq \uparrow_i^{\mathtt{MN}}(c)$, and consequently $(M,s) \in [\uparrow_i^{\mathtt{MN}}(c)]_i$.
- $\uparrow_i^{\mathtt{MN}}(\mathit{true}) = \mathit{true}$. Straightforward by definition of $\uparrow_i^{\mathtt{MN}}$.
- $\uparrow_i^{\mathtt{MN}}(c \cap d) = \uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d)$. We first prove monotonicity (i.e. if $c \subseteq d$ then $\uparrow_i^{\mathtt{MN}}(c) \subseteq \uparrow_i^{\mathtt{MN}}(d)$). For this, suppose $c \subseteq d$. If $(M,t) \in \uparrow_i^{\mathtt{MN}}(c)$ then by definition of $\uparrow_i^{\mathtt{MN}}$ t is determined w.r.t. $\xrightarrow{i}_M$. If there exists a state s with $(M,s) \in c$ s.t. $s \overset{i}{\twoheadrightarrow}_M t$ then $(M,s) \in d$, thus $(M,t) \in \uparrow_i^{\mathtt{MN}}(d)$. If such state does not exists, then t is multiply determined w.r.t. $\xrightarrow{i}_M$ and consequently $(M,t) \in \uparrow_i^{\mathtt{MN}}(d)$. We now proceed with the proof.
  * $\uparrow_i^{\mathtt{MN}}(c \cap d) \subseteq \uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d)$. Since $c \cap d \subseteq c$ and $c \cap d \subseteq d$, then $\uparrow_i^{\mathtt{MN}}(c \cap d) \subseteq \uparrow_i^{\mathtt{MN}}(c)$ and $\uparrow_i^{\mathtt{MN}}(c \cap d) \subseteq \uparrow_i^{\mathtt{MN}}(d)$ by monotonicity of $\uparrow_i^{\mathtt{MN}}$. Thus $\uparrow_i^{\mathtt{MN}}(c \cap d) \subseteq \uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d)$.
  * $\uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d) \subseteq \uparrow_i^{\mathtt{MN}}(c \cap d)$. Suppose $(M,t) \in \uparrow_i^{\mathtt{MN}}(c) \cap \uparrow_i^{\mathtt{MN}}(d)$ then from the definition of $\uparrow_i^{\mathtt{MN}}$, t is determined w.r.t. $\xrightarrow{i}_M$ of $\uparrow_i^{\mathtt{MN}}$. If there exists s s.t. $s \overset{i}{\twoheadrightarrow}_M t$, then $(M,s) \in c$ and $(M,s) \in d$. Suppose there exists such state s, therefore $(M,s) \in c \cap d$ and $(M,t) \in \uparrow_i^{\mathtt{MN}}(c \cap d)$. If not, then t is multiply determined w.r.t. $\xrightarrow{i}_M$ hence $(M,t) \in \uparrow_i^{\mathtt{MN}}(c \cap d)$ by definition of $\uparrow_i^{\mathtt{MN}}$.

- For every normal right inverse f of $[\cdot]_i$, we have $f \sqsubseteq \uparrow_i^{\mathtt{MN}}$.

  Suppose f is a normal right inverse. We then need to prove $\uparrow_i^{\mathtt{MN}}(c) \subseteq f(c)$ for every c. Recall that $\sqsubseteq$ is $\supseteq$ in $\mathbf{K}(\mathcal{S})$. It suffices to prove that if $(M,t) \in \uparrow_i^{\mathtt{MN}}(c)$ then $(M,t) \in f(c)$. Take any $(M,t) \in \uparrow_i^{\mathtt{MN}}(c)$. Then:

  1. t is determined w.r.t. $\xrightarrow{i}_M$, and

2. if t is uniquely determined w.r.t. $\xrightarrow{i}_M$, then $s \overset{i}{\rightarrowtriangle}_M t$ for some $(M, s) \in c$.

Suppose t is uniquely determined w.r.t. $\xrightarrow{i}_M$. Therefore, by Condition 1 of Lemma 4.4.1 (i.e. $\blacktriangleright_i(M, s) \subseteq f$ if $(M, s) \in c$) we have that $(M, t) \in f(c)$. Now suppose it is not uniquely determined, then t is multiply determined w.r.t. $\xrightarrow{i}_M$. Therefore by Condition 2 of Lemma 4.4.1 $(M, t) \in f(c)$.

□

In the following example we will illustrate that if we were to exclude the multiple determined states, $\uparrow_i^{\mathtt{MN}}(\cdot)$ would not be a right inverse of $[\cdot]_i$.

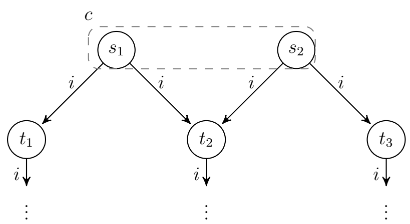


Figure 4.2 – Accessibility relations corresponding to the KS M for an agent i.

Example 4.4.1. *Consider Figure 4.2 with $c = \{(M, s_1), (M, s_2)\}$. From Definition 4.4.2 we obtain*

$$\uparrow_i^{\mathtt{MN}}(c) = \{(M, t_1), (M, t_2), (M, t_3)\}.$$

Let us now consider $h(c) = \uparrow_i^{\mathtt{MN}}(c) \setminus \{(M, t) \mid$ there exists $s, s', s \neq s', s \xrightarrow{i}_M t, s' \xrightarrow{i}_M t\}$ to be $\uparrow_i^{\mathtt{MN}}(c)$ minus the pointed KS's multiply determined. We would then obtain

$$h(c) = \{(M, t_1), (M, t_3)\}.$$

Notice that any self-map g such that $g(c) = h(c)$ cannot be a right inverse of $[\cdot]_i$ from Definition 4.3.1 we would have

$$[g(c)]_i = \emptyset \neq c.$$

□

Notice that $\uparrow_i^{\mathtt{MN}}(c)$ excludes indetermined states if $c \neq \mathit{true}$. It turns out that we can add them and still obtain a normal right inverse. We shall see in the next section that this kind of normal right inverse arises in the context of linear-time temporal logic.

Definition 4.4.3 (Normal-Right Inverse). *Let* $\mathbf{K}(\mathcal{S}) = (\mathit{Con}, \sqsubseteq)$ *be a Kripke scs over* $\mathcal{S} \subseteq \mathcal{M}^{\mathtt{D}}$. *Define* $\uparrow_i^{\mathtt{N}} : \mathit{Con} \to \mathit{Con}$ *as* $\uparrow_i^{\mathtt{N}}(c) \stackrel{\mathtt{def}}{=} \{(M,t) \mid \forall s : s \stackrel{i}{\twoheadrightarrow}_M t, (M,s) \in c\}$. □

Clearly $\uparrow_i^{\mathtt{N}}(c)$ includes every (M,t) such that t is indetermined w.r.t. $\stackrel{i}{\longrightarrow}_M$. We now show that it is indeed a normal right inverse.

Theorem 4.4.2. *Let* $\mathbf{K}(\mathcal{S}) = (\mathit{Con}, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *and* $\uparrow_i^{\mathtt{N}}$ *as in Definition 4.4.3. The self-map* $\uparrow_i^{\mathtt{N}}$ *is a normal right inverse of* $[\cdot]_i$.

Proof. To prove that $\uparrow_i^{\mathtt{N}}$ is a right inverse we prove $[\uparrow_i^{\mathtt{N}}(c)]_i = c$. To prove that it is a normal map, by Theorem 4.2.1, it suffices to prove that $\uparrow_i^{\mathtt{N}}(\mathit{true}) = \mathit{true}$ and $\uparrow_i^{\mathtt{N}}(c \sqcup d) = \uparrow_i^{\mathtt{N}}(c) \sqcup \uparrow_i^{\mathtt{N}}(d)$ (equivalent to proving that $\uparrow_i^{\mathtt{N}}(c \cap d) = \uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d)$).

- $[\uparrow_i^{\mathtt{N}}(c)]_i = c$. Suppose $(M,s) \in [\uparrow_i^{\mathtt{N}}(c)]_i$. Then $\blacktriangleright_i(M,s) \subseteq \uparrow_i^{\mathtt{N}}(c)$ from Definition 4.3.1. As s is determinant w.r.t. $\stackrel{i}{\longrightarrow}_M$, there exists t s.t. $s \stackrel{i}{\twoheadrightarrow}_M t$ and $(M,t) \in \uparrow_i^{\mathtt{N}}(c)$. Therefore by definition of $\uparrow_i^{\mathtt{N}}$, $(M,s) \in c$. Now suppose $(M,s) \in c$, then all states in $\blacktriangleright_i(M,s)$ are either uniquely determined by s or multiply determined w.r.t. $\stackrel{i}{\longrightarrow}_M$, therefore $\blacktriangleright_i(M,s) \subseteq \uparrow_i^{\mathtt{N}}(c)$, and consequently $(M,s) \in [\uparrow_i^{\mathtt{N}}(c)]_i$.

- $\uparrow_i^{\mathtt{N}}(\mathit{true}) = \mathit{true}$. We have $\uparrow_i^{\mathtt{N}}(\mathit{true}) = \{(M,t) \mid \forall s : s \stackrel{i}{\twoheadrightarrow}_M t, (M,s) \in \mathit{true}\}$. Since $(M,s) \in \mathit{true}$ trivially holds, $(M,t) \in \uparrow_i^{\mathtt{N}}(\mathit{true})$ for any (M,t). Thus $\uparrow_i^{\mathtt{N}}(\mathit{true}) = \mathit{true}$.

- $\uparrow_i^{\mathtt{N}}(c \cap d) = \uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d)$. We first prove monotonicity (i.e. if $c \subseteq d$ then $\uparrow_i^{\mathtt{N}}(c) \subseteq \uparrow_i^{\mathtt{N}}(d)$). For this, suppose $c \subseteq d$ and $(M,t) \in \uparrow_i^{\mathtt{N}}(c)$. If there exists a state s with $(M,s) \in c$ s.t. $s \stackrel{i}{\twoheadrightarrow}_M t$ then $(M,s) \in d$, thus $(M,t) \in \uparrow_i^{\mathtt{N}}(d)$. If such state does not exists, then $(M,t) \in \uparrow_i^{\mathtt{N}}(d)$. We now proceed with the proof.

– $\uparrow_i^{\mathtt{N}}(c \cap d) \subseteq \uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d)$. Since $c \cap d \subseteq c$ and $c \cap d \subseteq d$, by monotonicity of $\uparrow_i^{\mathtt{N}}$, $\uparrow_i^{\mathtt{N}}(c \cap d) \subseteq \uparrow_i^{\mathtt{N}}(c)$ and $\uparrow_i^{\mathtt{N}}(c \cap d) \subseteq \uparrow_i^{\mathtt{N}}(d)$. Therefore $\uparrow_i^{\mathtt{N}}(c \cap d) \subseteq \uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d)$.

– $\uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d) \subseteq \uparrow_i^{\mathtt{N}}(c \cap d)$. Suppose $(M,t) \in \uparrow_i^{\mathtt{N}}(c) \cap \uparrow_i^{\mathtt{N}}(d)$. If there exists s s.t. $s \xrightarrow{i}_M t$, then $(M,s) \in c$ and $(M,s) \in d$. Now suppose there exists such s, therefore $(M,s) \in c \cap d$ and $(M,t) \in \uparrow_i^{\mathtt{N}}(c \cap d)$. If this is not the case, then t is not determined w.r.t. $\xrightarrow{i}_M$ by any other state, hence $(M,t) \in \uparrow_i^{\mathtt{N}}(c \cap d)$ by definition of $\uparrow_i^{\mathtt{N}}$.

□

In the following example we will illustrate that we can include indetermined states by using $\uparrow_i^{\mathtt{N}}(c)$ and still have a right inverse of $[\cdot]_i$.

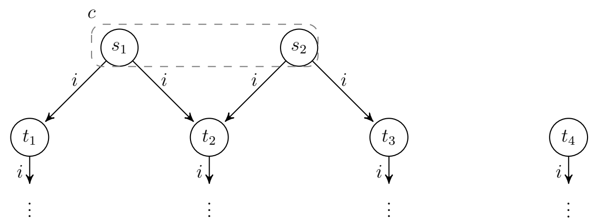


Figure 4.3 – Accessibility relations corresponding to the KS M for an agent i.

Example 4.4.2. *Consider Figure 4.3 with* $c = \{(M,s_1),(M,s_2)\}$. *From Definition 4.4.2 we obtain*

$$\uparrow_i^{\mathtt{MN}}(c) = \{(M,t_1),(M,t_2),(M,t_3)\}.$$

Notice that $\uparrow_i^{\mathtt{MN}}(\cdot)$ *excludes the indetermined state* t_4. *We can include it by using Definition 4.4.3. We then obtain*

$$\uparrow_i^{\mathtt{N}}(c) = \{(M,t_1),(M,t_2),(M,t_3),(M,t_4)\}.$$

By applying Definition 4.3.1 we can show that $\uparrow_i^{\mathtt{N}}(\cdot)$ *is still a right inverse of* $[\cdot]_i$. *We have*

$$[\uparrow_i^{\mathtt{N}}(c)]_i = \{(M,s_1),(M,s_2)\} = c.$$

□

4.4.2 Deriving Minimal Normal-Right Inverses

In Definition 4.4.3 we included in $\uparrow_i^{\mathtt{N}}(c)$ all indetermined states. We did not include, however, the states that are *uniquely determined* by states not in c. It turns out that, under certain conditions, we can add *all but one of them* to $\uparrow_i^{\mathtt{N}}(c)$ and obtain a *minimal* normal right inverse.

We shall see that not adding *all* uniquely determined states is a necessary condition to guarantee that the resulting self-map is still a right inverse. We shall use *choice functions* to select the uniquely-determined states that are not added. Nevertheless, we shall also show that such selections must obey certain conditions to guarantee that the resulting right inverse self-map is still normal.

Remark 4.4.1. *A map m is a* choice *function (or* selector*) for a collection of nonempty sets if it maps each set S in the collection to some element $m(S)$ of S. Recall that $\rhd_i(M,s)$ denotes the set of all (M,t) such that s* uniquely determines *t w.r.t. $\xrightarrow{i}_M$ (see Notation 4.3.2). Notice that $\rhd_i(M,s) \neq \emptyset$ since we are assuming that each (M,s) must be determining w.r.t $\xrightarrow{i}_M$.*

Below we define a minimal right inverse $\uparrow_{i,\mathfrak{s}}^{\mathtt{mN}}$, where $\mathfrak{s}$ is a family of selectors, following the above intuitions. Given $c \in Con$, we shall use a selector $\mathfrak{s}_{\bar{c}} \in \mathfrak{s}$ that chooses an element in each set in the collection $\{\ \rhd_i(M,s)\ \}_{(M,s)\notin c}$. These selected elements are not included in $\uparrow_{i,\mathfrak{s}}^{\mathtt{mN}}(c)$.

Definition 4.4.4 (Minimal Normal-Right Inverse)**.** *Let $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ be a Kripke scs over $\mathcal{S} \subseteq \mathcal{M}^{\mathcal{D}}$. Define $\uparrow_{i,\mathfrak{s}}^{\mathtt{mN}} : Con \rightarrow Con$ as*

$$\uparrow_{i,\mathfrak{s}}^{\mathtt{mN}}(c) \stackrel{\mathtt{def}}{=} \{(M,t) \mid \forall s : s \xrightarrow{i}_M t, (M,s) \in c\} \cup \bigcup_{(M,s)\notin c} \rhd_i(M,s) \setminus \{\mathfrak{s}_{\bar{c}}\ (\ \rhd_i(M,s)\ )\}$$

where $\mathfrak{s}$ is a family $\{\ \mathfrak{s}_{\bar{c}}\ \}_{c\in Con}$ such that

1. *$\mathfrak{s}_{\bar{c}}$ is a selector for $\{\ \rhd_i(M,s)\ \}_{(M,s)\notin c}$, and*

2. *if $c \subseteq d$ then $\mathfrak{s}_{\bar{c}}(\rhd_i(M,s)) = \mathfrak{s}_{\bar{d}}(\rhd_i(M,s))$ for every $(M,s) \notin d$.*

□

From the above definition it is easy to see that $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ indeed excludes uniquely-determined states selected by the choice function $\mathfrak{s}_{\bar{c}}$ in Definition 4.4.4 (1). In the following example we will illustrate that if we were to include these selected states, $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ would not be a right inverse of $[\cdot]_i$.

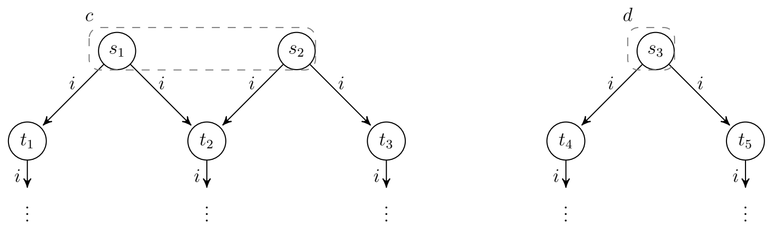


Figure 4.4 – Accessibility relations corresponding to the KS M for an agent i.

Example 4.4.3. *Consider Figure 4.4 with* $c = \{(M, s_1), (M, s_2)\}$ *and* $d = \{(M, s_3)\}$. *Let* $\mathfrak{s}$ *be a family of selectors that includes* $\mathfrak{s}_{\bar{d}}$ *and* $\mathfrak{s}_{\bar{c}}$. *Assume that* $\mathfrak{s}_{\bar{d}}(\rhd_i(M, s_1)) = (M, t_1)$, $\mathfrak{s}_{\bar{d}}(\rhd_i(M, s_2)) = (M, t_3)$ *and* $\mathfrak{s}_{\bar{c}}(\rhd_i(M, s_3)) = (M, t_5)$. *From Definition 4.4.4 we obtain*

$$\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) = \{(M, t_1), (M, t_2), (M, t_3), (M, t_4)\}.$$

Let us now consider $h(c) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cup \mathfrak{s}_{\bar{c}}(\rhd_i(M, s_3))$ *to be* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ *plus the pointed KS from* $\rhd_i(M, s_3)$ *selected by the choice function* $\mathfrak{s}_{\bar{c}}$. *We would then obtain*

$$h(c) = \{(M, t_1), (M, t_2), (M, t_3), (M, t_4), (M, t_5)\}.$$

Notice that any self-map g *such that* $g(c) = h(c)$ *cannot be a right inverse of* $[\cdot]_i$ *for from Definition 4.3.1 we would have*

$$[g(c)]_i = \{(M, s_1), (M, s_2), (M, s_3)\} \neq c.$$

□

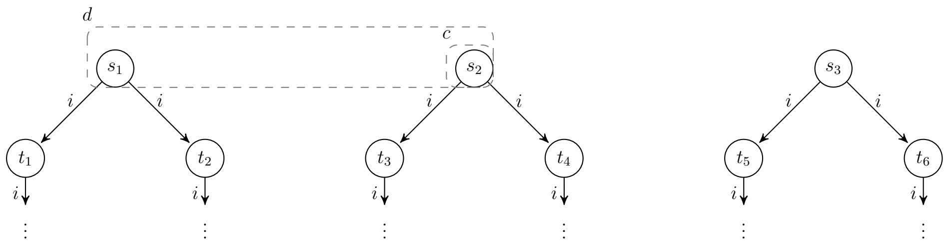


Figure 4.5 – Accessibility relations corresponding to the KS M for an agent i.

Notice that the selection of elements that are not included in $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ need to obey Condition (2) in Definition 4.4.4. This condition is needed for the normality of $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$. We illustrate this in the following example.

Example 4.4.4. *Consider Figure 4.5. Notice that* $c = \{(M, s_2)\}$ *and* $d = \{(M, s_1), (M, s_2)\}$. *Let* $\mathfrak{s}$ *be a family of selectors that includes* $\mathfrak{s}_{\bar{c}}, \mathfrak{s}_{\bar{d}}$ *and* $\mathfrak{s}_{\overline{c \cap d}}$. *Suppose that* $\mathfrak{s}_{\bar{c}}(\rhd_i(M, s_1)) = (M, t_1) = \mathfrak{s}_{\overline{c \cap d}}(\rhd_i(M, s_1))$, $\mathfrak{s}_{\bar{c}}(\rhd_i(M, s_3)) = (M, t_5) = \mathfrak{s}_{\overline{c \cap d}}(\rhd_i(M, s_3))$, *and* $\mathfrak{s}_{\bar{d}}(\rhd_i(M, s_3)) = (M, t_6)$.

We have $c \subseteq d$ *but* $\mathfrak{s}_{\bar{c}}(\rhd_i(M, s_3)) \neq \mathfrak{s}_{\bar{d}}(\rhd_i(M, s_3))$ *hence* $\mathfrak{s}$ *does not satisfy Condition (2) in Definition 4.4.4.*

Notice that if we were to drop Condition (2) in Definition 4.4.4 we would obtain the following.

$$
\begin{aligned}
\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) =& \{(M, t_3), (M, t_4), (M, t_2), (M, t_6)\} \\
\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) =& \{(M, t_1), (M, t_2), (M, t_3), (M, t_4), (M, t_5)\} \\
\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) =& \{(M, t_3), (M, t_4), (M, t_2), (M, t_6)\}
\end{aligned}
$$

And since $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) = \{(M, t_3), (M, t_4), (M, t_2)\}$ *then we would conclude*

$$\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) \neq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d).$$

Then $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *would not be a normal right inverse of* $[\cdot]_i$ *(Theorem 4.2.1)* □

Now we prove that the function $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\cdot)$ presented in Definition 4.4.4 is a normal right inverse of the space function $[\cdot]_i$.

Theorem 4.4.3. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *and* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *as in Definition 4.4.4. The self-map* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *is a normal right inverse of* $[\cdot]_i$.

Proof. To prove that $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ is a right inverse we prove $[\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)]_i = c$. To prove that it is a normal map, by Theorem 4.2.1, it suffices to prove $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\mathit{true}) = \mathit{true}$ and $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \sqcup d) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \sqcup \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$ (equivalent to proving $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$).

- $[\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)]_i = c$. Suppose $(M, s) \in [\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)]_i$. Then $\blacktriangleright_i(M, s) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ from Definition 4.3.1. Since every state from $\mathbf{K}(\mathcal{S})$ is determinant w.r.t. $\xrightarrow{i}_M$, there must be t s.t. $s \overset{i}{\twoheadrightarrow}_M t$ and $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$. Therefore by definition of $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$, $(M, s) \in c$.

  Now suppose $(M, s) \in c$, then for states in $\blacktriangleright_i(M, s)$ are either uniquely determined by s or multiply determined w.r.t. $\xrightarrow{i}_M$, therefore $\blacktriangleright_i(M, s) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$, and consequently $(M, s) \in [\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)]_i$ by Definition 4.3.1.

- $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\mathit{true}) = \mathit{true}$. Recall that *true* is the set of all pointed KS's (M, s) (with $M \in \mathcal{S}$) while *false* is the empty set. Therefore $\emptyset = \bigcup_{(M,s) \notin \mathit{true}} \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s))\}$. From Definition 4.4.4 we obtain $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\mathit{true}) = \{(M, t) \mid \forall s : s \overset{i}{\twoheadrightarrow}_M t, (M, s) \in \mathit{true}\} = \mathit{true}$.

- $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$. We first prove monotonicity of $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ (i.e. if $c \subseteq d$ then $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$). Suppose $c \subseteq d$ and $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$. We prove that $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$. If t is indetermined or multiply determined w.r.t. $\xrightarrow{i}_M$ then, from Definition 4.4.4, $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$. Otherwise, t is uniquely determined and thus we must have an s such that $s \overset{i}{\twoheadrightarrow}_M t$. If $(M, s) \in c$ then since $c \subseteq d$, we have $(M, s) \in d$ and thus $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$ by Definition 4.4.4. Otherwise $(M, s) \notin c$ and since $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ we conclude from Definition 4.4.4 that

  $$(M, t) \in \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s))\}.$$

  If $(M, s) \in d$, then $(M, t) \in \{(M, t) \mid \forall s : s \overset{i}{\twoheadrightarrow}_M t, (M, s) \in d\}$ and $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$. Otherwise $(M, s) \notin d$ and then from Condition (2) in Definition 4.4.4 we conclude that $(M, t) \in \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{d}}(\triangleright_i(M, s))\}$. Hence from Definition 4.4.4 we conclude $(M, t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$.

We now proceed to prove $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) = \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$. The inclusion $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d)$ is an immediate consequence of the the monotonicity of $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\cdot)$. To show that $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d)$ it is convenient to define the following sets:

$$
\begin{aligned}
A'_1 =& \{(M,t) \mid \forall s : s \xrightarrow{i}_M t, (M,s) \in c\},\\
A''_1 =& \{(M,t) \mid \forall s : s \xrightarrow{i}_M t, (M,s) \in d\},\\
A'_2 =& \bigcup_{(M,s) \notin c} \rhd_i(M,s) \setminus \{\mathfrak{s}_{\bar{c}}(\rhd_i(M,s))\},\\
A''_2 =& A''_1, \quad A'_3 = A'_1,\\
A''_3 =& \bigcup_{(M,s) \notin d} \rhd_i(M,s) \setminus \{\mathfrak{s}_{\bar{d}}(\rhd_i(M,s))\},\\
A'_4 =& A'_2, \quad A''_4 = A''_3.
\end{aligned}
$$

And

$$
\begin{aligned}
B_1 =& \{(M,t) \mid \forall s : s \xrightarrow{i}_M t, (M,s) \in c \cap d\},\\
B_2 =& \bigcup_{(M,s) \notin c \cap d} \rhd_i(M,s) \setminus \{\mathfrak{s}_{\overline{c \cap d}}(\rhd_i(M,s))\}.
\end{aligned}
$$

From Definition 4.4.4 we have $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d) = B_1 \cup B_2$. Using Definition 4.4.4 and distributive set laws we obtain $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) = (A'_1 \cup A'_2) \cap (A''_1 \cup A''_3) = \bigcup_{i \in I} A_i$ where $A_i = A'_i \cap A''_i$ for $i \in I = \{1,2,3,4\}$. It is easy to verify that $A_1 \subseteq B_1$ and $A_4 \subseteq B_2$. Using Condition (2) in Definition 4.4.4 we conclude $A'_2 \subseteq B_2$ and $A''_3 \subseteq B_2$. Thus, $A_2 \subseteq A'_2 \subseteq B_2, A_3 \subseteq A''_3 \subseteq B_2$. Therefore $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \cap \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(d) \subseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c \cap d)$.

□

We shall show that the family of normal right inverses in Definition 4.4.4 are *minimal* (w.r.t. $\sqsubseteq$) normal right inverses for the space function $[\cdot]_i$ in the sense that there are no normal right inverses below them. More precisely, h is a *minimal* normal right inverse of $[\cdot]_i$ iff h is a normal right inverse of $[\cdot]_i$ and there is no other normal right inverse $g(\cdot)$ of $[\cdot]_i$ such that $g \sqsubseteq h$.

Theorem 4.4.4. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *and* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *as in Definition 4.4.4. Then* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *is a minimal normal right inverse of* $[\cdot]_i$*.*

Proof. We prove that there is no other normal right inverse $g(\cdot)$ for the space function $[\cdot]_i$ such that $g(c) \sqsubseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ for every $c \in Con$. Recall that $g(c) \sqsubseteq \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ is equivalent to $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \subseteq g(c)$ (Remark 4.3.2).

By reduction to absurd let us assume that there exists a normal right inverse $g(\cdot)$ for $[\cdot]_i$, different from $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(\cdot)$, such that $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \subseteq g(c)$ for all $c \in Con$. Thus there must be (M, t) and $c \in Con$ such that $(M, t) \in g(c)$ and $(M, t) \notin \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$.

Since $(M, t) \notin \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$, we can use Lemma 4.4.1 and Definition 4.4.4 to show that

$$(M, s) \notin c \text{ and that } (M, t) = \mathfrak{s}_{\bar{c}}(\triangleright_i(M, s)) \tag{4.4.2}$$

where s uniquely determines t in M (i.e., $s \overset{i}{\twoheadrightarrow}_M t$).

From Equation 4.4.2 and Definition 4.4.4 it follows that $\triangleright_i(M, s) \setminus \{(M, t)\}$ is included in $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$. But $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \subseteq g(c)$ and $(M, t) \in g(c)$ thus

$$\triangleright_i(M, s) \subseteq g(c). \tag{4.4.3}$$

From Lemma 4.4.1(2) and the assumption that g is a normal right inverse, $g(c)$ includes all (M, t) such that t is multiply determined w.r.t. $\overset{i}{\longrightarrow}_M$. Consequently, using Equation 4.4.3 we conclude

$$\blacktriangleright_i(M, s) \subseteq g(c). \tag{4.4.4}$$

Since we assumed that g is a right inverse of $[\cdot]_i$ we have $[g(c)]_i = c$. We can use Equation 4.4.4 and Definition 4.3.1 to conclude $(M, s) \in [g(c)]_i$. Hence $(M, s) \in c$, but $(M, s) \neq c$ by Equation 4.4.2. □

The above theorem identifies a family of minimal normal right inverses indexed by collections of choice functions. The next theorem tell us that the family is *complete* in the sense that *every* normal right inverse is bounded from below by some minimal right inverse $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$.

Theorem 4.4.5. *Let* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *as in Definition 4.4.4. Suppose that* g *is a normal right inverse of the space function* $[\cdot]_i$*. Then* *there exists a minimal normal right inverse* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ *as in Definition 4.4.4 such that* $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}} \sqsubseteq g$.

Proof. Given a normal right inverse g of $[\cdot]_i$ we will show the existence of a minimal right inverse $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ of $[\cdot]_i$ for some $\mathfrak{s}$ such that for every $c \in \mathit{Con}$ if $(M,t) \in g(c)$ then $(M,t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ (see Remark 4.3.2).

From Theorem 4.4.3, Lemma 4.4.1 and Definition 4.4.4, $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ includes every (M,t) such that t is multiply determined or indetermined w.r.t. $\xrightarrow{i}_M$ as well as every (M,t) such that t is uniquely determined w.r.t. $\xrightarrow{i}_M$ by some s with $(M,s) \in c$. Consequently, it suffices to prove that for every (M,t) such that t is uniquely determined by some s with $(M,s) \notin c$ if $(M,t) \in g(c)$ then $(M,t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$. More precisely, it suffices to prove that there exists $\mathfrak{s}$ such that the following holds for any $c \in \mathit{Con}$:

$$\text{If } (M,t) \in g'(c) \text{ then } (M,t) \in \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \tag{4.4.5}$$

where $g'(e) \stackrel{\mathtt{def}}{=} g(e) \cap \bigcup_{(M,s)\notin e} \rhd_i(M,s)$.

We need to prove some properties about g'. The first property is the following:

$$\text{If } (M,s) \notin c \text{ then } \rhd_i(M,s) \setminus g'(c) \neq \emptyset. \tag{4.4.6}$$

This property follows from the fact that g is normal and a right inverse. We prove something stronger: If $(M,s) \notin c$ then $g'(c) \subset \rhd_i(M,s)$. Notice that if $(M,s) \notin c$ then $g'(c) \subseteq \rhd_i(M,s)$ since the sets $\rhd_i(M,s')$ of states uniquely-determined by s' are mutually exclusive. Then by reduction to absurd suppose $(M,s) \notin c$ and $g'(c) = \rhd_i(M,s)$. Since $g'(c) \subseteq g(c)$ we have $\rhd_i(M,s) \subseteq g(c)$. From Lemma 4.4.1(2) and the assumption that g is a normal right inverse, $g(c)$ includes all (M,t) such that t is multiply determined w.r.t. $\xrightarrow{i}_M$. Since $\rhd_i(M,s) \subseteq g(c)$ we then conclude $\blacktriangleright_i(M,s) \subseteq g(c)$. But g is a right inverse of $[\cdot]_i$ thus $[g(c)]_i = c$. Using Definition 4.3.1 we then conclude $(M,s) \in [g(c)]_i$. Hence $(M,s) \in c$, a contradiction with the assumption in Equation 4.4.6. Then it must be $g'(c) \subset \rhd_i(M,s)$.

The second property we prove about g' is the following:

$$\text{If } c \subseteq d \text{ and } (M,s) \notin d \text{ then } \rhd_i(M,s) \setminus g'(d) \;\subseteq\; \rhd_i(M,s) \setminus g'(c) \tag{4.4.7}$$

To prove this property, suppose that $c \sqsubseteq d$, $(M,s) \notin d$, and $(M,t) \in \rhd_i(M,s) \backslash g'(d)$. Thus $(M,t) \in \rhd_i(M,s)$ and $(M,t) \notin g'(d)$. Since $(M,s) \notin d$ then $(M,t) \in \bigcup_{(M,s)\notin d} \rhd_i(M,s)$. As $(M,t) \notin g'(d)$ then $(M,t) \notin g(d)$. Since g is normal then it is monotone (Corollary 4.2.1), consequently $c \sqsubseteq d$ and $(M,t) \notin g(d)$ implies $(M,t) \notin g(c)$. Hence $(M,t) \notin g'(c)$. It follows that $(M,t) \in \rhd_i(M,s) \setminus g'(c)$ as wanted.

To conclude the proof we now identify a minimal normal right inverse $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ that satisfies Equation 4.4.5. Let $\mathfrak{s}$ to be a family of choice functions $\{\mathfrak{s}_{\bar{c}}\}_{c \in Con}$ such that (I) $\mathfrak{s}_{\bar{c}}(\rhd_i(M,s)) \in \rhd_i(M,s) \setminus g'(c)$ and (II) $\mathfrak{s}_{\bar{c}}(\rhd_i(M,s)) = \mathfrak{s}_{\bar{d}}(\rhd_i(M,s))$ for every $(M,s) \notin d$ whenever $c \sqsubseteq d$. Notice that because of Equation 4.4.6 condition (I) can be fulfilled. Similarly, because of Equations 4.4.6 and 4.4.7, condition (II) can also be fulfilled. Clearly $\mathfrak{s}$ satisfies the selection conditions in Definition 4.4.4. Thus we have

$$\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) = \{(M,t) \mid \forall s : s \xrightarrow{i}_M t, (M,s) \in c\} \cup \bigcup_{(M,s)\notin c} \rhd_i(M,s) \setminus \{\mathfrak{s}_{\bar{c}}(\rhd_i(M,s))\}$$

Because of (I), this minimal right inverse $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ satisfies Equation 4.4.5 which concludes the proof.

□

In this section we have derived normal right inverses and classified them according to the underlying Kripke scs order. We focused on normal right inverses, because as we shall illustrate in the next section, they are ubiquitous in modal logic. We identified the greatest normal right inverse $\uparrow^{\mathtt{MN}}_{i}$, the normal right inverse $\uparrow^{\mathtt{N}}_{i}$ and the family of *all* minimal right inverses, denoted as $\{\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}\}$. The following corollary summarizes our classification.

Corollary 4.4.1 (Taxonomy)**.** *Fix a Kripke scs* $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *with* $\mathcal{S} \subseteq \mathcal{M}^{\mathcal{D}}$*. Then*

1. $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}} \sqsubseteq \uparrow^{\mathtt{N}}_{i} \sqsubseteq \uparrow^{\mathtt{MN}}_{i}$ *for every* $\mathfrak{s}$ *as in Definition 4.4.4, and*

2. *for every normal right inverse g of $[\cdot]_i$, there exists $\mathfrak{s}$ as in Definition 4.4.4 such that*

$$\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}} \sqsubseteq g \sqsubseteq \uparrow^{\mathtt{MN}}_{i} \quad (\textit{i.e.}, \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c) \supseteq g(c) \supseteq \uparrow^{\mathtt{MN}}_{i}(c) \textit{ for every } c \in Con.)$$

where $\uparrow^{\mathtt{MN}}_{i}, \uparrow^{\mathtt{N}}_{i}$ and $\uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}$ are given as in Definitions 4.4.2, 4.4.3 and 4.4.4 w.r.t. $\mathbf{K}(\mathcal{S})$.

The upper and lower bounds in Corollary 4.4.1(2) are useful to prove whether some e is in a given normal-right inverse g. Thus if $e \in \uparrow^{\mathtt{MN}}_{i}(c)$ then $e \in g(c)$ and if $e \notin \uparrow^{\mathtt{mN}}_{i,\mathfrak{s}}(c)$ for every $\mathfrak{s}$ then $e \notin g(c)$. We shall use these properties in the next section.

4.5 Applications

In this section we will illustrate and briefly discuss the results obtained in the previous sections in the context of modal logic.

Modal formulae can be interpreted as constraints in the scs $\mathbf{K}(\mathcal{S}_n(\Phi))$. First recall the definition of the notion of modal language given in Definition 3.1.1. In Definition 3.1.3 we have formally introduced when a pointed KS (M, s) satisfies a given formula ϕ, denoted $(M, s) \models \phi$. Notice that this notion of satisfiability is invariant under a standard equivalence on Kripke structures: *Bisimilarity*, itself a central equivalence in concurrency theory [Mil89].

Definition 4.5.1 (Bisimilarity)**.** *Let $\mathcal{B}$ a symmetric relation on pointed KS's. The relation is said to be a* bisimulation *iff for every $((M, s), (N, t)) \in \mathcal{B}$: (1) $\pi_M(s) = \pi_N(t)$ and (2) if $s \xrightarrow{i}_M s'$ then there exists t' s.t. $t \xrightarrow{i}_N t'$ and $((M, s'), (N, t')) \in \mathcal{B}$. We say that (M, s) and (N, t) are (strongly) bisimilar, written $(M, s) \cong (N, t)$ if there exists a bisimulation $\mathcal{B}$ such that $((M, s), (N, t)) \in \mathcal{B}$.* □

Bisimilarity-invariance The well-known result of *bisimilarity-invariance* of modal satisfiability [HM85] implies that if $(M, s) \cong (N, t)$ then (M, s)

and (M, t) satisfy exactly the same formulae in $\mathcal{L}_n(\Phi)$. Modal logics are typically interpreted over different classes of KS's obtained by imposing conditions on their accessibility relations. Let $\mathcal{S}_n(\Phi)$ be a non-empty set of n-agent Kripke structures over a set of primitive propositions Φ. A modal formula ϕ is said to be *valid* in $\mathcal{S}_n(\Phi)$ iff $(M, s) \models \phi$ for each (M, s) such that $M \in \mathcal{S}_n(\Phi)$.

We can interpret modal formulae as constraints in a given Kripke scs $\mathbf{C} = \mathbf{K}(\mathcal{S}_n(\Phi))$ as follows.

Definition 4.5.2 (Kripke Constraint Interpretation)**.** *Let* $\mathbf{C}$ *be a Kripke scs* $\mathbf{K}(\mathcal{S}_n(\Phi))$*. Given a modal formula* ϕ *in the modal language* $\mathcal{L}_n(\Phi)$*, its interpretation in the Kripke scs* $\mathbf{C}$ *is the constraint* $\mathbf{C}[\![\phi]\!]$ *inductively defined as follows:*

$$
\begin{array}{rcl}
\mathbf{C}[\![p]\!] & = & \{(M, s) \mid \pi_M(s)(p) = 1\} \\
\mathbf{C}[\![\phi \wedge \psi]\!] & = & \mathbf{C}[\![\phi]\!] \sqcup \mathbf{C}[\![\psi]\!] \\
\mathbf{C}[\![\neg\phi]\!] & = & \sim \mathbf{C}[\![\phi]\!] \\
\mathbf{C}[\![\Box_i\phi]\!] & = & [\ \mathbf{C}[\![\phi]\!]\ ]_i
\end{array}
$$

□

Remark 4.5.1. *One can verify that for any Kripke scs* $\mathbf{K}(\mathcal{S}_n(\Phi))$*, the Heyting negation* $\sim c$ *(Def. 2.3.2) is* $\Delta \setminus c$ *where* Δ *is the set of all pointed Kripke structures* (M, s) *such that* $M \in \mathcal{S}_n(\Phi)$ *(i.e., boolean negation). Similarly, Heyting implication* $c \rightarrow d$ *is equivalent to* $(\sim c) \cup d$ *(i.e., boolean implication).*

It is easy to verify that the constraint $\mathbf{C}[\![\phi]\!]$ includes those pointed KS (M, s), where $M \in \mathcal{S}_n(\Phi)$, such that $(M, s) \models \phi$. Thus, ϕ is valid in $\mathcal{S}_n(\Phi)$ if and only if $\mathbf{C}[\![\phi]\!] = \mathit{true}$.

Notice that from Proposition 4.3.1 and Theorem 4.2.1, each space function $[\cdot]_i$ of $\mathbf{K}(\mathcal{S}_n(\Phi))$ is a normal self-map. From Definitions 4.2.2 and 4.5.2 we can derive the following standard property stating that $\Box_i$ is a normal modal operator: (*Necessitation*) If ϕ is valid in $\mathcal{S}_n(\Phi)$ then $\Box_i\phi$ is valid in $\mathcal{S}_n(\Phi)$, and (*Distribution*) $\Box_i(\phi \Rightarrow \psi) \Rightarrow (\Box_i\phi \Rightarrow \Box_i\psi)$ is valid in $\mathcal{S}_n(\Phi)$.

4.5.1 Right-Inverse Modalities

Reverse modalities, also known as inverse modalities, arise naturally in many modal logics. For example in temporal logics they are past operators [PM92], in modal logics for concurrency they represent backward moves [PU11], in epistemic logic they correspond to utterances [HPRV15].

To illustrate our results in the previous sections, let us fix a modal language $\mathcal{L}_n(\Phi)$ (whose formulae are) interpreted in an arbitrary Kripke scs $\mathbf{C} = \mathbf{K}(\mathcal{S}_n(\Phi))$. Suppose we wish to extend it with modalities $\Box_i^{-1}$, called reverse modalities also interpreted over the same set of KS's $\mathcal{S}_n(\Phi)$ and satisfying some minimal requirement. The new language is given by the following grammar.

Definition 4.5.3 (Modal Language with Reverse Modalities)**.** *Let Φ be a set of primitive propositions. The modal language $\mathcal{L}_n^{+r}(\Phi)$ is given by the following grammar: $\phi, \psi, \ldots \ := \ p \mid \phi \wedge \psi \mid \neg\phi \mid \Box_i\phi \mid \Box_i^{-1}\phi$ where $p \in \Phi$ and $i \in \{1, \ldots, n\}$.* $\square$

The minimal semantic requirement for each $\Box_i^{-1}$ is that, regardless of the interpretation we give to $\Box_i^{-1}\phi$, we should have:

$$\Box_i\Box_i^{-1}\phi \ \Leftrightarrow \ \phi \quad \text{valid in } \mathcal{S}_n(\Phi). \tag{4.5.1}$$

We then say that $\Box_i^{-1}$ is a *right-inverse* modality for $\Box_i$ (by analogy to the notion of right inverse function).

Since $\mathbf{C}[\![\Box_i\phi]\!] = [\ \mathbf{C}[\![\phi]\!]\ ]_i$, we can use the results in the previous sections to derive semantic interpretations for $\Box_i^{-1}\phi$ by using a right inverse $\uparrow_i$ for the space function $[\cdot]_i$ in Definition 4.5.2. Assuming that such a right inverse exists, we can then interpret the reverse modality in $\mathbf{C}$ as

$$\mathbf{C}[\![\Box_i^{-1}\phi]\!] = \uparrow_i(\ \mathbf{C}[\![\phi]\!]\ ). \tag{4.5.2}$$

Since each $\uparrow_i$ is a right inverse of $[\cdot]_i$, it is easy to verify that the interpretation satisfies the requirement in Equation 4.5.1. Furthermore, from Theorem 4.3.1 we can conclude that for each $M \in \mathcal{S}_n(\Phi)$, $\xrightarrow{i}_M$ must necessarily be determinant-complete.

4.5.2 Normal Inverses Modalities

We can choose $\uparrow_i$ in Equation 4.5.2 from the set $\{\uparrow_i^{\mathtt{N}}, \uparrow_i^{\mathtt{MN}}, \uparrow_i^{\mathtt{M}}, \uparrow_{i,s}^{\mathtt{mN}}\}$ of right inverse constructions in Section 4.3.3. Assuming that $\uparrow_i$ is a normal self-map (e.g., either $\uparrow_i^{\mathtt{N}}$, $\uparrow_i^{\mathtt{MN}}$, or $\uparrow_{i,s}^{\mathtt{mN}}$), we can show from Definition 4.2.2 and Equation 4.5.2 that $\Box_i^{-1}$ is itself a normal modal operator in the following sense: (1) If ϕ is valid in $\mathcal{S}_n(\Phi)$ then $\Box_i^{-1}\phi$ is valid in $\mathcal{S}_n(\Phi)$, and (2) $\Box_i^{-1}(\phi \Rightarrow \psi) \Rightarrow (\Box_i^{-1}\phi \Rightarrow \Box_i^{-1}\psi)$ is valid in $\mathcal{S}_n(\Phi)$.

4.5.3 Inconsistency Invariance

We can conclude from Proposition 4.1.1(1) that since we assumed a right inverse of $[\cdot]_i$, we should have

$$\neg\Box_i ff \text{ valid in } \mathcal{S}_n(\Phi). \tag{4.5.3}$$

Indeed using the fact that $[\cdot]_i$ is a normal self-map with an inverse $\uparrow_i$ and Theorem 4.2.1, we can verify the following:

$$\begin{aligned} \mathbf{C}[\![\Box_i ff]\!] &= \mathbf{C}[\![\Box_i(ff \wedge \Box_i^{-1} ff)]\!] \\ &= \mathbf{C}[\![\Box_i(ff) \wedge \Box_i(\Box_i^{-1} ff)]\!] \\ &= \mathbf{C}[\![\Box_i(ff) \wedge ff]\!] = \mathbf{C}[\![ff]\!]. \end{aligned}$$

This implies $\Box_i ff \Leftrightarrow ff$ is valid in $\mathcal{S}_n(\Phi)$ and this amounts to say that $\neg\Box_i ff$ is valid in $\mathcal{S}_n(\Phi)$.

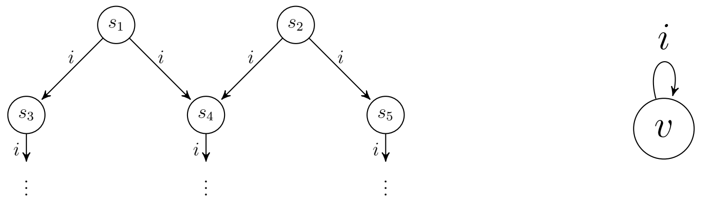
Modal systems such K_n or HM [HM85] where $\neg\Box_i ff$ is not an axiom cannot be extended with a reverse modality satisfying Equation 4.5.1 (without restricting their models). The issue is that the axiom $\neg\Box_i ff$, typically needed in epistemic, doxastic and temporal logics, would require the accessibility relations of agent i to be serial (recall that determinant-complete relations are necessarily serial). In fact $\Box_i ff$ is used in HM logic to express deadlocks w.r.t. to i; $(M, s) \models \Box_i ff$ iff there is no s' such that $s \xrightarrow{i}_M s'$. Clearly there cannot be state deadlocks w.r.t. i if $\xrightarrow{i}_M$ is required to be serial for each M.

4.5.4 Bisimilarity Invariance

Recall that bisimilarity invariance states that bisimilar pointed KS's satisfy the same formulae in $\mathcal{L}_n(\Phi)$. The addition of a reverse modality $\Box_i^{-1}$ may violate this invariance, in the sense that bisimilar pointed KS's may no longer satisfy the same formulae in $\mathcal{L}_n^{+r}(\Phi)$. This can be viewed as saying that the addition of inverse modalities increases the distinguishing power of the original modal language. We prove this fact below as an application of our taxonomy of normal right inverses in Section 4.4.

Let us suppose that the chosen right inverse $\uparrow_i$ in Equation 4.5.2 is *any* normal self-map whatsoever. Now take v and s_4 as in Figure 4.6. Suppose that $\pi_{M_5}(v) = \pi_{M_1}(s_i)$ for every s_i in the states of M_1. Clearly $(M_1, s_4) \cong (M_5, v)$. Since s_4 is multiply determined then from Definition 4.4.2 $(M, s_4) \in \uparrow_i^{\mathtt{MN}}(\mathit{false})$. Using Corollary 4.4.1(2) we obtain $(M, s_4) \in \uparrow_i(\mathit{false})$, and thus $(M_1, s_4) \models \Box_i^{-1}\mathit{ff}$.

Since v is uniquely determined, applying Definition 4.4.4 we conclude that $(M_5, v) \notin \uparrow_{i,\mathfrak{s}}^{\mathtt{mN}}(\mathit{false})$ for any $\mathfrak{s}$. From Corollary 4.4.1(2) it follows that $(M_5, v) \notin \uparrow_i(\mathit{false})$ and thus $(M_5, v) \not\models \Box_i^{-1}\mathit{ff}$.



(a) M_1 (b) M_5

Figure 4.6 – Accessibility relations for an agent i. In each sub-figure we omit the corresponding KS M_K

Thus, from the bounds for normal right inverses provided in this part of the dissertation we have shown that regardless of the normal interpretation of $\Box_i^{-1}$, the formula $\Box_i^{-1}\mathit{ff}$ can uniquely identify determined states from multiply determined ones but bisimilarity cannot.

4.5.5 Temporal Operators

We conclude this section with a brief discussion on some right inverse linear-time modalities. Let us suppose that we have two agents $n = 2$ in our modal language $\mathcal{L}_n(\Phi)$ under consideration. Thus, we can interpret them in Kripke scs as $\mathbf{C} = \mathbf{K}(\mathcal{S}_2(\Phi))$.

Assume further that the intended meaning of the two modalities $\Box_1$ and $\Box_2$ are the *next* operator ($\circledcirc$) and the *henceforth/always* operator ($\Box$), respectively, in a *linear-time* temporal logic. In order to obtain the intended meaning we take $\mathcal{S}_2(\Phi)$ to be the largest set such that: If $M \in \mathcal{S}_2(\Phi)$, where M is a 2-agent KS, the accessibility relation $\xrightarrow{1}_M$ is isomorphic to the successor relation on the natural numbers and $\xrightarrow{2}_M$ is the reflexive and transitive closure of $\xrightarrow{1}_M$. The relation $\xrightarrow{1}_M$ is intended to capture the linear flow of time. Intuitively, $s \xrightarrow{1}_M t$ means that the state t is the only next state for s. Similarly, $s \xrightarrow{2}_M t$, with $s \neq t$, is intended to capture the fact that t is one of the infinitely many future states for s.

Notice that the accessibility relation $\xrightarrow{1}_M$ is determinant-complete, because this is isomorphic to the successor relation, therefore every state is uniquely determining its successor, and by Theorem 4.3.1 there exists a right inverse of $[\cdot]_i$. If we apply Equation 4.5.2 taking as the inverse $\uparrow_1 = \uparrow^{\mathtt{M}}_1$, i.e., the greatest right inverse of $[\cdot]_1$, we obtain $\Box_1^{-1} = \circleddash$, which corresponds to a past modality known in the literature as the *strong* previous operator [PM92]. Formally, the operator $\circleddash$ is given by $(M, t) \models \circleddash\, \phi$ iff there exists s such that $s \xrightarrow{M}_1 t$ and $(M, s) \models \phi$.

If we take $\uparrow_i$ to be the normal right inverse $\uparrow^{\mathtt{N}}_i$, we obtain $\Box_1^{-1} = \widetilde{\circleddash}$ the past modality known as *weak* previous operator [PM92]. The formal definition of the operator $\widetilde{\circleddash}$ is given by $(M, t) \models \widetilde{\circleddash}\, \phi$ iff for every s if $s \xrightarrow{M}_1 t$ then $(M, s) \models \phi$. Notice that the only difference between the two operators is the following: If s is an indetermined/initial state w.r.t. $\xrightarrow{1}_M$ then $(M, s) \not\models \circleddash\, \phi$ and $(M, s) \models \widetilde{\circleddash}\, \phi$ for any ϕ. It follows that $\circleddash$ is not a normal operator, since $\circleddash tt$ is not valid in $\mathcal{S}_2(\Phi)$ but tt is.

Notice that the accessibility relation $\xrightarrow{2}_M$ is not determinant-complete: Because there are not determinant states in the structure. Take any increasing chain $s_0 \xrightarrow{1}_M s_1 \xrightarrow{1}_M \ldots$ Then, state s_1 is not determinant because for every s_j such that $s_1 \xrightarrow{2}_M s_j$ we also have $s_0 \xrightarrow{2}_M s_j$. Theorem 4.3.1 tell us that there is no right inverse $\uparrow_2$ of $[\cdot]_i$ that can give us an operator $\Box_2^{-1}$ satisfying Equation 4.5.1 .

By analogy to the above-mentioned past operators, one may think that the past operator *it-has-always-been* $\boxminus$ [RS97] may provide a reverse modality for $\Box$ in the sense of Equation 4.5.1. The operator is given by $(M, t) \models \boxminus\phi$ iff $(M, s) \models \phi$ for every s such that $s \xrightarrow{2}_M t$. $\Box \boxminus \phi \;\Rightarrow\; \phi$ is valid in $\mathcal{S}_2(\Phi)$ but $\phi \Rightarrow\; \Box \boxminus\; \phi$ is not.

4.6 Summary

We gave an algebraic counter-part of the notion of normality from modal logic: A self-map is normal if and only if preserves finite suprema. We then studied the existence and derivation of right inverses (extrusions) of space functions for the Kripke spatial constraint systems. We showed that being *determinant-complete* is the weakest condition on KS's that guarantees the existence of such right inverses. We identified the greatest normal right inverse as well as all minimal right inverses of any given space function. To illustrate our results we applied them to modal logic by using space functions and their right inverses as the semantic counterparts of box modalities and their right inverse modalities. We discussed their implications in the context of modal concepts such as bisimilarity invariance, inconsistency invariance and temporal modalities.

Part III

Knowledge in Terms of Space

Five

Introduction

In this chapter we shall propose an algebraic treatment of the epistemic notion of *knowledge* by using scs's as the framework of reference. We shall make use of some notions and concepts in modal logics that were introduced in past sections.

As we mentioned before, an important question in epistemology has been whether the epistemic notion of knowledge is definable in terms of belief. In [HSS09] the authors have addressed this question having as the framework of reference the epistemic and doxastic logics. They proved under which circumstances epistemic logic $S5$ cannot (can) be explicitly (implicitly) defined in terms of the belief logic $KD45$.

We shall show in this part of the thesis that scs's besides capturing the notion of belief, can also be used to derive the epistemic notion of knowledge. The following describes the main contribution of this chapter.

- *Knowledge in Terms of Global Space.* We shall represent knowledge by using a derived spatial operation that expresses global information. The new representation is shown to obey the epistemic principles of the logic for knowledge $S4$. We also show a sound and complete spatial constraint systems interpretation of $S4$ formulae. In previous work [KPPV12] spatial constraint systems were required to satisfy additional properties in order to capture $S4$ knowledge. Namely, space functions had to be closure operators. Here we will show that $S4$

knowledge can be captured in spatial constraint systems without any further requirements.

Organization.

This part of the dissertation is structured as follows. In Section 5.1 we shall propose an algebraic characterization of the epistemic notion of *S4 knowledge* by means of a derived operator from the underlying scs. Finally, in Section 5.2 we shall present some concluding remarks and related work.

5.1 S4 Knowledge as Global Information

In [HPRV15] the authors have shown how spatial constraint systems can be used to represent epistemic concepts such as *beliefs, lies* and *opinions*. In this section we shall show that spatial constraint systems can also be used to represent the epistemic concept of knowledge by means of the global information operator given in Definition 2.5.2.

5.1.1 Knowledge Constraint System.

In [KPPV12] the authors extended the notion of spatial constraint system to account for *knowledge*. The idea there was to *impose* the knowledge axioms over the spatial constraint systems. We shall refer to the extended notion in [KPPV12] as *S4 constraint systems* since it is meant to capture the epistemic logic for knowledge S4. Roughly speaking, one may wish to use $[c]_i$ to represent not only some information c that agent i has but rather a *fact* that he knows. Given the domain theoretical nature of constraint systems, it allows for a rather simple and elegant characterization of knowledge by requiring space functions to be *Kuratowski closure operators* [MT44]: i.e., monotone, extensive and idempotent functions that preserve bottom and lubs.

We now introduce the notion of *Knowledge constraint system* proposed in [KPPV12] in order to capture the S4 knowledge behaviour by using spatial constraint systems. We shall use this definition as a reference for the characterization we shall propose.

Definition 5.1.1 (Knowledge Constraint System (s4cs) [KPPV12]). *An n-agent* S4 constraint system (n-s4cs) $\boldsymbol{C}$ *is a scs whose space functions* $[\cdot]_1, \ldots, [\cdot]_n$ *are also* closure operators. *Thus, in addition to S.1 and S.2 in Definition 2.5.1, each* $[\cdot]_i$ *also satisfies:*

EP.1 $[c]_i \sqsupseteq c$ *and*

EP.2 $[[c]_i]_i = [c]_i$.

□

Intuitively, in an n-s4cs, $[c]_i$ states that the agent i has knowledge of c in its store $[\cdot]_i$. The axiom EP.1 says that if agent i knows c then c must hold, hence $[c]_i$ has at least as much information as c. The epistemic principle that an agent i is aware of its own knowledge (*the agent knows what he knows*) is realized by EP.2. Also the epistemic assumption that agents are *idealized reasoners* follows from the monotonicity of space functions (Remark 2.5.1); for if c is a consequence of d ($d \sqsupseteq c$) then if d is known to agent i, so is c, $[d]_i \sqsupseteq [c]_i$.

Recall that modal logics are interpreted over families of Kripke structures (Definition 3.1.2) obtained by restricting their accessibility relations. We use $\mathcal{M}_n(\Phi)$ to denote the set of *all* Kripke structures over the set of primitive propositions Φ (Notation 4.3.1). We shall use $\mathcal{M}_n^{rt}(\Phi)$ to denote the set of those n-agents Kripke structures, over the set of primitive propositions Φ, whose accessibility relations are *reflexive* and *transitive*. As in the previous chapter, for notational convenience, we take the set Φ of primitive propositions and n to be fixed from now on and omit them often from the notation. E.g., we write $\mathcal{M}^{rt}$ instead of $\mathcal{M}_n^{rt}(\Phi)$.

Henceforth we use $\mathbf{C}^{rt}$ to denote the Kripke constraint system $\mathbf{K}(\mathcal{M}^{rt})$ (Definition 4.3.1). In [KPPV12] it was shown that $\mathbf{C}^{rt}$ is in fact an S4 constraint system.

Proposition 5.1.1 ([KPPV12]). $\boldsymbol{C}^{rt}$ *is an* s4cs.

Recall our interpretation of formulae of the modal language $\mathcal{L}_n(\Phi)$ (Definition 3.1.1) using Kripke spatial constraint systems (Definition 4.5.2). In particular the interpretation of the formula $\Box_i(\phi)$ in $\mathbf{C}^{rt}$, denoted as $\mathbf{C}^{rt}[\![\Box_i(\phi)]\!]$, is given by the constraint $[\mathbf{C}^{rt}[\![\phi]\!]]_i$. Let us now recall the notion of validity in the modal logic $S4$ [Hin62].

Definition 5.1.2. *Let ϕ be a modal formula from the modal language $\mathcal{L}_n(\Phi)$. The formula ϕ is said to be $S4$-valid iff for every (M, s) where $M \in \mathcal{M}^{rt}(\Phi)$ and s is a state of M, we have $(M, s) \models \phi$.* □

Notation 5.1.1. *In the modal logic $S4$, the box modality $\Box_i$ is often written as K_i. The formula $K_i(\phi)$ specifies that agent i knows ϕ.*

The following proposition from [KPPV12] is an immediate consequence of the above definition. It states the correctness w.r.t. validity of the interpretation of $S4$ formulae in $\mathbf{C}^{rt}$.

Proposition 5.1.2 ([KPPV12])**.** *$\mathbf{C}^{rt}[\![\phi]\!] = true$ if and only if ϕ is $S4$-valid.*

The above gives a brief summary of the use in [KPPV12] of *Kuratowski closure operators* $[c]_i$ to capture knowledge. In what follows we show an alternative interpretation of knowledge as the global construct $[\![c]\!]_G$ in Definition 2.5.2.

5.1.2 Knowledge as Global Information.

In last section we saw how spatial constraint systems can be used in order to capture the epistemic notion of $S4$ Knowledge by imposing its axioms to be hold over the spatial constraint system. In this section we shall show that we can in fact obtain a representation of this epistemic notion of $S4$ Knowledge by deriving it directly from the space functions of the spatial constraint system. The main idea is that if the space functions of the underlying scs are *continuous*, then we can *derive* another constraint system in which the space functions are in fact Kuratowski closure operators and thus satisfy the $S4$ axioms.

Recall the definition of the global information operator ($\llbracket\cdot\rrbracket_{\{i\}}$) given in Definition 2.5.2. From that definition we obtain the following equation:

$$\llbracket c\rrbracket_{\{i\}} = c \sqcup [c]_i \sqcup [[c]_i]_i \sqcup [[[c]_i]_i]_i \sqcup \ldots = c \sqcup [c]_i \sqcup [c]_i^2 \sqcup [c]_i^3 \sqcup \ldots = \bigsqcup_{j=0}^{\infty} [c]_i^j \tag{5.1.1}$$

where $[e]_i^0 \stackrel{\text{def}}{=} e$ and $[e]_i^{j+1} \stackrel{\text{def}}{=} [\, [e]_i^j \,]_i$. For the sake of simplicity we shall use $\llbracket\cdot\rrbracket_i$ as an abbreviation of $\llbracket\cdot\rrbracket_{\{i\}}$.

Intuitively, $\llbracket c\rrbracket_i$ says that c holds *globally* w.r.t. i: c holds outside and in every nested space of agent i. The idea then is to show that the global function $\llbracket c\rrbracket_i$ can also be used to represent the fact that an agent i knows the information c.

In order to show that the global function $\llbracket c\rrbracket_i$ can be used to represent the knowledge of c by agent i, we shall show that it satisfies the conditions for being called a Kuratowski closure operator. The fact that $\llbracket\cdot\rrbracket_i$ is a Kuratowski closure operator implies that it satisfies the epistemic axioms EP.1 and EP.2 above: It is easy to see that $\llbracket c\rrbracket_i$ satisfies $\llbracket c\rrbracket_i \sqsupseteq c$ (EP.1). Under certain natural assumption we shall see that it also satisfies $\llbracket\llbracket c\rrbracket_i\rrbracket_i = \llbracket c\rrbracket_i$ (EP.2). Furthermore, we can combine knowledge with our belief interpretation of space functions: Clearly, $\llbracket c\rrbracket_i \sqsupseteq [c]_i$ holds for any c. This reflects the epistemic principle that *whatever is known is also believed* [Hin62].

We now show that any *spatial constraint system* with continuous space functions $[\cdot]_1, \ldots, [\cdot]_n$ induces an s4cs with space functions $\llbracket\cdot\rrbracket_1, \ldots \llbracket\cdot\rrbracket_n$. This means that we can built a new spatial constraint system having *global* functions (therefore satisfying $S4$ Knowledge axioms) as its space functions directly from any spatial constraint system having *continuous* functions as its space functions.

Definition 5.1.3. *Given an scs* $\mathcal{C} = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$*, we use* $\mathcal{C}^*$ *to denote the tuple* $(Con, \sqsubseteq, \llbracket\cdot\rrbracket_1, \ldots, \llbracket\cdot\rrbracket_n)$*.* □

One can show that $\mathcal{C}^*$ is also a spatial constraint system. Furthermore it is an s4cs as stated next.

Theorem 5.1.1. *Let* $\mathcal{C} = (Con, \sqsubseteq, [\cdot]_1, \ldots, [\cdot]_n)$ *be a spatial constraint system. If* $[\cdot]_1, \ldots, [\cdot]_n$ *are continuous functions then* $\mathcal{C}^*$ *is an n-agent s4cs.*

Proof. We need to show that each $[\![\cdot]\!]_i$ satisfies S.1, S.2, EP.1 and EP.2.

- We prove that $[\![\cdot]\!]_i$ satisfies S.1: $[\![true]\!]_i = true$. Since $[\cdot]_i$ satisfies S.1 we can use $[true]_i = true$ to show, by induction on j, that $[true]_i^j = true$ for any j. Then from Equation 5.1.1 we conclude $[\![true]\!]_i = \bigsqcup\{true\} = true$ as wanted.

- We prove that $[\![\cdot]\!]_i$ satisfies S.2: $[\![c \sqcup d]\!]_i = [\![c]\!]_i \sqcup [\![d]\!]_i$. Since $[\cdot]_i$ satisfies S.2 we can use $[c \sqcup d]_i = [c]_i \sqcup [d]_i$ to show by induction on j that $[c \sqcup d]_i^j = [c]_i^j \sqcup [d]_i^j$ for any j. We then obtain the following equation
$$[\![c \sqcup d]\!]_i = \bigsqcup_{j=0}^{\infty}[c \sqcup d]_i^j = \bigsqcup_{j=0}^{\infty}([c]_i^j \sqcup [d]_i^j). \tag{5.1.2}$$
Clearly $[\![c]\!]_i = \bigsqcup_{j=0}^{\infty}[c]_i^j \sqsubseteq \bigsqcup_{j=0}^{\infty}([c]_i^j \sqcup [d]_i^j) \sqsupseteq \bigsqcup_{j=0}^{\infty}[d]_i^j = [\![d]\!]_i$. Therefore $[\![c \sqcup d]\!]_i \sqsupseteq [\![c]\!]_i \sqcup [\![d]\!]_i$. It remains to prove $[\![c]\!]_i \sqcup [\![d]\!]_i \sqsupseteq [\![c \sqcup d]\!]_i$.

  Notice that $[\![c]\!]_i \sqcup [\![d]\!]_i = (\bigsqcup_{j=0}^{\infty}[c]_i^j) \sqcup (\bigsqcup_{j=0}^{\infty}[d]_i^j)$ is an upper bound of the set $S = \{[c]_i^j \sqcup [d]_i^j \mid j \geq 0\}$. Therefore $[\![c]\!]_i \sqcup [\![d]\!]_i \sqsupseteq \bigsqcup S = \bigsqcup_{j=0}^{\infty}([c]_i^j \sqcup [d]_i^j) = [\![c \sqcup d]\!]_i$ as wanted.

- We prove that $[\![\cdot]\!]_i$ satisfies EP.1: $[\![c]\!]_i \sqsupseteq c$. Immediate consequence of Equation 5.1.1.

- Finally we prove that $[\![\cdot]\!]_i$ satisfies EP.2: $[\![[\![c]\!]_i]\!]_i = [\![c]\!]_i$. Since $[\![\cdot]\!]_i$ satisfies EP.1 we have $[\![[\![c]\!]_i]\!]_i \sqsupseteq [\![c]\!]_i$. It remains to prove that $[\![c]\!]_i \sqsupseteq [\![[\![c]\!]_i]\!]_i$.

  Let $S = \{[c]_i^k \mid k \geq 0\}$. Notice that $[\![c]\!]_i = \bigsqcup S$. One can verify from the definition of $[\cdot]_i^j$ that for any j
$$[S]_i^j = \{[[c]_i^k]_i^j \mid k \geq 0\} = \{[c]_i^{k+j} \mid k \geq 0\} \subseteq S$$
  From the continuity of $[\cdot]_i$, one can show by induction on j the continuity of $[\cdot]_i^j$ for any j. It then follows that $[\bigsqcup S]_i^j = \bigsqcup[S]_i^j$ for any j. Since for any j, $[S]_i^j \subseteq S$ we conclude that for every j, $\bigsqcup S \sqsupseteq [S]_i^j$. Therefore $\bigsqcup S \sqsupseteq \bigsqcup_{j=0}^{\infty}[S]_i^j$. This concludes the proof since from Equation 5.1.1 $[\![c]\!]_i = \bigsqcup S$ and $[\![[\![c]\!]_i]\!]_i = \bigsqcup_{j=0}^{\infty}[\bigsqcup S]_i^j = \bigsqcup_{j=0}^{\infty}[S]_i^j$.

□

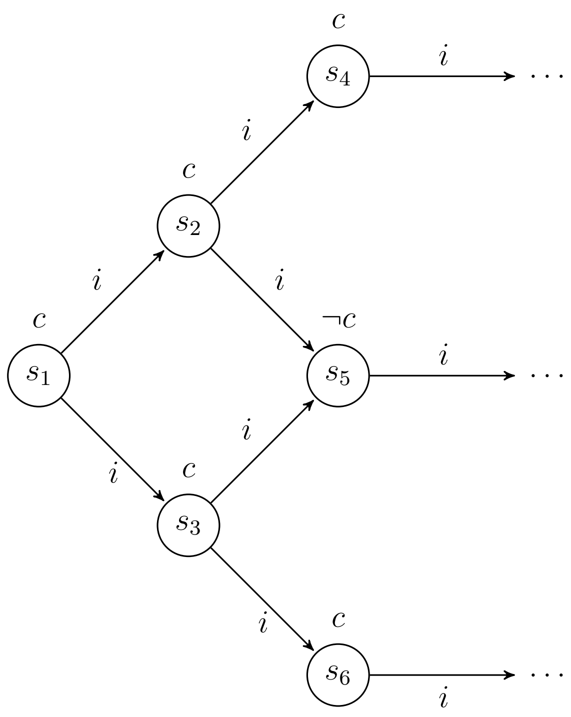


Figure 5.1 – Illustration of a serial Kripke structure M violating the $S4$ Knowledge axioms

The following example illustrates the use of the global operator to capture the notion of $S4$ Knowledge.

Example 5.1.1. *In Figure 5.1 we present a serial Kripke structure M violating the $S4$ Knowledge axioms. Notice that states s_3 and s_2 do not satisfy Axiom EP.1, because $s_3 \xrightarrow{i}_M s_5, s_2 \xrightarrow{i}_M s_5$ and $s_5 \not\models c$. Also, notice that state s_1 violates Axiom EP.2, because $s_1 \xrightarrow{i}_M s_2$ and $s_1 \xrightarrow{i}_M s_3$. Therefore s_1 satisfies Axiom EP.1 but neither s_3 nor s_2 can satisfy it.* □

Consider Example 5.1.1. From Equation 5.1.1 we know that the global operator makes a given piece of information c available everywhere, i.e. c would hold in every state of the Kripke structure. Therefore we would not have the problem of not satisfying the Knowledge axioms as in Example 5.1.1.

The next step is to prove that $S4$ Knowledge can be captured using the global interpretation of space. In order to do so from now on $\mathbf{C}$ denotes the Kripke constraint system $\mathbf{K}(\mathcal{M})$ (Definition 4.3.1). Notice that unlike $\mathbf{C}^{rt}$, constraints in $\mathbf{C}$, and consequently also in $\mathbf{C}^*$, are sets of *unrestricted* (pointed) Kripke structures. It is very important to notice that although $\mathbf{C}$ *is not an S4 constraint system*, from the above theorem, its induced scs $\mathbf{C}^*$ *is*. The reason is because the space functions in $\mathbf{C}^*$ are global functions and we already know that they satisfy the $S4$ Knowledge axioms. Furthermore, just like $\mathbf{C}^{rt}$, we can give in $\mathbf{C}^*$ a sound and complete compositional interpretation of formulae in $S4$.

We now define the compositional interpretation in our constraint system $\mathbf{C}^*$ of modal formulae. Notice that $\mathbf{C}^*$ is a powerset ordered by reversed set inclusion, hence it is a frame (Remark 4.2.1). Recall our definition of the negation constraint $\sim c$ for frames (Definition 4.2.1). Then, the interpretation of modal formula in $\mathbf{C}^*$ is given as follows.

Definition 5.1.4. *Let ϕ be a modal formula from the modal language $\mathcal{L}_n(\Phi)$. The interpretation of ϕ in $\mathbf{C}^*$ is inductively defined as follows:*

$$\begin{aligned} \mathbf{C}^*[\![p]\!] &= \{(M,s) \in \Delta |\ \pi_M(s)(p) = 1\ \} \\ \mathbf{C}^*[\![\phi \wedge \psi]\!] &= \mathbf{C}^*[\![\phi]\!] \sqcup \mathbf{C}^*[\![\psi]\!] \\ \mathbf{C}^*[\![\neg\phi]\!] &= \sim \mathbf{C}^*[\![\phi]\!] \\ \mathbf{C}^*[\![\Box_i\phi]\!] &= [\![\ \mathbf{C}^*[\![\phi]\!]\ ]\!]_i \end{aligned}$$

where Δ is the set of all pointed Kripke structures (M,s) such that $M \in \mathcal{M}_n(\Phi)$. □

Notice that $\Box_i\phi$ is interpreted in terms of the global operator. Since $\mathbf{C}^*$ is a powerset ordered by reversed inclusion the least upper bound is given by set intersection. Thus, from Equation 5.1.1

$$\mathbf{C}^*[\![\Box_i\phi]\!] = [\![\ \mathbf{C}^*[\![\phi]\!]\ ]\!]_i = \bigsqcup_{j=0}^{\infty}[\mathbf{C}^*[\![\phi]\!]]_i^j = \bigcap_{j=0}^{\infty}[\mathbf{C}^*[\![\phi]\!]]_i^j \tag{5.1.3}$$

In particular, notice that from Theorem 5.1.1 and Axiom EP.2 it follows that $\mathbf{C}^*[\![\Box_i\phi]\!] = \mathbf{C}^*[\![\Box_i(\Box_i\phi)]\!]$ as expected for an $S4$-knowledge modality; i.e., if agent i knows ϕ he knows that he knows it.

We conclude this section with the following theorem stating the correctness w.r.t. validity of the interpretation of $S4$ Knowledge as global operator.

Theorem 5.1.2. *$\mathbf{C}^*[\![\phi]\!] = true$ if and only if ϕ is $S4$-valid.*

Proof. Let Δ be the set of all pointed Kripke structures (M, s) such that $M \in \mathcal{M}$. Similarly, let Δ^{rt} be the set of all pointed Kripke structures (M, s) such that $M \in \mathcal{M}^{rt}$. Given $M \in \mathcal{M}$, we use $M^* \in \mathcal{M}^{rt}$ to denote the Kripke structure that results from M by replacing its accessibility relations with their corresponding transitive and reflexive closure.

From Definitions 4.3.1 and 5.1.3 we conclude that $\Delta = true$ in $\mathbf{C}^*$ and $\Delta^{rt} = true$ in $\mathbf{C}^{rt}$. From Definitions 4.3.1 and Proposition 5.1.2, it suffices to prove that

$$\mathbf{C}^*[\![\phi]\!] = \Delta \text{ if and only if } \mathbf{C}^{rt}[\![\phi]\!] = \Delta^{rt} \tag{5.1.4}$$

Property 5.1.4 is a corollary of the following two properties:

$$\text{For all } (M, s) \in \Delta^{rt} : \; (M, s) \in \mathbf{C}^{rt}[\![\phi]\!] \text{ if and only if } (M, s) \in \mathbf{C}^*[\![\phi]\!] \tag{5.1.5a}$$

$$\text{For all } (M, s) \in \Delta : \; (M, s) \in \mathbf{C}^*[\![\phi]\!] \text{ if and only if } (M^*, s) \in \mathbf{C}^*[\![\phi]\!] \tag{5.1.5b}$$

Proof of 5.1.5a Let $(M, s) \in \Delta^{rt}$. We proceed by induction on the size of ϕ. The base case $\phi = p$ is trivial. For the inductive step here we show the most interesting case: $\phi = \Box_i \psi$ (the other cases follow directly from the induction hypothesis and the compositionality of the interpretations).

($\Rightarrow$) Assume $(M, s) \in \mathbf{C}^{rt}[\![\Box_i \psi]\!]$. From Equation 5.1.3 we want to prove that $(M, s) \in \bigcap_{j=0}^{\infty} [\mathbf{C}^*[\![\psi]\!]]_i^j$. Take an arbitrary sequence $s_1, s_2, \ldots$ such that $s = s_0 \xrightarrow{i}_M s_1 \xrightarrow{i}_M s_2 \xrightarrow{i}_M \ldots$. From Definition 4.3.1 and Equation 5.1.3 it suffices to show that $(M, s_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$. From the assumption and Definition 4.3.1 we know that for every t such that $s \xrightarrow{i}_M t$ we have $(M, t) \in \mathbf{C}^{rt}[\![\psi]\!]$. From the assumption we also know that $\xrightarrow{i}_M$ is transitive and reflexive: We thus conclude that $(M, s_k) \in \mathbf{C}^{rt}[\![\psi]\!]$ for

$k = 0, 1, \ldots$. From the induction hypothesis, we derive $(M, s_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$.

($\Leftarrow$) Assume $(M, s) \in \mathbf{C}^*[\![\phi]\!]$. From Equation 5.1.3, we know that $(M, s) \in \bigcap_{j=0}^{\omega}[\mathbf{C}^*[\![\psi]\!]]_i^j$. Then $(M, s) \in [\mathbf{C}^*[\![\psi]\!]]_i$. From Definition 4.3.1, for every t such that $s \xrightarrow{i}_M t$ we have $(M, t) \in \mathbf{C}^*[\![\psi]\!]$. From the induction hypothesis, we can conclude that $(M, t) \in \mathbf{C}^{rt}[\![\psi]\!]$ for every t such that $s \xrightarrow{i}_M t$. This shows that $(M, s) \in [\mathbf{C}^{rt}[\![\psi]\!]]_i$ since $[\mathbf{C}^{rt}[\![\psi]\!]]_i = \mathbf{C}^{rt}[\![\phi]\!]$.

Proof of 5.1.5b Let $(M, s) \in \Delta$. We proceed by induction on the size of ϕ. The base case $\phi = p$ is trivial. For the inductive step, we show the case $\phi = \Box_i\psi$ (as in the previous proof the other cases follow directly from the induction hypothesis and the compositionality of the interpretations).

($\Rightarrow$) Assume $(M, s) \in \mathbf{C}^*[\![\phi]\!]$. Take an arbitrary sequence $s_1, s_2, \ldots$ such that $s = s_0 \xrightarrow{i}_{M^*} s_1 \xrightarrow{i}_{M^*} s_2 \xrightarrow{i}_{M^*} \ldots$. From Equation 5.1.3 it suffices to show that $(M^*, s_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$. Notice that $\xrightarrow{i}_{M^*}$ is the transitive and reflexive closure of $\xrightarrow{i}_M$, thus we have $s(\xrightarrow{i}_{M^*})^* s_k$ if and only if $s \xrightarrow{i}_{M^*} s_k$. Consequently, let us take an arbitrary t such that $s \xrightarrow{i}_{M^*} t$: It is sufficient to show that $(M^*, t) \in \mathbf{C}^*[\![\psi]\!]$. Since $\xrightarrow{i}_{M^*}$ is the transitive and reflexive closure of $\xrightarrow{i}_M$, there must exist $s = t_0 \xrightarrow{i}_M t_1 \xrightarrow{i}_M t_2 \xrightarrow{i}_M \ldots$ such that $t_j = t$ for some $j \geq 0$. From the assumption and Equation 5.1.3 we have $(M, s) \in \bigcap_{j=0}^{\omega}[\mathbf{C}^*[\![\psi]\!]]_i^j$. Thus for any $t_1, t_2, \ldots$ such that $s = t_0 \xrightarrow{i}_M t_1 \xrightarrow{i}_M t_2 \xrightarrow{i}_M \ldots$ we have $(M, t_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$. We conclude that $(M, t) \in \mathbf{C}^*[\![\psi]\!]$, and thus from the induction hypothesis we obtain $(M^*, t) \in \mathbf{C}^*[\![\psi]\!]$.

($\Leftarrow$) Assume $(M^*, s) \in \mathbf{C}^*[\![\phi]\!]$. Take an arbitrary sequence $s_1, s_2, \ldots$ such that $s = s_0 \xrightarrow{i}_M s_1 \xrightarrow{i}_M s_2 \xrightarrow{i}_M \ldots$. From Equation 5.1.3, it suffices to show that $(M, s_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$. Notice that $s \xrightarrow{i}_{M^*} s_k$ for $k = 0, 1, \ldots$ since $\xrightarrow{i}_{M^*}$ is the transitive and reflexive closure of $\xrightarrow{i}_M$. From the assumption and Equation 5.1.3 we know that for every t such that $s \xrightarrow{i}_{M^*} t$ we have $(M^*, t) \in \mathbf{C}^*[\![\psi]\!]$. We then conclude that for $k = 0, 1, \ldots$, $(M^*, s_k) \in \mathbf{C}^*[\![\psi]\!]$. We use the induction hypothesis to conclude that $(M, s_k) \in \mathbf{C}^*[\![\psi]\!]$ for $k = 0, 1, \ldots$.

$\Box$

5.2 Summary

In this chapter we have shown how to represent the epistemic notion of $S4$ knowledge by using spatial constraint systems. We have shown that for any spatial constraint system in which their space functions are continuous then we can derive a constraint system in which the space functions are the global operators. We have shown that the global operators satisfies the Kuratowski closure operator axioms, therefore they satisfy the $S4$ Knowledge axioms. We also showed that our characterization is complete w.r.t. $S4$ knowledge validity.

Part IV

Distributed Information in Terms of Space

Six

Introduction

In this part of the dissertation we shall propose an approach to characterize the notion of *Distributed Information* in terms of spatial constraint systems.

In the analysis of multi-agent scenarios one might be interested in knowing whether the distributed and epistemic nature of these kind of systems rise some unwanted behaviour. The potential communication of the agents in the systems makes even more difficult the analysis of programs than in the single-agent case. In epistemology, this potential way of communication is known as *distributed knowledge* [FHMV95]. Intuitively, in distributed knowledge we might wonder whether a group of agents G knows a certain piece of information c whenever they combine the individual knowledge they have. Consider the situation in which a given agent i knows the information c and agent j knows the information $c \rightarrow d$. Then, if agents i and j where to communicate the knowledge they have then they would know d even though none of them know d individually.

As previously introduced, the authors in [FHMV95] propose a modal language for the analysis of a group of agents in a system. The modalities in the language represent situations in which *every* agent in the group knows a fact (known as *common knowledge*) and situations in which a fact is known by *combining* the information owned individually by the agents of a group (*distributed knowledge*).

In this part of the dissertation we shall give an algebraic characterization of the notion of distributed information. We shall define the notion of *distributed space* as an operator aiming to hold the information that is distributed among the (local) spaces of the member of a group of agents. Therefore, given a group of agents G, the *distributed information* of a subgroup $I \subseteq G$ can be intuitively defined as the collective/aggregate information from all its members.

The main contributions in this chapter can be summarized as follows

- We extend the definition of spatial constraint systems in order to account for an infinite number of agents.
- We show that the notion of *distributed information* can be characterized in scs's by means of operators over the underlying scs.
- We show that if an *infinite group* of agents G have distributed information of a certain piece of information c, then there exists a *finite subgroup* H of G which also has distributed information of c.

Organization.

This part of the dissertation is structured as follows. In Section 6.1 we shall give some formal definitions of the notion of *Galois connection* and its properties. Then, in Section 6.2 we shall adapt the definition of spatial constraint systems to account for an infinite number of agents. Then, in Section 6.3 we shall propose an algebraic characterization of the epistemic notion of *distributed information*. Finally, in Section 6.4 we shall present some concluding remarks.

6.1 Background

In this part of the dissertation we shall make use of the notions of constraint system, spatial constraint systems and spatial constraint systems with extrusion introduced in Sections 2.3.2,2.4 and 2.5 respectively. Also, we shall give some background on *Galois connections*, which is an important notion for our results in this part of the dissertation.

Intuitively, a Galois connection can be defined as the way in which two functions transform objects in the two direction between two different "worlds" [DEW13]. These worlds can represented through posets, lattices, etc, and the transforming functions would depend on the definitions of these worlds. An important feature over the functions is that if we transform an element of a world into the other world, then if we transform the same object back to the starting world a certain stability would be reached. As a consequence, if we apply three transformations to a given object in one of the worlds then the result would be the same than applying just one transformation to the same object. Formally, a Galois connection can be defined as follows.

Definition 6.1.1 (Galois connection)**.** *Consider posets* $A = (S_1, \sqsubseteq)$ *and* $B = (S_2, \sqsubseteq)$ *with maps* $\alpha : A \to B$ *and* $\beta : B \to A$*. We say that* α *and* β *form a* Galois connection*, denoted* (α, β)*, if* $\forall_{a,b}$ *with* $a \in A, b \in B$

$$\alpha(a) \sqsubseteq b \text{ iff } a \sqsubseteq \beta(b).$$

Where α *is known as the* left adjoint *and* β *is known as the* right adjoint*.*

□

We now introduce some equivalent definitions for Galois connections (also known as *adjunctions*).

Proposition 6.1.1. *[AGM95] Let* $A = (S_1, \sqsubseteq)$ *and* $B = (S_2, \sqsubseteq)$ *be posets and* $\alpha : A \to B$ *and* $\beta : B \to A$ *be monotone functions. Then the following are equivalent:*

1. *The right adjoint* β uniquely determines *the left adjoint* α*, i.e.* $\alpha(x) = Min\ \{y \in A \mid x \sqsubseteq \beta(y)\}$
2. *The left adjoint* α uniquely determines *the right adjoint* β*, i.e.* $\beta(y) = Max\ \{x \in B \mid \alpha(x) \sqsubseteq y\}$
3. $\alpha \circ \beta \sqsubseteq id_B$ *and* $\beta \circ \alpha \sqsupseteq id_A$*,*
4. $\forall x \in A, \forall y \in B.\ (x \sqsubseteq \beta(y) \Leftrightarrow \alpha(x) \sqsubseteq y)$*.*

Where id_A represents the identity function on a set A.

We now present some important properties of Galois connections that we shall use in what follows in this part of the dissertation.

Proposition 6.1.2. *[AGM95] Let $A = (S_1, \sqsubseteq)$ and $B = (S_2, \sqsubseteq)$ be two posets. For any Galois connection (α, β) where $\alpha : A \to B$ and $\beta : B \to A$ then:*

1. $\beta \circ \alpha \circ \beta = \beta$ *and* $\alpha \circ \beta \circ \alpha = \alpha$
2. *The image of β and the image of α are order-isomorphic. The isomorphisms are given by the restrictions of β and α to $im(\alpha)$ and $im(\beta)$, respectively.*
3. β *is surjective* $\Leftrightarrow \beta \circ \alpha = id_A \Leftrightarrow \alpha$ *is injective,*
4. α *is surjective* $\Leftrightarrow \alpha \circ \beta = id_B \Leftrightarrow \beta$ *is injective,*
5. *The left adjoint α* preserves least upper bounds *(suprema), the right adjoint β* preserves greatest lower bounds *(infima).*

Where id_A represents the identity function on a set A.

The proofs of Propositions 6.1.1 and 6.1.2 can be found in [AGM95].

6.2 Spatial Constraint Systems

As introduced in Section 2.5, the authors in [KPPV12] extended the notion of cs to account for distributed and multi-agent scenarios where agents have their own space for local information and for performing their computations. Here we shall adapt the notion of scs to account for an arbitrary, possibly infinite, set of agents.

Definition 6.2.1 (Indexed Spatial Constraint System)**.** *Given a cs $(Con, \sqsubseteq)$, a* space function *is a continuous self-map f on Con such that (S.1) $f(true) = true$ and (S.2) $f(c \sqcup d) = f(c) \sqcup f(d)$ for all $c, d \in Con$. A*

(G-indexed) spatial constraint system (scs) *is a cs equipped with a (possibly infinite) indexed family* $\{[\cdot]_i\}_{i\in G}$ *of space functions.*

We shall use $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$ *to denote a scs* $\mathbf{C} = (Con, \sqsubseteq)$ *with space functions* $\{[\cdot]_i\}_{i\in G}$*. We shall refer to the index set* G *as the group of agents of* $\mathbf{C}$ *and to* $[\cdot]_i$ *as the* space function *of agent* i*. We shall often omit* G *when understood from the context. Subsets of* G *are referred to as sub-groups of* G*.* □

Property S.1 says that space functions are strict maps (i.e bottom preserving). Intuitively, it states that having an empty local space amounts to nothing. Property S.2 states that space functions preserve (finite) lubs and it allows us to join and distribute the local information of agent i. As presented in Remark 2.5.1 it follows from S.2 that space functions are monotonic. The epistemic intuition is that if agent i has c in its space and d can be derived from c then agent i can also derive d in its space.

Continuity and Arbitrary Number of Agents. In this presentation of spatial constraint systems we are assuming that space functions are continuous and that the set of agents can be infinite.

Remark 6.2.1 (Spatial Inconsistency, Indistinguishability)**.** *Let us discuss briefly some important spatial and epistemic behaviours that are allowed to occur in space functions. Namely, inconsistent confinement, freedom of opinion, and indistinguishability. In a scs nothing prevents us from having* $[\mathit{false}]_i \neq \mathit{false}$*. Intuitively, inconsistencies generated by an agent may be confined within its own space (inconsistency confinement). It is also possible to have* $[c]_i \sqcup [d]_j \neq \mathit{false}$ *even when* $c \sqcup d = \mathit{false}$*; i.e. we may have agents whose information is inconsistent with that of others. This reflects the distributive nature of the agents as they may have different information about the same fact. It also reflects their epistemic nature as agents may have different beliefs (freedom of opinion). Analogous to inconsistency confinement, we could have* $[c]_i = [d]_i$ *for* $c \neq d$*. Depending on the intended model this could be interpreted as saying that agent* i *cannot distinguish* c *from* d

(indistinguishability). We shall say that c and d are indistinguishable *by agent i, written $c \cong_i d$, exactly when $[c]_i = [d]_i$.*

6.2.1 Extrusion and utterance.

As presented in Section 2.6, we can also equip an agent i with an *extrusion* function $\uparrow_i : Con \to Con$. We shall extend the definition of scs's with extrusion in order to account for an infinite number of agents.

Definition 6.2.2 (Extrusion)**.** *Let* **C** *be a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$. We say that $\uparrow_i$, where $i \in G$, is an extrusion function for* **C** *iff $\uparrow_i$ is a right inverse of $[\cdot]_i$, i.e., iff (E.1) for every $c \in Con$, $[\ \uparrow_i c\ ]_i = c$. We use $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G}, \{\uparrow_j\}_{j\in H})$ where $H \subseteq G$ to denote a G-indexed scs with extrusion functions $\{\uparrow_j\}_{j\in H}$.* □

As we mentioned early in this manuscript, using Properties E.1 and S.2 one can verify $[c \sqcup \uparrow_i d]_i = [c]_i \sqcup d$. From a spatial point of view, agent i *extrudes* d from its local space. From an epistemic view this can be seen as an agent i that believes c and *utters* d to the outside world. If d is inconsistent with c, i.e., $c \sqcup d = \mathit{false}$, we can see the utterance as an intentional *lie* by agent i: The agent i utters an assertion inconsistent with its own beliefs.

6.3 Distributed Spaces

Given a group of agents G, the *distributed information* of a subgroup $I \subseteq G$ is the collective/aggregate information from all its members. It can be seen as the information that would result if the agents in I were to exchange their local information. In this section we shall define the notion of *distributed space* as a space function $\Delta[\cdot]_I$ holding the information that is distributed among the local spaces of the members of I.

The intended behavior we wish to capture can be intuitively described as follows: *The group I has the information e distributed in c, written $c \sqsupseteq \Delta[e]_I$, exactly when e follows from the combined information that the* *members of I have in c* . The following example, which we will use throughout the chapter, illustrates this.

Example 6.3.1. *Consider a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$ where $G = \{1, 2, 3\}$ and $(Con, \sqsubseteq)$ is a constraint frame. Let $c = [a]_1 \sqcup [a \to b]_2 \sqcup [b \to e]_3$. Let us assume that c is consistent and that for all $d \cong_1 a$ and for all $d' \cong_2 (a \to b)$, we have $d \sqcup d' \not\sqsupseteq e$. This assumption implies that the combination of information indistinguishable from a by agent 1 and information indistinguishable from $a \to b$ by agent 2 does not entail e, see Remark 6.2.1.*

The spatial constraint c specifies the situation where $a, a \to b$ and $b \to e$ are in the spaces of agent 1, 2 and 3, respectively. Neither agent holds e in their space in c. Nevertheless, the information e can be derived combining the local information of the three agents in c since $a \sqcup (a \to b) \sqcup (b \to e) \sqsupseteq e$. Thus we expect to have $c \sqsupseteq \Delta[e]_I$ if $I = \{1, 2, 3\}$. Analogously, from the assumption made in this example, we should have $c \not\sqsupseteq \Delta[e]_I$ if $I = \{1, 2\}$. □

6.3.1 Properties of Distributed Spaces

We shall define an indexed family of self-maps $\Delta[\cdot]_I$ that capture the above intuition. First we need notation involving indexed families of self-maps.

Notation 6.3.1. *Let $\mathbf{C} = (Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$ be a spatial cs. Given two self-maps f and g on Con we define $f \sqsubseteq_s g$ iff $f(c) \sqsubseteq g(c)$ for every $c \in Con$. For notational convenience, we shall use $\{f_I\}_{I \subseteq G}$ to denote the family of self-maps $\{f_I\}_{I \in \mathcal{P}(G)}$ indexed by subsets (subgroups) of G.*

Similarly given two indexed families $\mathcal{F} = \{f_I\}_{I \subseteq G}$, $\mathcal{G} = \{g_I\}_{I \subseteq G}$ of self-maps on Con we write $\mathcal{F} \sqsubseteq_G \mathcal{G}$ iff $f_I \sqsubseteq_s g_I$ for every $I \subseteq G$. It is easy to see that since $\mathbf{C}$ is a complete lattice, so are $(\mathtt{s}(Con), \sqsubseteq_s)$ and $(G(Con), \sqsubseteq_G)$ where $\mathtt{s}(Con)$ and $G(Con)$ are the set of self-maps on Con and the set of families of self-maps on Con indexed by G.

We now wish to single out a few fundamental properties that will help us determine distributed spaces. First of all a distributed space $\Delta[\cdot]_I$ is itself a space function (Property D.1). Second, we should have $\Delta[e]_I = [e]_i$

if $I = \{i\}$ (Property D.2) for the distributed space of a single agent is its own space. Finally, if a subgroup I has some distributed information c then any subgroup J that includes I should also have the same distributed information, i.e., $\Delta[e]_I \sqsupseteq \Delta[e]_J$ if $I \subseteq J$ (Property D.3).

Unfortunately, Properties D1-D3 do not determine $\Delta[\cdot]_I$ uniquely. Furthermore, if we were to define $\Delta[\cdot]_I$ as the constant function $\Delta[e]_I = true$, D1-D3 would be trivially met for subgroups I such that $|I| > 1$. But then we would have $true \sqsupseteq \Delta[e]_I$ for every e thus implying that any e could be distributed in the empty information *true* amongst the agents in I. Thus this choice would not be a suitable candidate for our definition of distributed space as it does not fully capture our intuition.

We shall define the $\Delta[\cdot]_I$'s as the greatest self-maps satisfying D1-D3 and justify that such canonical definition captures the intended behavior.

Definition 6.3.1. *Let* $\mathbf{C}$ *be a scs* $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$*. A family* $\{\delta[\cdot]_I\}_{I \subseteq G}$ *of self-maps on Con is said to be a* group distribution candidate *(gdc) of* $\mathbf{C}$ *if for each* $I, J \subseteq G$ *and each* $c \in Con$*:*

D.1 $\delta[\cdot]_I$ *is a space function in* $\mathbf{C}$*,*

D.2 $\delta[c]_I = [c]_i$ *if* $I = \{i\}$*, and*

D.3 $\delta[c]_I \sqsupseteq \delta[c]_J$ *if* $I \subseteq J$*.*

Define the distributed spaces *of* $\mathbf{C}$*,* $\{\Delta[\cdot]_I\}_{I \subseteq G}$*, as the least upper bound, w.r.t* $\sqsubseteq_G$*, of all group distribution candidates of* $\mathbf{C}$*. We shall say that a group* $I \subseteq G$ *has* e distributed *in* c *iff* $c \sqsupseteq \Delta[e]_I$*.* □

We now present an example of a family of self-maps which is in fact a group distribution candidate.

Example 6.3.2. *Let* $\{\delta[\cdot]_I\}_{I \subseteq G}$ *be a family of self-maps on a scs* $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$*, where*

$$\delta[c]_I = \begin{cases} [c]_i & \text{if } I = \{i\} \\ true & \text{otherwise} \end{cases}$$

for every $c \in Con$. Then it is easy to see that $\{\delta[\cdot]_I\}_{I\subseteq G}$ satisfies the properties in Definition 6.3.1, i.e. is a gdc. □

In the following remark we simplify the notation for group distribution candidates.

Remark 6.3.1. *Notice that from Definition 6.3.1 $\Delta[\cdot]_I = \bigsqcup\{\delta[\cdot]_I \mid \delta[\cdot]_I \in \{\delta[\cdot]_H\}_{H\subseteq G}$ and $\{\delta[\cdot]_H\}_{H\subseteq G}$ is a gdc$\}$ for all $\Delta[\cdot]_I \in \{\Delta[\cdot]_J\}_{J\subseteq G}$. For the sake of readability we shall write $\bigsqcup\{\delta[\cdot]_I\}$ instead of $\bigsqcup\{\delta[\cdot]_I \mid \delta[\cdot]_I \in \{\delta[\cdot]_H\}_{H\subseteq G}$ and $\{\delta[\cdot]_H\}_{H\subseteq G}$ is a gdc$\}$.*

The following theorem states that the distributed spaces $\Delta[\cdot]_I$ are the greatest self-maps that satisfy Properties D1-D3.

Theorem 6.3.1. *Given a G-indexed scs $\mathbf{C}$, its family of distributed spaces $\{\Delta[\cdot]_I\}_{I\subseteq G}$ is the greatest group distribution candidate of $\mathbf{C}$ w.r.t $\sqsubseteq_G$.*

Proof. From Remark 6.3.1 we know that $\{\Delta[\cdot]_J\}_{J\subseteq G} = \bigsqcup\{\{\delta[\cdot]_J\}_{J\subseteq G} \mid \{\delta[\cdot]_J\}_{J\subseteq G}$ *is a gdc*$\}$. Then it suffices to prove that $\{\Delta[\cdot]_J\}_{J\subseteq G}$ is a gdc, i.e. that it satisfies the properties in Definition 6.3.1. Then, for all $\Delta[\cdot]_I \in \{\Delta[\cdot]_J\}_{J\subseteq G}$:

D.1 $\Delta[\cdot]_I$ is a space function in $\mathbf{C}$. From Definition 2.5.1 we shall prove that $\Delta[\cdot]_I$ satisfies Axioms S.1, S.2 and continuity.

- Axiom S.1 : $\Delta[\mathit{true}]_I = \mathit{true}$. Applying definition of $\Delta[\cdot]_I$ we know that $\Delta[\mathit{true}]_I = \bigsqcup\{\delta[\mathit{true}]_I\} = \bigsqcup\{\mathit{true}\} = \mathit{true}$ as wanted.
- Axiom S.2 : $\Delta[c \sqcup d]_I = \Delta[c]_I \sqcup \Delta[d]_I$. We prove that $\Delta[c \sqcup d]_I \sqsupseteq \Delta[c]_I \sqcup \Delta[d]_I$. First we shall show that $\Delta[\cdot]_I$ is monotone. Notice that monotonicity of $\Delta[\cdot]_I$ follows directly from monotonicity of $\delta[\cdot]_I$ (because $\Delta[\cdot]_I$ is defined in terms of $\delta[\cdot]_I$ and $\delta[\cdot]_I$ satisfies properties in Definition 6.3.1).

  Then, by monotonicity of $\Delta[\cdot]_I$ we know that $\Delta[c \sqcup d]_I \sqsupseteq \Delta[c]_I$ and $\Delta[c \sqcup d]_I \sqsupseteq \Delta[d]_I$. Therefore $\Delta[c \sqcup d]_I \sqsupseteq \Delta[c]_I \sqcup \Delta[d]_I$. Now we prove that $\Delta[c]_I \sqcup \Delta[d]_I \sqsupseteq \Delta[c \sqcup d]_I$. We know that $\bigsqcup\{\delta[\cdot]_I\} \sqsupseteq \delta[\cdot]_H$, $\delta[\cdot]_H \in \{\delta[\cdot]_I\}_{I\subseteq G}$ where $\{\delta[\cdot]_I\}_{I\subseteq G}$ is a gdc.

Then $\bigsqcup\{\delta[c]_I\} \sqcup \bigsqcup\{\delta[d]_I\} \sqsupseteq \delta[c]_H \sqcup \delta[d]_H$ for any $H \subseteq G$. Therefore, $\bigsqcup\{\delta[c]_I\} \sqcup \bigsqcup\{\delta[d]_I\}$ is an upper bound of $\bigsqcup\{\delta[c]_I \sqcup \delta[d]_I\}$. Given that $\delta[\cdot]_I$ preserves $\sqcup$ we have $\bigsqcup\{\delta[c]_I \sqcup \delta[d]_I\} = \bigsqcup\{\delta[c \sqcup d]_I\}$ and from definition of $\Delta[\cdot]_I$ we have $\Delta[c]_I \sqcup \Delta[d]_I \sqsupseteq \Delta[c \sqcup d]_I$.

– Continuity : $\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I \sqsupseteq \Delta[\bigsqcup_{i=1}^{\infty} c_i]_I$ for any increasing chain $c_1 \sqsupseteq c_2 \sqsupseteq \dots$. Assume $\mathcal{F} = \{\delta[\cdot]_I\}_{I \subseteq G}$. Then by Remark 6.3.1 we obtain the following.
$\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I = \bigsqcup_{i=1}^{\infty} \bigsqcup\{\delta[c_i]_I\}$. Unfolding $\bigsqcup_{i=1}^{\infty}$ we obtain

$$\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I = \bigsqcup\{\delta[c_1]_I\} \sqcup \bigsqcup\{\delta[c_2]_I\} \sqcup \dots$$

From Remark 6.3.1 we know that $\Delta[\cdot]_I \sqsupseteq \delta[\cdot]_I$ for all $\delta[\cdot]_I \in \mathcal{F}$, where $\mathcal{F}$ is a gdc. Then

$$\Delta[c_1]_I \sqcup \Delta[c_2]_I \sqcup \dots \sqsupseteq \delta[c_1]_I \sqcup \delta[c_2]_I \sqcup \dots$$

for all $\delta[\cdot]_I \in \mathcal{F}$, where $\mathcal{F}$ is a gdc. Therefore, $\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I \sqsupseteq \bigsqcup_{i=1}^{\infty} \delta[c_i]_I$ for all $\delta[\cdot]_I \in \mathcal{F}$, where $\mathcal{F}$ is a gdc.

Now, by continuity of $\delta[\cdot]_I$ we have that $\bigsqcup_{i=1}^{\infty} \delta[c_i]_I = \delta[\bigsqcup_{i=1}^{\infty} c_i]_I$. Therefore $\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I \sqsupseteq \bigsqcup_{i=1}^{\infty} \delta[c_i]_I = \delta[\bigsqcup_{i=1}^{\infty} c_i]_I$ for all $\delta[\cdot]_I \in \mathcal{F}$, where $\mathcal{F}$ is a gdc. Finally, from Remark 6.3.1 we have

$$\Delta[\bigsqcup_{i=1}^{\infty} c_i]_I = \bigsqcup\{\delta[\bigsqcup_{i=1}^{\infty} c_i]_I \mid \delta[\cdot]_I \in \{\delta[\cdot]_H\}_{H \subseteq G} \text{ and } \{\delta[\cdot]_H\}_{H \subseteq G} \text{ is a gdc}\}$$

Thus, $\bigsqcup_{i=1}^{\infty} \Delta[c_i]_I \sqsupseteq \Delta[\bigsqcup_{i=1}^{\infty} c_i]_I$.

D.2 $\Delta[c]_I = [c]_i$ if $I = \{i\}$. Trivial from the definition of $\delta[\cdot]_I$ (Def. 6.3.1).

D.3 $\Delta[c]_I \sqsupseteq \Delta[c]_J$ if $I \subseteq J$. Given that the members of the families $\Delta[c]_I$ and $\Delta[c]_J$ are ordered indexwise, then we know that $\delta[c]_I \sqsupseteq \delta[c]_J$. Therefore $\Delta[c]_I \sqsupseteq \Delta[c]_J$ is trivially true from the definition of $\delta[\cdot]_I$ (Def. 6.3.1).

□

It is not difficult to see from Theorem 6.3.1 that $\Delta[\mathit{true}]_\emptyset = \mathit{true}$ and $\Delta[e]_\emptyset = \mathit{false}$ for every $e \neq \mathit{true}$. This realizes the intuition that the empty subgroup $\emptyset$ *does not* have any information whatsoever distributed in a consistent c (for if $c \sqsupseteq \Delta[e]_\emptyset$ and $c \neq \mathit{false}$ then $e = \mathit{true}$).

Another immediate consequence of Theorem 6.3.1 is that to prove that a given subgroup I *does not* have e distributed in c, i.e., $c \not\sqsupseteq \Delta[e]_I$, it suffices to find a group distribution candidate $\{\delta[\cdot]_J\}_{J\subseteq G}$ such that $c \not\sqsupseteq \delta[e]_I$. Notice that we can restrict ourselves to those group distribution candidates with constant maps $\delta[d]_J = \mathit{true}$ for every J, with $|J| > 1$, not included in the given I. We shall illustrate this and Theorem 6.3.1 in the following example.

Example 6.3.3. *Let $c = [a]_1 \sqcup [a \to b]_2 \sqcup [b \to e]_3$ as in Example 6.3.1. We want to prove $c \sqsupseteq \Delta[e]_I$ for $I = \{1,2,3\}$. From D.2 we have $c = \Delta[a]_{\{1\}} \sqcup \Delta[a \to b]_{\{2\}} \sqcup \Delta[b \to e]_{\{3\}}$. We can then use D.3 to obtain $c \sqsupseteq \Delta[a]_I \sqcup \Delta[a \to b]_I \sqcup \Delta[b \to e]_I$. Finally with the help of D.1 and Proposition 4.2.1 we infer $c \sqsupseteq \Delta[a \sqcup (a \to b) \sqcup (b \to e)]_I \sqsupseteq \Delta[e]_I$, thus $c \sqsupseteq \Delta[e]_I$ as wanted.*

We now want to show $c \not\sqsupseteq \Delta[e]_I$ for $I = \{1,2\}$. It suffices to find a group distribution candidate $\{\delta[\cdot]_J\}_{J\subseteq G}$ such that $c \not\sqsupseteq \delta[e]_I$. For groups $\{2,3\}$ and $\{1,2,3\}$ we can take the constant self-maps $\delta[d]_{\{2,3\}} \stackrel{\text{def}}{=} \mathit{true}$ and $\delta[d]_{\{1,2,3\}} \stackrel{\text{def}}{=} \mathit{true}$. We then take $\delta[d]_{\{i\}} \stackrel{\text{def}}{=} [d]_i$ for each $i \in I$, and let $\delta[\mathit{true}]_\emptyset = \mathit{true}$ and $\delta[d]_\emptyset = \mathit{false}$ for every $d \in \mathit{Con} \setminus \{\mathit{true}\}$. Finally, we take $\delta[d]_{\{1,2\}} \stackrel{\text{def}}{=} \bigsqcap\{[d']_1 \sqcup [d'']_2 \mid d' \sqcup d'' \sqsupseteq d\}$. One can verify that $\{\delta[\cdot]_J\}_{J\subseteq G}$ is a group distribution candidate. From the assumption in Example 6.3.1 one can show that $c \not\sqsupseteq \delta[e]_I$. Hence, from Definition 6.3.1, $c \not\sqsupseteq \Delta[e]_I$. □

The following example considers a scenario in which we have a group of agents $G = \{1, 2, 3\}$ and the combined local information of agents 1 and 2 entails a given constraint b.

Example 6.3.4. *Let $c = [a]_1 \sqcup [a \rightarrow b]_2 \sqcup [b \rightarrow e]_3$ as in Example 6.3.1. We want to prove $c \sqsupseteq \Delta[b]_I$ for $I = \{1, 2, 3\}$. Given that agent 1 has information a and agent 2 has $a \rightarrow b$ then, from property D.2 in Definition 6.3.1, we have $c \sqsupseteq \Delta[a]_{\{1\}} \sqcup \Delta[a \rightarrow b]_{\{2\}}$. Now we can make use of property D.3 in Definition 6.3.1 to obtain $c \sqsupseteq \Delta[a]_I \sqcup \Delta[a \rightarrow b]_I$. Finally with the help of property D.1 in Definition 6.3.1 and Axiom S.2 we infer $c \sqsupseteq \Delta[a \sqcup a \rightarrow b]_I \sqsupseteq \Delta[b]_I$, thus $c \sqsupseteq \Delta[b]_I$ as wanted.* □

We have not yet justified that our definition captures the intended meaning of distributed information. We shall do this in the next section by giving a characterization of spatially distributed information in terms of agent's *views*. Such a characterization will also give us a more explicit definition of distributed spaces.

6.3.2 Views and Distributed Spaces

We now introduce the notion of agent and group view. The *view* (or *projection*) of a spatial constraint c by an agent represents all the information the agent may see or have in c. The group view is the combined views of the agents of the group.

Definition 6.3.2 (Views). *Let $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$ be a scs. Given $i \in G$, the agent i's view of $c \in Con$ is defined as $c^i = \bigsqcup\{e \mid c \sqsupseteq [e]_i\}$. Similarly, given $I \subseteq G$, the subgroup I's view of c is defined as $c^I = \bigsqcup\{c^i \mid i \in I\}$ if $c \neq false$, otherwise $c^I = false$.* □

The following corollary states that the bigger the group of agents the bigger the group view.

Corollary 6.3.1. *Let $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$ be a scs. Then, for all $c \in Con$*

$$c^G \sqsupseteq c^{G'} \text{ if } G' \subseteq G.$$

The view of *false* by any group being equal to *false* says that anything can be observed in an inconsistent piece of information.

Example 6.3.5. *Let* $c = [a]_1 \sqcup [a \to b]_2 \sqcup [b \to e]_3$ *as in Example 6.3.1. Then* $c^{\emptyset} = true$, $c^{\{1\}} \sqsupseteq a$, $c^{\{1,2\}} \sqsupseteq a \sqcup (a \to b) = a \sqcup b$ *and* $c^{\{1,2,3\}} \sqsupseteq a \sqcup (a \to b) \sqcup (b \to e) = a \sqcup b \sqcup e$. □

We shall use the following result in the proof of the correctness of our representation of distributed information.

Lemma 6.3.1. *Let* $\{\delta^1[\cdot]_I\}_{I \subseteq G}$ *be the family of self-maps on Con where* $\delta^1[c]_J = Min\ \{e \mid e^J \sqsupseteq c\}$ *and* $c \in Con$*. Assume that the minimum element of* $\{e \mid e^J \sqsupseteq c\}$ *exists. Then* $\{\delta^1[\cdot]_I\}_{I \subseteq G}$ *is a gdc.*

Proof. In order to prove that $\{\delta^1[\cdot]_I\}_{I \subseteq G}$ is a gdc it suffices to prove that for all $\delta^1[\cdot]_J \in \{\delta^1[\cdot]_I\}_{I \subseteq G}$, $\delta^1[\cdot]_J$ satisfies the properties in Definition 6.3.1.

D.1 $\delta^1[\cdot]_J$ is a space function in **C**. In order to prove that $\delta^1[\cdot]_J$ is a space function we shall prove that it satisfies Axioms S.1 and S.2. Additionally, we shall prove that $\delta^1[\cdot]_J$ also satisfies continuity, i.e. $\bigsqcup_{i=1}^{\infty} \delta^1[c_i]_J \sqsupseteq \delta^1[\bigsqcup_{i=1}^{\infty} c_i]_J$ where $c_1 \sqcup c_2 \sqcup \ldots$ is an increasing chain. By definition of $\delta^1[\cdot]_J$ and Proposition 6.1.1 we know that $\delta^1[\cdot]_J$ and $(\cdot)^J$ form a Galois connection. In addition, by Proposition 6.1.2 we know that $\delta^1[\cdot]_J$ satisfies continuity and preserves lub's. Therefore it suffices to prove that it satisfies Axiom S.1.

- $\delta^1[true]_J = true$. Applying the definition of $\delta^1[\cdot]_J$ we obtain $\delta^1[true]_J = Min\ \{e \mid e^J \sqsupseteq true\}$. By Definition 6.3.2 we know that $true^J = true$, then $true \in \{e \mid e^J \sqsupseteq true\}$. Therefore $\delta^1[true]_J = true$ as wanted.

D.2 $\delta^1[c]_J = [c]_i$ if $I = \{i\}$. Let $S = \{e \mid e^i \sqsupseteq c\}$ where $\delta^1[c]_J = Min\ S$. Take $d = [c]_i$. By the definition of d^i (Def. 6.3.2) and given that $d^i \sqsupseteq c$ we know that $d \in S$. Furthermore we know that d is the minimum element in S, therefore $d = [c]_i = \delta^1[c]_{\{i\}}$.

D.3 $\delta^1[c]_J \sqsupseteq \delta^1[c]_K$ if $J \subseteq K$. Let $S_1 = \{e \mid e^J \sqsupseteq c\}$ and $S_2 = \{e \mid e^K \sqsupseteq c\}$ where $\delta^1[c]_J = Min\ S_1, \delta^1[c]_K = Min\ S_2$ respectively. We know that for all e' if $e' \in S_1$ then $e' \in S_2$. Then clearly $Min\ S_1 \sqsupseteq Min\ S_2$ and $\delta^1[c]_J \sqsupseteq \delta^1[c]_K$.

□

From Proposition 2.6.1 we can obtain the following corollary stating that the greatest gdc preserves arbitrary lubs. We shall use this result in the proof of our next theorem.

Corollary 6.3.2. *Let $\{\Delta[\cdot]_I\}_{I \subseteq G}$ be the distributed spaces of a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$. Then $\Delta[\cdot]_I$ preserves arbitrary lubs.*

The following central theorem justifies our definition of distributed spaces: It says that c entails the distributed information e of a given group I exactly when the combined views of c by the members of the group entail e. This expresses the correctness of our definition in the sense that it captures the intended meaning of distributed information stated at the beginning of this section.

Theorem 6.3.2. *Let $\{\Delta[\cdot]_I\}_{I \subseteq G}$ be the distributed spaces of a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$. Then $c \sqsupseteq \Delta[e]_I$ if and only if $c^I \sqsupseteq e$ for every $I \subseteq G$ and $c, e \in Con$.*

Proof. First, let $\{\delta^1[\cdot]_I\}_{I \subseteq G}$ be the family of self-maps, where $\delta^1[c]_I = Min\ \{e \mid e^I \sqsupseteq c\}$ and $c \in Con$. Then, from Proposition 6.1.2 it suffices to prove that for all $c \in Con, I \subseteq G$ then $\Delta[c]_I = \delta^1[c]_I$. We prove the two directions separately.

- $(\Rightarrow)$ $\Delta[c]_I \sqsupseteq \delta^1[c]_I$. Applying definition of $\Delta[\cdot]_I$ (Def. 6.3.1) we have $\Delta[c]_I = \bigsqcup\{\delta[\cdot]_I \mid \delta[\cdot]_I \in \{\delta[\cdot]_H\}_{H \subseteq G}$ *and* $\{\delta[\cdot]_H\}_{H \subseteq G}$ *is a gdc*$\}$. Let $S = \{\delta[c]_I \mid \delta[\cdot]_I \in \{\delta[\cdot]_H\}_{H \subseteq G}$ *and* $\{\delta[\cdot]_H\}_{H \subseteq G}$ *is a gdc*$\}$. From Lemma 6.3.1 we know that $\{\delta^1[\cdot]_I\}_{I \subseteq G}$ is a gdc. Therefore, we know that $\{\delta^1[\cdot]_I\}_{I \subseteq G} \in S$ and given that $\Delta[c]_I = \bigsqcup S$ then $\Delta[c]_I \sqsupseteq \delta^1[c]_I$.

- ($\Leftarrow$) $\delta^1[c]_I \sqsupseteq \Delta[c]_I$. Let $e = Min\ \{d \mid d^I \sqsupseteq c\} = \delta^1[c]_I$. By definition of $(\cdot)^I$ (Def. 6.3.2) we have that

$$e^I = \bigsqcup\{a_i \mid e \sqsupseteq [a_i]_i \ and\ i \in I\} \sqsupseteq c. \tag{6.3.1}$$

Let $S = \{a_i \mid e \sqsupseteq [a_i]_i \ and\ i \in I\}$ and $S' = \{[a_i]_i \mid a_i \in S\}$. From property D.2 in Definition 6.3.1 we have that $[a_i]_i = \Delta[a_i]_{\{i\}}$ for all $i \in I$. From Property D.3 in Definition 6.3.1 we have that

$$\bigsqcup_{i\in I}[a_i]_i = \bigsqcup_{i\in I}\Delta[a_i]_{\{i\}} \sqsupseteq \bigsqcup_{i\in I}\Delta[a_i]_I. \tag{6.3.2}$$

From Property D.1 in Definition 6.3.1 and Corollary 6.3.2 we have that

$$\bigsqcup_{i\in I}\Delta[a_i]_I \sqsupseteq \Delta[\bigsqcup_{i\in I} a_i]_I. \tag{6.3.3}$$

Now, from Equation 6.3.1 we know $\bigsqcup S \sqsupseteq c$. Then from monotonicity of $\Delta[\cdot]_i$ we have that

$$\Delta[\bigsqcup S]_I \sqsupseteq \Delta[c]_I. \tag{6.3.4}$$

From Equation 6.3.2 and Equation 6.3.3 we know that

$$\bigsqcup_{i\in I}[a_i]_i \sqsupseteq \bigsqcup_{i\in I}\Delta[a_i]_I \sqsupseteq \Delta[\bigsqcup_{i\in I} a_i]_I. \tag{6.3.5}$$

We also know that

$$\Delta[\bigsqcup S]_I = \Delta[\bigsqcup_{i\in I} a_i]_I. \tag{6.3.6}$$

Therefore by Equation 6.3.6, Equation 6.3.5 and Equation 6.3.4 we have

$$\bigsqcup_{i\in I}[a_i]_i \sqsupseteq \bigsqcup_{i\in I}\Delta[a_i]_I \sqsupseteq \Delta[c]_I.$$

Recall that $\bigsqcup_{i\in I}[a_i]_i = \bigsqcup S'$ and that $\bigsqcup S = e^I$, therefore $e = \delta^1[c]_I = \bigsqcup_{i\in I}[a_i]_i$ and thus $\delta^1[c]_I \sqsupseteq \Delta[c]_I$.

□

The following example illustrates Theorem 6.3.2 with our recurring example.

Example 6.3.6. *Let* $c = [a]_1 \sqcup [a \to b]_2 \sqcup [b \to e]_3$ *as in Example 6.3.1. For* $I = \{1, 2, 3\}$ *from Example 6.3.5 we have* $c^I \sqsupseteq e$*. It follows from Theorem 6.3.2 that* $c \sqsupseteq \Delta[e]_I$ *as wanted. For* $I = \{1, 2\}$*, from the assumptions in Example 6.3.1 we have* $c^I \not\sqsupseteq e$*. Thus* $c \not\sqsupseteq \Delta[e]_I$ *by Theorem 6.3.2.* □

Theorem 6.3.2 states that the self-maps $\Delta[\cdot]_I$ and $(\cdot)^I$ form a *Galois connection* : With $\Delta[\cdot]_I$ and $(\cdot)^I$ being the lower and the upper adjoints. From Proposition 6.1.1 we know that an essential fact of a Galois connection is that upper u and l lower adjoints uniquely determine each other: In particular $l(x)$ is the least element y such that $u(y)$ is above x. This fact gives the following more explicit definition of $\Delta[\cdot]_I$.

Corollary 6.3.3. *Let* $\{\Delta[\cdot]_I\}_{I \subseteq G}$ *be the distributed spaces of a scs* $(Con, \sqsubseteq, \{[\cdot]_i\}_{i \in G})$*. For every* $e \in Con, I \subseteq G$*, we have* $\Delta[e]_I = Min\ \{c \mid c^I \sqsupseteq e\}$*.*

In what follows we discuss some noteworthy implications of the correspondence between views and distributed information.

Views as Extrusion

An interesting fact arising from the correspondence in Theorem 6.3.2 is the relation between agent views and the notion of extrusion discussed in Section 6.2.1. Given a space $[\cdot]_i$, it is natural to ask how to derive an extrusion function $\uparrow_i$ for it.

As presented in Part II, from set theory we know that $[\cdot]_i$ admits extrusion, i.e. it has a right inverse, iff $[\cdot]_i$ is surjective. Also in Part II we gave explicit constructions of extrusion functions for scs's. It turns out that the notion of view gives us a more constructive extrusion for a given space: Views are extrusion functions. More precisely, if $[\cdot]_i$ is surjective then the agent view operator $(\cdot)^i$ is an extrusion function for the space $[\cdot]_i$.

Corollary 6.3.4. *Let $\mathcal{C}$ be a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$. If $[\cdot]_i$ is surjective then $[e^i]_i = e$ for every $e \in Con$.*

Corollary 6.3.4 is a consequence of the following theorem.

Theorem 6.3.3. *Let $\{\Delta[\cdot]_I\}_{I\subseteq G}$ be the distributed spaces of a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$. For every $e \in Con, I \subseteq G$, we have (1) $e \sqsupseteq \Delta[e^I]_I$, and (2) if $\Delta[\cdot]_I$ is surjective then $\Delta[e^I]_I = e$.*

Proof. The proof of (1) follows directly from Property (3) in Proposition 6.1.1. The proof of (2) follows directly from Property (3) in Proposition 6.1.2. □

In the following result we shall prove that distributed information can be also captured as : the *smallest* combination of agents' spaces such that if we join all their local information we can entail certain piece of information. We shall represent this intuition with the following family of self-maps. Suppose that we have a group of agents G and a family of self-maps $\{\delta^2[\cdot]_I\}_{I\subseteq G}$ where $\delta^2[c]_I = \bigsqcap\{\bigsqcup_{i\in I}[e_i]_i \mid \bigsqcup_{i\in I} e_i \sqsupseteq c\}$ for all $I \subseteq G$ and a given constraint c. We shall prove that $\{\delta^2[\cdot]_I\}_{I\subseteq G}$ is the greatest distribution candidate by proving that $\{\delta^2[\cdot]_I\}_{I\subseteq G}$ is equal to the distributed spaces $\{\Delta[\cdot]_I\}_{I\subseteq G}$.

Theorem 6.3.4. *Let $\{\Delta[\cdot]_I\}_{I\subseteq G}$ be the distributed spaces of a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$ and $\{\delta^2[\cdot]_I\}_{I\subseteq G}$ be a family of self-maps on Con where $\delta^2[c]_I = \bigsqcap\{\bigsqcup_{i\in I}[e_i]_i \mid \bigsqcup_{i\in I} e_i \sqsupseteq c\}$, for all $I \subseteq G$, $c \in Con$. Then $\Delta[\cdot]_I = \delta^2[\cdot]_I$.*

Proof. From Corollary 6.3.3 and Lemma 6.3.1 it suffices to prove that $\{\delta^2[\cdot]_I\}_{I\subseteq G}$ is equal to the family $\{\delta^1[\cdot]_I\}_{I\subseteq G}$ where $\delta^1[\cdot]_I = Min\ \{e \mid e^I \sqsupseteq c\}$. Let $S_1 = \{e \mid e^I \sqsupseteq c\}$, $S_2 = \{\bigsqcup_{i\in I}[e_i]_i \mid \bigsqcup_{i\in I} e_i \sqsupseteq c\}$ and $e' = \delta^1[c]_I$. Notice that $\delta^2[c]_I = \bigsqcap S_2$ and $\delta^1[c]_I = Min\ (S_1)$. To prove that $\delta^1[\cdot]_I = \delta^2[\cdot]_I$ we shall prove that $e' \in S_2$ and thus $e' = \bigsqcap S_2$. Applying $(\cdot)^I$ (Def. 6.3.2) we obtain $(e')^I = \bigsqcup\{d_i \mid e' \sqsupseteq [d_i]_i \ and\ i \in I\}$. Then, we know that $e' \sqsupseteq \bigsqcup_{i\in I}[d_i]_i$. Also, from definition of $\delta^1[\cdot]_I$ we know that $(e')^I \sqsupseteq c$, therefore $\bigsqcup_{i\in I} d_i \sqsupseteq c$. But we also know that e' is the minimum element of S_1 then it cannot be bigger than $\bigsqcup_{i\in I}[d_i]_i$ and thus $e' = \bigsqcup_{i\in I}[d_i]_i$. Thus $e' \in S_2$ and given that $e' = Min\ S_1$ then $e' = \bigsqcap S_2 = \delta^2[c]_I$ and $\delta^1[c]_I = \delta^2[c]_I$. □

Compactness of Distributed Information

Suppose that a group with infinitely many agents I have e distributed in c, it is natural to ask whether a finite subset of I may also have e distributed in c. Consider the following modified version of Example 6.3.1.

Example 6.3.7. *Consider a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$ where $G = \mathbb{N}$ and $(Con, \sqsubseteq)$ is a constraint frame. Consider $a_0, a_1, \ldots \in Con$ such that all joined together produced e i.e., $\bigsqcup_{i\geq 0} a_i = e$. Take $c = [\ a_0\ ]_0 \sqcup \bigsqcup_{i>0} [\ a_{i-1} \rightarrow a_i\ ]_i$. From Theorem 6.3.1 it follows that G has e distributed in c, i.e., $c \sqsupseteq \Delta[e]_G$. One may wonder if there exists a finite subset $I \subseteq G$ such that $c \sqsupseteq \Delta[e]_I$.* □

The following theorem provides an answer to the question above: if e is a compact element in the underlying cs then the finite subgroup that has e distributed in c is guaranteed to exists.

Theorem 6.3.5. *Let $\{\Delta[\cdot]_I\}_{I\subseteq G}$ be the distributed spaces of a scs $(Con, \sqsubseteq, \{[\cdot]_i\}_{i\in G})$. Whenever e is a compact element, if $c \sqsupseteq \Delta[e]_I$ then there exists a* finite *set $J \subseteq I$ such that $c \sqsupseteq \Delta[e]_J$.*

Proof. The proof proceed by cases. If I is finite then $J = I$. If I is infinite then assume that $c \sqsupseteq \Delta[e]_I$. Then from Theorem 6.3.2 we know that.

$$e \sqsubseteq c^I. \tag{6.3.7}$$

By Definition 6.3.2 we have $c^I = \bigsqcup\{c^i \mid i \in I\} = \bigsqcup\{c^{\{i\}} \mid \{i\} \subseteq I\}$. Let $I_I = \{c^{\{i\}} \mid \{i\} \subseteq I\}$. Now, take $S_I = \{c^H \mid H \subseteq_{fin} I\}$. Notice that $I_I \subseteq S_I$. Therefore

$$c^I = \bigsqcup I_I \sqsubseteq \bigsqcup S_I. \tag{6.3.8}$$

We now prove that S_I is a directed set. Take $c^{H_1}, c^{H_2} \in S_I$. Given that H_1 and H_2 are finite sets then their union $H_1 \cup H_2$ will be finite as well, therefore $c^{H_1\cup H_2} \in S_I$. From Corollary 6.3.1 we have that $c^{H_1\cup H_2} \sqsupseteq c^{H_1}$ and $c^{H_1\cup H_2} \sqsupseteq c^{H_2}$. Thus S_I is directed. Now, from Equation 6.3.7 and Equation 6.3.8 we know that $e \sqsubseteq c^I \sqsubseteq \bigsqcup S_I$. Therefore, from the assumption that e is compact and given that S_I is directed, then from the definition of compactness in Notation 2.4.1 (i.e. if $c \in Con$ is compact then for any directed subset D of Con, $c \sqsubseteq \bigsqcup D$ implies $c \sqsubseteq d$ for some $d \in D$.), we know that there exists a $J \subseteq_{fin} I$ s.t. $c^J \in S_I$ and $e \sqsubseteq c^J$ as wanted. □

6.4 Summary

In this chapter we have extended the definition of scs in order to account for an infinite number of agents. We gave an algebraic characterization of the notion of distributed information We started by finding families of functions over a scs satisfying the distributed information axioms. We have called those families *group distribution candidates* (gdc's). We proved that distributed information can be represented in scs's as the *greatest* gdc. We also observed that the notion of distributed information forms a *Galois connection* with the notion of *group view*. We intuitively defined group view as the combination of the individual information that every agent i belonging to a group of agents G sees or has within its own space. By means of Galois connection properties we gave a representation of distributed information in terms of group view.

Finally, we proved that if an infinite set of agents G has distributed information of a compact constraint c then there exists a finite subset of agents H such that H also has distributed information of c.

Seven

Conclusions and Related Work

The last part of this manuscript is devoted to provide concluding remarks on the expressiveness of spatial constraint systems. We shall also discuss on related and future work.

7.1 Concluding Remarks

In this thesis we have studied the expressiveness of spatial constraint systems (scs's). We have used the definition of scs's as complete lattices endow with self-maps (space functions) on the elements of the lattice representing the concept of space. We have also used spatial constraint systems with extrusion (scse's), which extend scs's with right inverses for space functions to account for the mobile behaviour of agents.

We have studied the extrusion problem for a meaningful family of scs's called *Kripke spatial constraint systems* (Kripke scs's). We started by providing an algebraic representation of the notion of normality for modal logics. We observed that a self-map on the elements of scs's can be called normal if and only if preserves finite suprema. We then studied the minimal conditions guaranteeing the construction of right inverses for space functions in Kripke scs's. We provided the main results of this part of the thesis : (1) We proved that the condition of being *determinant complete* is the minimal condition on Kripke structures assuring the construction of right inverses. (2) We identified a complete taxonomy of right inverses over Kripke scs.

We then tackled the problem of representing the epistemic notion of *knowledge* by means of scs's. As a contribution in this part of the thesis we have derived an algebraic characterization of the epistemic notion of $S4$ knowledge. We started by studying the representation of knowledge as spatial constraint system (known as *knowledge constraint systems*) given in [KPPV12]. We have identified that we could satisfy the $S4$ knowledge axioms by means of a *Kuratowski closure operator*. Then we proved that the *global operator*, an operator derived from scs's to make a certain piece of information available in any nesting of spaces is a Kuratowski closure operator. Next we proved that for any spatial constraint system whose space function are *continuous* then we can derive a scs satisfying $S4$ knowledge in which the space functions are global operators. As a final result for this part of the dissertation we proved the *completeness* of our characterization w.r.t. $S4$ knowledge validity.

Finally we have characterized the notion of *distributed information* in terms of scs's. In order to account for an infinite number of agents we have extended the definition of scs's. We have called this extension *indexed* scs's. Then, we gave an algebraic treatment to the notion of distributed information. The algebraic representation of distributed information is given by means of families of functions over the underlying indexed scs satisfying the distributed information properties. We have called such families *group distributed candidates* (gdc). We have then represented distributed information as the *greatest* gdc. We have identified that distributed information forms a *Galois connection* with the notion of *group view*. Then, by Galois connection properties, we have provided a representation of distributed information in terms of views. Lastly, we have obtained a *compactness* result. We proved that if a given infinite set of agents G has distributed information about certain piece of (compact) information c, then there exists a finite subset of agents $H \subset G$ such that H also has distributed information of c.

7.2 Related and Future Work

The Extrusion Problem.

In previous work [HPRV15] the authors have derived an inverse modality but only for the specific case of a logic of belief. The work was neither concerned with giving a *complete* characterization of the existence of right inverses nor deriving *normal* inverses. The constraint systems in this work can be seen as modal extension of geometric logic [Vic96]. Modal logic have also been studied from an algebraic perspective by using modal extensions of boolean and Heyting algebras in [Mac81, BDRV02, CZ97]. These works, however, do not address issues related to inverse modalities. Inverse modalities have been used in temporal, epistemic and logic for concurrency. In [RS97] the authors discusses inverse temporal and epistemic modalities from a proof theoretic perspective. The work in [PU11, DNF90, GKP92] uses modal logic with reverse modalities for specifying true concurrency and [DNMV90, DNV95] use backward modalities for characterizing branching bisimulation. None of these works are concerned with an algebraic approach or with deriving inverse modalities for modal languages.

As future work, it is natural to consider spatial extensions of the constraint systems from [GSPV15] towards a more general framework of spatial soft ccp.

Knowledge in Terms of Space.

In epistemology the question whether knowledge is definable in terms of belief has always played an important role among researches. Some authors have studied this question in the general framework of epistemic and doxastic logics. In [HSS09] the authors give some theorems and proofs about when the epistemic logic S5 can or cannot be defined (explicitly or implicitly) in terms of belief (KD45). The notion of knowledge we have considered here is that corresponding to the epistemic logic $S4$. In [KPPV12] the authors introduce the notion of *knowledge constraint systems* (s4cs) in order to capture the behaviour of the $S4$ epistemic logic by means of scs's.

However the way in which they characterize this notion is by *imposing* the $S4$ knowledge axioms over the constraint systems (restricting this way the expressiveness of the scs). We instead satisfied the $S4$ knowledge axioms through a *derived* operator from the underlying scs.

As part of the future work we plan to study whether is possible to characterize the notion of $S5$ knowledge by means of derived constructors over scs's.

Distributed Information in Terms of Space.

The authors in [FHMV95] have noticed the importance of the analysis of multi-agent systems and have equipped a modal language with modalities for the analysis of groups of agents in those systems. The modalities in this language represent not only the epistemic notion of *knowledge* but also the notions of *common* and *distributed knowledge*.

In [HS04] the authors study the epistemic notions of *knowledge* and *common knowledge* in the context of an infinite amount of agents. They argue that although the *reasoning* about these epistemic notions are common practice in computer science, this reasoning has always assumed a *finite* number of agents. Therefore, all the results obtained in the analysis of these notions require having a finite amount of agents. Nevertheless, we can find that in many applications the number of agents is not known in advance and can be unbounded. Thus it is important to provide tools and techniques allowing the reasoning about these epistemic notions allowing an *infinite* amount of agents.

The authors in [HS04] do not explicitly study the notion of distributed knowledge. However they provide a discussion in which they say that for a finite set of agents adding an operator characterizing this notion should not impose new difficulties. Furthermore, they discuss that for an infinite set of agents adding distributed knowledge might lead to new complications. For example one might need to make assumptions to distinguish between singletons and sets with larger cardinalities, giving that common knowledge and distributed knowledge coincide just if the group of agents is a singleton.

Following that intuition, we have decided to tackle the problem of having distributed information among an infinite set of agents from an algebraic perspective.

As a future work we shall equip scse's with operational and denotational semantics allowing us to use our proposed characterization of distributed information for the analysis of programs, i.e. to use distributed information for verification purposes.

Bibliography

[ABP$^+$11] Andrés Aristizábal, Filippo Bonchi, Catuscia Palamidessi, Luis Pino, and D. Valencia, Frank. Deriving labels and bisimilarity for concurrent constraint programming. In *Proceedings of the 14th International Conference on Foundations of Software Science and Computation Structures, FOSSACS 2011*, LNCS, pages 138–152. Springer, 2011.

[AGM95] Samson Abramsky, Dov M Gabbay, and Thomas SE Maibaum. *Handbook of logic in computer science (vol. 3): semantic structures.* Oxford University Press, 1995.

[BDPP95] Frank S. Boer, Alessandra Di Pierro, and Catuscia Palamidessi. Nondeterminism and infinite computations in constraint programming. *Theoretical Computer Science*, pages 37–78, 1995.

[BDRV01] Patrick Blackburn, Maarten De Rijke, and Yde Venema. Modal logic, volume 53 of cambridge tracts in theoretical computer science, 2001.

[BDRV02] Patrick Blackburn, Maarten De Rijke, and Yde Venema. *Modal Logic.* Cambridge University Press, 1st edition, 2002.

[BJ72] TS Blyth and MF Janowitz. *Residuation Theory, International Series of Monographs in Pure and Applied Mathematics*, volume 102. Pergamon Press, Oxford, 1972.

[BvBW06] Patrick Blackburn, Johan FAK van Benthem, and Frank Wolter. *Handbook of modal logic*, volume 3. Elsevier, 2006.

[Car04] Luca Cardelli. Brane calculi. In *International Conference on Computational Methods in Systems Biology*, pages 257–278. Springer, 2004.

[CC02] Luís Caires and Luca Cardelli. A spatial logic for concurrency (part ii). In *Proceedings of the 13th International Conference of Concurrency Theory, CONCUR 2002*, pages 209–225, 2002.

[CC03] Luís Caires and Luca Cardelli. A spatial logic for concurrency (part i). *Information and Computation*, pages 194–235, 2003.

[CG98] Luca Cardelli and Andrew D. Gordon. Mobile ambients. In *Proceedings of the First International Conference on Foundations of Software Science and Computation Structure, FoSSaCS'98*, pages 140–155, 1998.

[CGP99] Edmund M Clarke, Orna Grumberg, and Doron Peled. *Model checking*. MIT press, 1999.

[CZ97] Alexander Chagrov and Michael Zakharyaschev. Modal logic, volume 35 of oxford logic guides, 1997.

[DEW13] Klaus Denecke, Marcel Erné, and Shelly L Wismath. *Galois connections and applications*, volume 565. Springer Science & Business Media, 2013.

[DGG07] André DeHon, Jean-Louis Giavitto, and Fréderic Gruau. 06361 executive report–computing media languages for space-oriented computation. In *Dagstuhl Seminar Proceedings*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2007.

[DNF90] Rocco De Nicola and Gian Luigi Ferrari. Observational logics and concurrency models. In *FSTTCS'90*, volume 472 of *LNCS*, pages 301–315. Springer, 1990.

[DNMV90] Rocco De Nicola, Ugo Montanari, and Frits Vaandrager. Back and forth bisimulations. In *CONCUR '90*, volume 458 of *LNCS*, pages 152–165. Springer, 1990.

[DNV95] Rocco De Nicola and Frits Vaandrager. Three logics for branching bisimulation. *Journal of the ACM (JACM)*, 42(2):458–487, 1995.

[DP02] Brian A Davey and Hilary A Priestley. *Introduction to lattices and order*. Cambridge university press, 2nd edition, 2002.

[FHMV95] Ronald Fagin, Joseph Y Halpern, Yoram Moses, and Moshe Y Vardi. *Reasoning about knowledge*. MIT press Cambridge, 4th edition, 1995.

[Get63] Edmund L Gettier. Is justified true belief knowledge? *analysis*, 23(6):121–123, 1963.

[GHK+03] Gerhard Gierz, Karl Heinrich Hofmann, Klaus Keimel, Jimmie D. Lawson, Michael Mislove, and Dana S. Scott. *Continuous lattices and domains*. Cambridge University Press, 1st edition, 2003.

[GHP+17] Michell Guzman, Stefan Haar, Salim Perchy, Camilo Rueda, and Frank D Valencia. Belief, knowledge, lies and other utterances in an algebra for space and extrusion. *Journal of Logical and Algebraic Methods in Programming*, 86(1):107–133, 2017.

[GKP92] Ursula Goltz, Ruurd Kuiper, and Wojciech Penczek. Propositional temporal logics and equivalences. In *CONCUR'92*, volume 630 of *LNCS*, pages 222–236. Springer, 1992.

[Gol03] Robert Goldblatt. Mathematical modal logic: a view of its evolution. *Journal of Applied Logic*, 1(5):309–392, 2003.

[GSPV15] Fabio Gadducci, Francesco Santini, Luis F Pino, and Frank D Valencia. A labelled semantics for soft concurrent constraint programming. In *International Conference on Coordination Languages and Models*, pages 133–149. Springer, 2015.

[Hin62] Jaakko Hintikka. *Knowledge and belief*. Cornell Univeristy Press, 1962.

[HM85] Matthew Hennessy and Robin Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM (JACM)*, 32(1):137–161, 1985.

[HPRV15] Stefan Haar, Salim Perchy, Camilo Rueda, and Frank D. Valencia. An algebraic view of space/belief and extrusion/utterance for concurrency/epistemic logic. In *PPDP 2015*, pages 161–172. ACM, 2015.

[HS04] Joseph Y Halpern and Richard A Shore. Reasoning about common knowledge with infinitely many agents. *Information and Computation*, 191(1):1–40, 2004.

[HSS09] Joseph Y Halpern, Dov Samet, and Ella Segev. Defining knowledge in terms of belief: The modal logic perspective. *The Review of Symbolic Logic*, pages 469–487, 2009.

[KPPV12] Sophia Knight, Catuscia Palamidessi, Prakash Panangaden, and Frank D Valencia. Spatial and epistemic modalities in constraint-based process calculi. In *CONCUR 2012*, volume 7454 of *LNCS*, pages 317–332. Springer, 2012.

[Mac81] DS Macnab. Modal operators on heyting algebras. *Algebra Universalis*, 12(1):5–29, 1981.

[McM03] Kenneth L McMillan. *Model checking*. John Wiley and Sons Ltd., 2003.

[Mil80] Robin Milner. *A calculus of communicating systems*. Springer, 1980.

[Mil89] Robin Milner. *Communication and concurrency*, volume 84. Prentice hall New York etc., 1989.

[MS92] Robin Milner and Davide Sangiorgi. Barbed bisimulation. *Automata, Languages and Programming*, pages 685–695, 1992.

[MT44] John Charles Chenoweth McKinsey and Alfred Tarski. The algebra of topology. *Annals of mathematics*, pages 141–191, 1944.

[ORV13] Carlos Olarte, Camilo Rueda, and Frank D Valencia. Models and emerging trends of concurrent constraint programming. *Constraints*, 18(4):535–578, 2013.

[Par81] David Park. Concurrency and automata on infinite sequences. *Theoretical computer science*, pages 167–183, 1981.

[PM92] Amir Pnueli and Zohar Manna. *The temporal logic of reactive and concurrent systems*. Springer, 1992.

[Pop94] Sally Popkorn. *First steps in modal logic*. Cambridge University Press, 1st edition, 1994.

[Pri67] Arthur N Prior. *Past, present and future*, volume 154. Oxford University Press, 1967.

[PSSS93] Prakash Panangaden, Vijay Saraswat, Philip J Scott, and RAG Seely. A hyperdoctrinal view of concurrent constraint programming. In *Workshop of Semantics: Foundations and Applications, REX*, pages 457–476. Springer, 1993.

[PU11] Iain Phillips and Irek Ulidowski. A logic with reverse modalities for history-preserving bisimulations. In *EXPRESS'11*, volume 64 of *EPTCS*, pages 104–118, 2011.

[RS97] Mark Ryan and Pierre-Yves Schobbens. Counterfactuals and updates as inverse modalities. *Journal of Logic, Language and Information*, pages 123–146, 1997.

[SRP91] Vijay A Saraswat, Martin Rinard, and Prakash Panangaden. Semantic foundations of concurrent constraint programming. In *POPL'91*, pages 333–352. ACM, 1991.

[Vic96] Steven Vickers. *Topology via logic.* Cambridge University Press, 1st edition, 1996.

Index

ÉCOLE DOCTORALE
Sciences et technologies de l'information et de la communication (STIC)

Titre : Sur l'expressivité des systèmes de contraintes spatiales

Mots clés : contraintes, logique modale, théorie d'ordre, logiques épistémiques, calculs de processus

Résumé : Les comportement épistémiques, mobiles et spatiaux sont omniprésent dans les systèmes distribués d'aujourd'hui. La nature épistémique intrinsèque de ces types de systèmes, provient des interactions des éléments qui en font parties. La plupart des gens sont familiarisés avec les systèmes numériques où les utilisateurs partagent leurs croyances, opinions et même des mensonges intentionnels (des canulars). Les modèles de ces systèmes doivent prendre en compte des interactions avec les autres ainsi que leurs qualité distribuée. Les comportements spatiaux et mobiles sont exposées par des applications et des données qui se déplaçant à travers des espaces (éventuellement imbriqués) définis par, par example, des cercles d'amis, des groupes et des dossiers communs. Nous croyons donc qu'une solide compréhension de la notion d'espace et mobilité spatial ainsi que le flux d'information épistémique est pertinente dans de nombreux modèles de systèmes distribués d'aujourd'hui. Afin d'analyser la connaissance, l'espace et la mobilité dans les systèmes distribués, nous élargissons sur la théorie mathématique- ment simple et élégante des systèmes de contraintes (sc), utilisée pour représenter l'information et le changement d'information dans les systèmes concurrents. Dans le modèle déclaratif formel connu sous le nom de programmation concurrent par contraintes, les systèmes de contraintes fournissent les domaines de base et les opérations pour les fondements sémantiques de ce modèle. Les systèmes des contraintes spatiales (scs) sont des structures algébriques qui étendent les sc pour le raisonnement sur les comportement spatiaux et épistémiques de base tels que la croyance et l'extrusion. Les affirmations spatiales et épistémiques peuvent être considérées comme des modalités spécifiques. D'autres modalités peuvent être utilisées pour les assertions concernant le temps, les connaissances et même pour l'analyse des groupes parmi d'autres concepts utilisés dans la spécification et la vérification des systèmes concurrents. Dans cette thèse nous étudions l'expressivité des systèmes de contraintes spatiales dans la perspective plus large du comportement modal et épistémique. Nous montrerons que les systèmes de contraintes spatiales sont suffisamment robustes pour capturer des modalités inverses et pour obtenir de nouveaux résultats pour les logiques modales. Également, nous montrerons que nous pouvons utiliser les scs pour exprimer un comportement épistémique fondamental comme connaissance. Enfin, nous allons donner une caractérisation algébrique de la notion de l'information distribuée au moyen de constructeurs sur scs.

Title : On the expressiveness of spatial constraint systems

Keywords : constraints, modal logic, order theory, epistemic logics, process calculi

Abstract : Epistemic, mobile and spatial behaviour are commonplace in to- day's distributed systems. The intrinsic epistemic nature of these systems arises from the interactions of the elements of which they are comprised. Most people are familiar with digital systems where users share their beliefs, opinions and even intentional lies (hoaxes). Models of such systems must take into account the interactions with others as well as the distributed quality presented by them. Spatial and mobile behaviour are exhibited by applications and data moving across possibly nested spaces defined by, for example, friend circles, groups, and shared folders. Thus a solid understanding of the notion of space and spatial mobility as well as the flow of epistemic information is relevant in many models of today's distributed systems. In order to analyze knowledge, space, and mobility in distributed systems, we expand upon the mathematically simple and elegant theory of constraint systems (cs), used to represent information and information change in concurrent systems. In the formal declarative model known as concurrent constraint programming, constraint systems provide the basic domains and operations for the semantic foundations of this model. Spatial constraint systems (scs's) are algebraic structures that extend cs's for reasoning about basic spatial and epistemic behaviour such as belief and extrusion. Both, spatial and epistemic assertions, can be viewed as specific modalities. Other modalities can be used for assertions about time, knowledge and even the analysis of groups among other concepts used in the specification and verification of concurrent systems.
In this thesis we study the expressiveness of spatial constraint systems in the broader perspective of modal and epistemic behaviour. We shall show that spatial constraint systems are sufficiently robust to capture inverse modalities and to derive new results for modal logics. We shall show that we can use scs's to express a fundamental epistemic behaviour such as knowledge. Finally we shall give an algebraic characterization of the notion of distributed information by means of constructors over scs's.

Université Paris-Saclay
Espace Technologique / Immeuble Discovery
Route de l'Orme aux Merisiers RD 128 / 91190 Saint-Aubin, France