



HAL
open science

Characterizing Right Inverses for Spatial Constraint Systems with Applications to Modal Logic

Michell Guzmán, Salim Perchy, Camilo Rueda, Frank Valencia

► **To cite this version:**

Michell Guzmán, Salim Perchy, Camilo Rueda, Frank Valencia. Characterizing Right Inverses for Spatial Constraint Systems with Applications to Modal Logic. *Theoretical Computer Science*, 2018, 744 (56–77). hal-01675010v2

HAL Id: hal-01675010

<https://inria.hal.science/hal-01675010v2>

Submitted on 15 May 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Characterizing Right Inverses for Spatial Constraint Systems with Applications to Modal Logic[☆]

Michell Guzmán

INRIA-LIX, École Polytechnique de Paris

Salim Perchy

Loria, INRIA Grand-Est

Camilo Rueda

Pontificia Universidad Javeriana de Cali

Frank D. Valencia

CNRS & LIX, École Polytechnique de Paris

Pontificia Universidad Javeriana de Cali

Abstract

Spatial constraint systems are algebraic structures from concurrent constraint programming to specify spatial and epistemic behavior in multi-agent systems. In this paper spatial constraint systems are used to give an abstract characterization of the notion of normality in modal logic and to derive right inverse/reverse operators for modal languages. In particular, a necessary and sufficient condition for the existence of right inverses is identified and the abstract notion of normality is shown to correspond to the preservation of finite suprema. Furthermore, a taxonomy of normal right inverses is provided, identifying the greatest normal right inverse as well as the complete family of minimal right inverses. These results are applied to existing modal languages such

[☆]This work has been partially supported by the Colciencias project 125171250031 CLASSIC, STIC Am-Sud 17-STIC-04 project EPIC, and COFINPRO2016 project 4473 CREATE.

as the weakest normal modal logic, Hennessy-Milner logic, and linear-time temporal logic. Some implications of these results are also discussed in the context of modal concepts such as bisimilarity and inconsistency invariance.

Keywords: constraint systems, concurrent constraint programming, concurrency theory, modal logic, inverse operators.

Contents

1	Introduction	4
2	Background: Spatial Constraint Systems	9
3	Constraint Frames and Normal Self Maps	15
3.1	Normal Self-Maps	18
3.2	Summary	20
4	Extrusion Problem for Kripke Constraint Systems	20
4.1	KS and Kripke SCS	21
4.2	Complete Characterization of the Existence of Right Inverses	23
4.3	Deriving Maximum Right-Inverse	26
4.4	Summary	27
5	Deriving Normal Right-Inverses	27
5.1	Deriving Maximum Normal Right Inverse	29
5.2	Deriving Minimal Normal-Right Inverses	33
5.3	Summary	40
6	Applications	42
6.1	Right-Inverse Modalities	44
6.2	Normal Inverses Modalities	45
6.3	Inconsistency Invariance	45
6.4	Bisimilarity Invariance	45
6.5	Temporal Operators	46

7 Concluding Remarks and Related Work	48
Index	50

1. Introduction

In this paper we give an abstract characterization of the notion of normality in modal logic and derive right inverse operators for modal languages. We shall do this by using *spatial constraint systems* [1].

Spatial Constraint Systems. Constraint systems are algebraic structures for the semantics of process calculi from concurrent constraint programming (CCP) [2, 3, 1, 4, 5, 6]. They specify the domain and elementary operations and partial information upon which programs (processes) of these calculi may act. In this paper we shall study constraint systems as semantic structures for modal logic.

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

A distinctive property of CCP processes is that they can be interpreted as both concurrent computational entities and logic specifications (e.g., process composition can be seen as parallel execution and conjunction). The CCP operations and their logical counterparts typically have a corresponding elementary construct or operation on the elements of the constraint system. In particular, parallel composition and conjunction correspond to the *join* operation while local variables and existential quantification correspond to a cylindrification operation on the set of constraints [2] that project away the information of a given existential (or local) variable.

Similarly, the notion of computational space and the epistemic notion of belief in the spatial concurrent constraint programming (SCCP) process calculi [1] correspond to a family of functions $[\cdot]_i : Con \rightarrow Con$ 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 Extrusion Problem. Given a space function $[\cdot]_i$, the *extrusion problem* consists in finding/constructing 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 $\uparrow_i : Con \rightarrow Con$ 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 $[\cdot]_i$. The extruded information c may not necessarily be part of the information residing in the space of agent i . For example, using properties of space and extrusion functions we shall see that $[d \sqcup \uparrow_i c]_i = [d]_i \sqcup c$ specifying that c is extruded (while d is still in the space of i). The extruded c could be inconsistent with d (i.e., $c \sqcup d = false$), it could be related to d (e.g., $c \sqsubseteq d$), or simply unrelated to d . From an epistemic perspective, we can use \uparrow_i to express *utterances* by agent i and such utterances could be intentional lies (i.e., inconsistent with their beliefs), informed opinions (i.e., derived from the beliefs), or simply arbitrary statements (i.e., unrelated to their beliefs). One can then think of extrusion/utterance as the *right inverse* of space/belief.

Modal Logic. Modal logics [7] extend propositional logic with $n \geq 1$ operators $\Box_1, \Box_2, \dots, \Box_n$, 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. For example, in doxastic modal logic, the logic of belief, the formula $\Box_i \phi$ (often written as $B_i \phi$) specifies that agent i believes ϕ and the formula $\Box_i \neg \Box_j \psi$ specifies that agent i believes that the agent j does not believe ψ . We shall also be interested in inverse modalities. An operator \Box_i^{-1} is a (*right*) *inverse modality* for \Box_i if the formula $\Box_i \Box_i^{-1} \phi$ is logically equivalent to ϕ . Inverse operators arise as, among others, past operators in temporal logic [8], utterances in epistemic logic [9],

and backward moves in modal logic for concurrency [10]

Kripke Semantics. The most representative semantic models for modal logics are Kripke Structures (KS) [11]. A KS M can be represented as a state graph with $n \geq 1$ transition relations $\xrightarrow{1}_M, \xrightarrow{2}_M, \dots, \xrightarrow{n}_M$ and a function π_M that specifies the set of propositions $\pi_M(s)$ that are true at each state (or world) s of M . A *pointed* KS is a pair (M, s) where s is a state of M . We shall say that (M, s) is *model of* (or *satisfies*) a propositional formula p if $p \in \pi_M(s)$. Boolean operators are defined as usual; (M, s) is a model of $\phi \wedge \psi$ if it is a model of both ϕ and ψ , and it is a model of $\neg\phi$ if it is not model of ϕ . For the modal case, (M, s) is said to be a model of $\Box_i\phi$ if (M, t) is a model of ϕ for every t reachable from s through an i -labelled transition; i.e. $s \xrightarrow{i}_M t$. We shall use $\llbracket \phi \rrbracket$ to denote the set of all models of ϕ . Different families of KS give rise to different modal logics. For example, the theorems of the S5 epistemic logic are those modal formulae that are satisfied by all pointed KS whose transition relations are equivalences.

Normal Modal Operators. In modal logic one is often interested in *normal* modalities: Roughly, a modal operator \Box_i is normal in a given modal logic system if (1) $\Box_i\phi$ is a theorem whenever ϕ is a theorem, and (2) $\Box_i(\phi \Rightarrow \psi) \Rightarrow (\Box_i\phi \Rightarrow \Box_i\psi)$ is a theorem. Normal modalities are ubiquitous in logic; e.g., the box operator in the the logic system **K** and its extensions [11], and the always and next operators as well as the weak past operator from temporal logic [12], the necessity operator from Hennessy-Milner (HM) logic [13], the knowledge operator from epistemic logic, the belief operator from doxastic logic [14] are all normal. In fact, any operator \Box_i whose models are defined with the above-mentioned Kripke semantics, is normal.

This paper. Although the notion of spatial constraint system is intended to give an algebraic account of spatial and epistemic assertions, we shall show in this paper that it is sufficiently robust to give an algebraic account of more general *modal logic* assertions. The main focus of this paper is the study of the above-mentioned *extrusion problem*

for meaningful families of SCS's called *Kripke spatial constraint systems* introduced in [1].

The constraints (or elements) of Kripke SCS's are sets of pointed KS, i.e., models of modal logics formulae, ordered by reversed inclusion. Consequently, they can be used as semantic domains for modal logic in a natural way. For example, let us suppose that the set of models $\llbracket \phi \rrbracket$ is a constraint in a Kripke SCS. The set $\llbracket \Box_i \phi \rrbracket$ can be compositionally obtained, using the space function $[\cdot]_i$ of agent i , as the constraint $\llbracket \llbracket \phi \rrbracket \rrbracket_i$ in the Kripke SCS. Furthermore, if there exists an extrusion function \uparrow_i for the space function $[\cdot]_i$ (see extrusion problem above) then we can derive an inverse modality \Box_i^{-1} for \Box_i by defining $\llbracket \Box_i^{-1} \phi \rrbracket$ as $\uparrow_i \llbracket \phi \rrbracket$.

Contributions. This paper contributes to the theory of constraint systems [2] by giving an algebraic characterization of the notion of normal modalities as space functions and by providing a complete taxonomy of extrusion functions from Kripke SCS's. It also contributes to the theory of modal logic by using this characterization and taxonomy to prove new expressiveness results for various modal systems and related notions. More precisely, our contributions are the following:

1. We give an algebraic characterization of the notion of *normality* in modal logic by building upon work on Geometric Logic [15] and show that this abstract notion corresponds exactly to preservation of the join operation.
2. 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.
3. We give a characterization and derivations of extrusion functions that are normal (and thus they correspond to normal inverse modalities). In particular we identify the *maximum* normal extrusion function and a family of *minimal* normal extrusion functions.
4. Finally we discuss the application of our results in the context of specific modal systems and related concepts such as the minimal logic system \mathbf{K} [11], the modal

logic of linear-time temporal logic (LTL) [12], and bisimilarity. In particular, we prove that there is no reverse modality for the box operator of the logic system \mathbf{K} nor for the always operator of LTL and we also derive existing inverse modalities for the next operator of LTL. We use our taxonomy results to show that, unlike HM-logic formulae, bisimilarity is not preserved by formulae in HM-logic extended with a reverse modality for the necessity operator: I.e., there are bisimilar pointed KS that do not satisfy the same formulae. This shows that the extended logic is more expressive than HM-logic.

This submission is the extended and revised version of [16] for the special issue of selected papers from *ICTAC 2016*. In this version we have included all proofs as well as new material in Section 5 corresponding to the derivation of *all* minimal normal right inverses. The work in [16] does not study minimal right inverses. This submission thus offers a complete taxonomy of normal right inverses. The new material is also used in the Applications section to prove an expressiveness result in the context of bisimilarity invariance in Section 6.4.

In previous work [17] and its extended version [18] the authors derived an inverse modality but only for the specific case of a logic of belief and only a sufficient (but not necessary) condition was identified. Unlike the present work, [17, 18] were neither concerned with giving a *complete* characterization of the existence of right inverses, a complete taxonomy of extrusion functions, nor deriving *normal* inverses.

Organization.

This article is structured as follows. In Section 2 we recall the notions of constraint system and spatial constraint system. In Section 3 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. Section 4 and Section 5 are the core sections of the paper. In Section 4 we identify necessary and sufficient conditions for the existence of right inverses of space functions. In Section 5 we turn our attention to the derivation and classification of normal right inverses. Finally, in Section 6 we discuss and apply our results in the context of the modal system \mathbf{K} , LTL, and bisimilarity.

For the sake of clarity and readability this paper includes an index table for notation and a table of contents.

2. Background: Spatial Constraint Systems

In this section we recall the notion of basic constraint system [3] and the more recent notion of spatial constraint system [1]. We presuppose basic knowledge of order theory and modal logic [19, 7, 11, 20].

The concurrent constraint programming model of computation [2] 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*.

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

Notation 2.1 (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) (also called supremum or join) of the elements in S if it exists, and similarly $\bigsqcap S$ is the greatest lower bound (GLB) (also called infimum or meet) of the elements in S if it exists. We say that \mathbf{C} is a complete lattice iff each subset of Con has a LUB and a GLB in Con . A non-empty set $S \subseteq Con$ is directed iff every finite subset of S has an upper 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 $f : Con \rightarrow Con$. Let (Con, \sqsubseteq) be a complete lattice. The self-map f on Con preserves the LUB of a set $S \subseteq Con$ iff $f(\bigsqcup S) = \bigsqcup \{f(c) \mid c \in S\}$. The preservation of the GLB of a set is defined analogously. We say f preserves finite/infinite suprema iff it preserves the LUB of arbitrary finite/infinite sets. Preservation of finite/infinite infima is defined similarly. The order \sqsubseteq extends naturally to self-maps; we shall write $f \sqsubseteq g$, where f and g are self-maps on Con if and only if $f(c) \sqsubseteq g(c)$ for every $c \in Con$.*

Constraint systems can be formalized as *complete algebraic lattices* [3]¹. The elements of the lattice, the *constraints*, represent (partial) information. A constraint c can be viewed as an *assertion*. The lattice order \sqsubseteq is meant to capture (reverse) entailment of information: $c \sqsubseteq d$, alternatively written $d \sqsupseteq c$, means that the assertion d represents at least 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 2.2 (Constraint Systems [3]). *A constraint system (CS) \mathbf{C} is a complete algebraic² 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.* □

We shall now describe two typical concrete constraint systems taken from [18, 3, 2].

Example 2.3 (Herbrand Constraint System [2, 3, 18]). *The Herbrand CS captures syntactic equality between terms t, t', \dots built from a first-order alphabet \mathcal{L} with variables x, y, \dots , function symbols, and equality $=$. The constraints are (equivalent classes of) sets of equalities over the terms of \mathcal{L} : E.g., $\{x = t, y = t\}$ is a constraint. The relation $c \sqsubseteq d$ holds if the equalities in c follow from those in d : E.g., $\{x = y\} \sqsubseteq \{x = t, y = t\}$. The constraint *false* is the set of all term equalities in \mathcal{L} and *true* is (the equivalence class of) the empty set. The compact elements are the*

¹An alternative syntactic characterization of CS, akin to Scott information systems, is given in [2].

²Recall CS's can also be used as semantic domains for concurrent constraint programming (CCP) languages. The restriction to algebraic lattices is for semantic purposes and a full discussion of this technical issue is beyond the scope of this paper (the reader is referred to [3]). Briefly speaking, CCP programs can post or query constraints but these constraints must be compact (finite). The result of a possibly infinite computation is the join of all the (compact) constraints posted during the computation. Algebraicity of constraint systems [3] means that any constraint is the join of compact constraints below it. This assumption is needed for correspondence results between the denotational semantics and their operational counterpart.

(equivalence class of) finite sets of equalities. The LUB is the (equivalence class of) set union. \square

In the above example constraints are represented as set of equations and thus the join (LUB) of constraints corresponds to the equivalent class of *union* of their equations. We can also view a constraint c as a representation of a set of variable assignments [21]. For instance a constraint $x > 42$ can be thought of as the set of assignments mapping x to a value greater than 42; i.e., the solutions to (or models of) $x > 42$. In this case the join of constraints naturally corresponds to the *intersection* of their assignments, *false* as the empty set of assignments, and *true* as the set of all assignments.

Example 2.4 (Boolean Constraint System [18, 21]). *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 $(\mathcal{P}(\mathcal{A}(\Phi)), \supseteq)$: 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. \square*

Notice that logic propositions can be straightforwardly interpreted as constraints in $\mathbf{B}(\Phi)$. Let $\mathcal{L}_0(\Phi)$ be the propositional language built from Φ by the grammar

$$\phi, \psi, \dots := p \mid \phi \wedge \psi \mid \neg\phi \quad (1)$$

where $p \in \Phi$. We shall use the classical abbreviations $\phi \vee \psi$ for $\neg(\neg\phi \wedge \neg\psi)$, $\phi \Rightarrow \psi$ for $\neg\phi \vee \psi$, *ff* for $p \wedge \neg p$, and *tt* for $\neg\text{ff}$. A boolean assignment π *satisfies* ϕ iff $\pi \models \phi$ where \models is defined inductively as follows: $\pi \models p$ iff $\pi(p) = 1$, $\pi \models \phi \wedge \psi$ iff $\pi \models \phi$ and $\pi \models \psi$, and $\pi \models \neg\phi$ iff $\pi \not\models \phi$. We interpret each formula ϕ as the constraint $\mathbf{B}[\phi] \stackrel{\text{def}}{=} \{\pi \in \mathcal{A}(\Phi) \mid \pi \models \phi\}$ in $\mathbf{B}(\Phi)$. Clearly $\mathbf{B}[\phi] \sqsubseteq \mathbf{B}[\psi]$ holds iff $\psi \Rightarrow \phi$ is valid, i.e., satisfied by every truth assignment.

Spatial Constraint Systems. The authors of [1] extended the notion of CS to account for distributed and multi-agent scenarios where agents have their own space for keeping their local information and performing their computations.

Let $G = \{1, \dots, n\}$ be an underlying set of agents. Intuitively, each agent $i \in G$ has a *space* function $[\cdot]_i$ from constraints to constraints. Recall that constraints can be viewed as assertions. Thus, given a constraint c , we can 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). Both interpretations convey the idea that c is local to agent i . Similarly, $[[c]_j]_i$ is 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* spaces that belong to agents i and j , respectively. From a computational/ concurrency point of view, we think of \sqcup as parallel composition. As mentioned before, from a logic point of view the join of information \sqcup corresponds to conjunction.

Definition 2.5 (Spatial Constraint System [1]). *An n -agent ($n > 0$) spatial constraint system (n -SCS) \mathbf{C} is a CS (Con, \sqsubseteq) equipped with n self-maps $[\cdot]_1, \dots, [\cdot]_n$ 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 to. Notice that requiring S.1 and S.2 is equivalent to requiring that each $[\cdot]_i$ preserves *finite suprema*. Also S.2 implies that each $[\cdot]_i$ is monotonic: I.e., if $c \sqsupseteq d$ then $[c]_i \sqsupseteq [d]_i$.

Remark 2.6. *Notice that S.1 and S.2 are the only requirements for space functions. For example, nothing prevents us from having $[false]_i \neq false$. Intuitively, inconsistencies generated by an agent may be confined within its own space. It is also possible to have $[c]_i \sqcup [d]_j \neq false$ even when $c \sqcup d = false$, i.e., we may have agents whose information is inconsistent with that of other agents. We could also have $[c]_i = [d]_i$ for*

some c, d such that $c \neq d$. Depending on the intended model this could be interpreted as saying that agent i cannot distinguish c from d .

The following SCS instance will be properly defined in Section 4.1. We find it convenient to briefly describe it in the following example to give some intuitions about space functions.

Example 2.7 (Kripke SCS's [1]). *In Example 2.4 we illustrated the Boolean CS $\mathbf{B}(\Phi)$ whose elements represent sets of boolean assignments ordered by reversed inclusion. This allowed us to interpret each boolean formula ϕ , built from Φ , as the constraint $\mathbf{B}[\phi]$ including all the assignments that satisfy ϕ .*

Similarly, we can provide a family of SCS's, called Kripke SCS's [1], whose constraints are sets of pointed Kripke structures ordered by reversed inclusion. Recall that pointed Kripke structures, briefly described in the introduction and defined in Section 4.1, are the standard models of modal logic. In a Kripke SCS each constraint c is equated to a set of pointed KS pairs of the form (M, s) . The order \sqsubseteq is \supseteq , false is \emptyset , true is the set of all pointed KS under consideration and the join operator \sqcup is \cap . The space functions are given by $[c]_i = \{(M, s) \mid \forall t : s \xrightarrow{i}_M t, (M, t) \in c\}$. This allows us to interpret any modal formula as a constraint; the set of pointed KS that satisfy the formula. In particular if $\llbracket \phi \rrbracket$ is the constraint interpretation in a Kripke SCS then $\llbracket \llbracket \phi \rrbracket \rrbracket_i$ is the interpretation of $\Box_i \phi$.

To give a spatial intuition about the definition of $[c]_i$ in a Kripke SCS, let us consider states (or worlds) s and t such that $s \xrightarrow{i}_M t$. Intuitively, $s \xrightarrow{i}_M t$ means that t is a world that agent i considers possible in the world s (of a KS M). In spatial terms we can think of t as being a local world for agent i wrt to the outside world s . Then if the constraint $[c]_i$ holds true in the outside world s of agent i (wrt M), i.e., $(M, s) \in [c]_i$ then c must be true in their local world t (wrt M), i.e., $(M, t) \in c$.

We refer the reader to Section 4.1 to see Kripke SCS's in full detail. □

Extrusion and utterance. We can also equip each agent i with an *extrusion* function $\uparrow_i : \text{Con} \rightarrow \text{Con}$. Intuitively, within a space context $[\cdot]_i$, the assertion $\uparrow_i c$ specifies that

c must be posted outside of agent i 's space. This is captured by requiring the following *extrusion* axiom:

$$(E.1) \quad [\uparrow_i c]_i = c. \quad (2)$$

In other words, we view *extrusion/utterance* as the right inverse of *space/belief* (and thus space/belief as the left inverse of extrusion/utterance).

Definition 2.8 (Extrusion). *Given an n -SCS $(Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$, 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 Axiom E.1 in Eq. 2 is satisfied. \square*

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., $c \sqcup d = \text{false}$, we can see the utterance as an intentional *lie* by agent i : The agent i utters an assertion inconsistent with their own beliefs.

Example 2.9. *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 .*

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 $f : X \rightarrow Y$ 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 [9] gives some constructive extrusion functions. It also identifies a distinctive property of space functions for which a right inverse exists.

Proposition 2.10 ([9]). *Let $[\cdot]_i$ be a space function of SCS. Then*

1. *If $[\text{false}]_i \neq \text{false}$ 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 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 2.10 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 \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. The notion of *normality* is important because it is in normal modal logics where the axiom K (i.e. $\Box_i(\phi \rightarrow \psi) \rightarrow (\Box_i\phi \rightarrow \Box_i\psi)$) is satisfied, which allow us to provide modal logics with a Kripke semantics.

3. 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 [15] 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 the maps preserving finite suprema.

We can define a general form of implication by adapting the corresponding notion from Heyting Algebras to constraint systems. Intuitively, a *Heyting implication* $c \rightarrow d$ in our setting corresponds to the *weakest constraint* one needs to join c with to derive d . Similarly, the negation of a constraint c , written $\sim c$, can be seen as the *weakest constraint inconsistent* with c .

Definition 3.1 (Constraint Frames). *A constraint system (Con, \sqsubseteq) is said to be a constraint frame iff its LUBs distribute over arbitrary meets: For every $c \in Con$ and $S \subseteq Con$ we have $c \sqcup \prod S = \prod \{c \sqcup e \mid e \in S\}$. Given a constraint frame (Con, \sqsubseteq) and $c, d \in Con$, define Heyting implication $c \rightarrow d$ as $\prod \{e \in Con \mid c \sqcup e \sqsubseteq d\}$ and Heyting negation as $\sim c \stackrel{\text{def}}{=} c \rightarrow \text{false}$. \square*

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

Proposition 3.2. *Let (Con, \sqsubseteq) be a constraint frame. For every $c, d, e \in Con$:*

1. $c \sqcup (c \rightarrow d) = c \sqcup d$
2. $c \sqsubseteq (d \rightarrow e)$ iff $c \sqcup d \sqsubseteq e$
3. $c \rightarrow d = \text{true}$ iff $c \sqsubseteq d$

Proof. The proof proceeds as follows:

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

This is proved in Lemma 1 (Modus Ponens) of [18]. For self-containedness we have included the proof of this item.

We need to prove $c \sqcup \prod \{e \mid d \sqsubseteq e \sqcup c\} = c \sqcup d$. Recall that by definition LUBs distribute over arbitrary meets in any frame.

- First we prove $c \sqcup \prod \{e \mid d \sqsubseteq e \sqcup c\} \sqsubseteq c \sqcup d$. Let $S = \{e \mid d \sqsubseteq e \sqcup c\}$. Since $d \in S$, we conclude $\prod S \sqsubseteq d$. Thus, $c \sqcup \prod S \sqsubseteq c \sqcup d$ as wanted.

- We now prove $c \sqcup d \sqsubseteq c \sqcup \prod\{e \mid d \sqsubseteq e \sqcup c\}$. Let $S = \{e \mid d \sqsubseteq e \sqcup c\}$. Distributing the LUB over the meet we obtain $c \sqcup \prod S = \prod\{c \sqcup e \mid e \in S\}$. Since each $c \sqcup e \sqsupseteq d$ for every $e \in S$ then $d \sqsubseteq \prod\{c \sqcup e \mid e \in S\} = c \sqcup \prod S$. Therefore $c \sqcup d \sqsubseteq c \sqcup c \sqcup \prod S = c \sqcup \prod S$.

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

- Assume $c \sqsupseteq (d \rightarrow e)$. Then we have $c \sqcup d \sqsupseteq (d \rightarrow e) \sqcup d$. Therefore, from Proposition 3.2 (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 $d \rightarrow e = \prod S$ where $S = \{a \mid a \sqcup d \sqsupseteq e\}$, and $c \in S$, then $c \sqsupseteq (d \rightarrow e)$.

3. $c \rightarrow d = true$ iff $c \sqsupseteq d$. By Proposition 3.2 (2), we know that $e \sqsupseteq (c \rightarrow d)$ iff $e \sqcup c \sqsupseteq d$. Let us take $e = true$. Then, $true \sqsupseteq (c \rightarrow d)$ iff $true \sqcup c \sqsupseteq d$. Therefore, $true = (c \rightarrow d)$ iff $c \sqsupseteq d$ as wanted.

□

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

Example 3.3. A simple example can be obtained by taking $d = a \rightarrow b$ in Example 2.9, i.e., $e = [c \sqcup \uparrow_i[a]_j]_i \sqcup [a \rightarrow b]_j$. We can use Proposition 3.2 item 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 .

Example 3.4. A more complex example involves extrusion from i to j and back from j to i by letting a to be the conditional $d \rightarrow \uparrow_j[d]_i$ in the constraint $e = [c \sqcup \uparrow[a]_j]_i \sqcup [d]_j$ defined in Example 2.9. 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 $e \sqsupseteq [a \sqcup d]_i$. □

3.1. Normal Self-Maps

In modal logics one is often interested in *normal modal* operators. The formulae of a modal logic are those of propositional logic extended with modal operators. Roughly speaking, a modal logic operator m is normal iff (1) the formula $m(\phi)$ 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 $m(\phi \Rightarrow \psi) \Rightarrow (m(\phi) \Rightarrow m(\psi))$ is a theorem. Since constraints can be viewed as logic assertions, we can think of modal operators as self-maps on constraints. Thus, using Heyting implication³, we can express the normality condition in constraint frames as follows.

Definition 3.5 (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) $m(c \rightarrow d) \rightarrow (m(c) \rightarrow m(d)) = true$ for each $c, d \in Con$. \square*

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 3.5 is equivalent to the seemingly simpler condition: $m(c \sqcup d) = m(c) \sqcup m(d)$.

Theorem 3.6 (Normality & Finite Suprema). *Let \mathbf{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 , $\forall c, d \in Con : f(c \rightarrow d) \rightarrow (f(c) \rightarrow f(d)) = true$ 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 $\forall c, d \in Con : f(c \rightarrow d) \rightarrow (f(c) \rightarrow f(d)) = true$. 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 3.2(3) we obtain

$$f((c \sqcup d) \rightarrow d) \sqsupseteq f(c \sqcup d) \rightarrow f(d). \quad (3)$$

³Logical (boolean) implication is a particular instance of Heyting implication [15].

From Proposition 3.2(3) $(c \sqcup d) \rightarrow d = \text{true}$. Since $f(\text{true}) = \text{true}$ we have $f((c \sqcup d) \rightarrow d) = \text{true}$. We must then have, from Equation 3, $f(c \sqcup d) \rightarrow f(d) = \text{true}$ as well. Using Proposition 3.2(3) we obtain $f(c \sqcup d) \sqsupseteq f(d)$. In a similar fashion, by exchanging c and d in Equation 3, 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 3.2(3) we have

$$f(c \rightarrow (d \rightarrow c \sqcup d)) \sqsupseteq f(c) \rightarrow f(d \rightarrow c \sqcup d). \quad (4)$$

Using Proposition 3.2 one can verify that $c \rightarrow (d \rightarrow c \sqcup d) = \text{true}$. Since $f(\text{true}) = \text{true}$ we have $f(c \rightarrow (d \rightarrow c \sqcup d)) = \text{true}$. From Equation 4, we must then have $f(c) \rightarrow f(d \rightarrow c \sqcup d) = \text{true}$ and by using Proposition 3.2(3) we conclude $f(c) \sqsupseteq f(d \rightarrow c \sqcup d)$. From the assumption and Proposition 3.2(3) $f(d \rightarrow c \sqcup d) \sqsupseteq f(d) \rightarrow f(c \sqcup d)$. We then have $f(c) \sqsupseteq f(d \rightarrow c \sqcup d) \sqsupseteq f(d) \rightarrow f(c \sqcup d)$. Thus $f(c) \sqsupseteq f(d) \rightarrow f(c \sqcup d)$ and then using Proposition 3.2(2) we obtain $f(c) \sqcup f(d) \sqsupseteq f(c \sqcup d)$ as wanted.

- Assume that $\forall c, d \in \text{Con} : f(c \sqcup d) = f(c) \sqcup f(d)$. Take two arbitrary $c, d \in \text{Con}$. We shall prove $f(c \rightarrow d) \rightarrow (f(c) \rightarrow f(d)) = \text{true}$. From Proposition 3.2(2-3) it suffices to prove $f(c \rightarrow d) \sqcup f(c) \sqsupseteq f(d)$. Using the the assumption and Proposition 3.2(1) we obtain $f(c \rightarrow d) \sqcup f(c) = f(c \sqcup (c \rightarrow d)) = f(c \sqcup d) = f(c) \sqcup f(d) \sqsupseteq f(d)$ 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 3.7. *Let \mathbf{C} be a constraint frame $(\text{Con}, \sqsubseteq)$ and let f be a normal self-map on Con . If $c \sqsubseteq d$ then $f(c) \sqsubseteq f(d)$.*

3.2. Summary

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

4. Extrusion Problem for Kripke Constraint Systems

This and the next section are the main and more technical parts of the paper. We will study the extrusion/right inverse problem for a meaningful family of spatial constraint systems (SCS); 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) are a fundamental mathematical tool in logic and computer science. They 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 $\llbracket \phi \rrbracket$ 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 $\llbracket \Box_i \phi \rrbracket = \{(M, s) \mid \forall t : s \xrightarrow{i}_M t, (M, t) \in \llbracket \phi \rrbracket\}$.

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 $\llbracket \Box_i \phi \rrbracket = \llbracket \phi \rrbracket]_i$.

Inverse modalities \Box_i^{-1} , also known as a reverse modalities, are used in many modal logics. In tense logics they represent past operators [22] and in epistemic logic they represent utterances [9]. 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 $\llbracket \Box_i^{-1}\phi \rrbracket = \uparrow_i(\llbracket \phi \rrbracket)$ (thus $\llbracket \Box_i(\Box_i^{-1}\phi) \rrbracket = \llbracket \phi \rrbracket$). Therefore deriving extrusion functions and establishing the weakest conditions under which they exist 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 3.5).

4.1. KS and Kripke SCS

We begin by recalling some notions and notations related to Kripke models.

Definition 4.1 (Kripke Structures). *An n -agent ($n > 0$) Kripke Structure (KS) M over a set of primitive propositions Φ is a tuple $(S, \pi, \mathcal{R}_1, \dots, \mathcal{R}_n)$ where*

- S is a non-empty set of states,
- $\pi : S \rightarrow (\Phi \rightarrow \{0, 1\})$ 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 . □

We shall use the following notation in the rest of the paper.

Notation 4.2. *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 through \mathcal{R}_i from the pointed KS (M, s) . 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 .*

We now define the Kripke SCS w.r.t. a set $\mathcal{S}_n(\Phi)$ of pointed KS. Notice that in Example 2.4 constraints represented sets of boolean assignments, which allowed us to interpret each propositional formula as a constraint (i.e. the set of assignments satisfying the formula). Similarly, in a Kripke SCS constraints are sets of (pointed) KS models. This will allow us to interpret each modal formula as constraints.

Definition 4.3 (Kripke Spatial Constraint Systems [1]). *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 $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ where $Con = \mathcal{P}(\Delta)$, $\sqsubseteq = \supseteq$, and for every $i \in \{1, \dots, n\}$:

$$[c]_i \stackrel{\text{def}}{=} \{(M, s) \in \Delta \mid \blacktriangleright_i(M, s) \subseteq c\} \quad (5)$$

□

In the Kripke-based semantics of modal logic (Section 6) 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 4.4. *The structure $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ is a complete algebraic lattice given by a powerset ordered by reversed inclusion \supseteq [23]. The LUB (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 LUBs are intersections so the requirement of distributivity is satisfied. Furthermore, each $[\cdot]_i$ preserves arbitrary suprema (intersection) and thus, from Theorem 3.6 it is a normal self-map.*

Proposition 4.5 ([18]). *Let $\mathbf{K}(\mathcal{S}_n(\Phi)) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ as in Definition 4.3.*

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

4.2. 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 4.6. *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 4.7 (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.) □

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

Example 4.8. *Figure 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 1.(ii) and Figure 1.(iii)). Also some non-tree like structures such as Figure 1.(i) and Figure 1.(v). Figure 1.(iv) shows a non determinant-complete accessibility relation by taking the transitive closure of Figure 1.(iii). □*

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

Notation 4.9. *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*

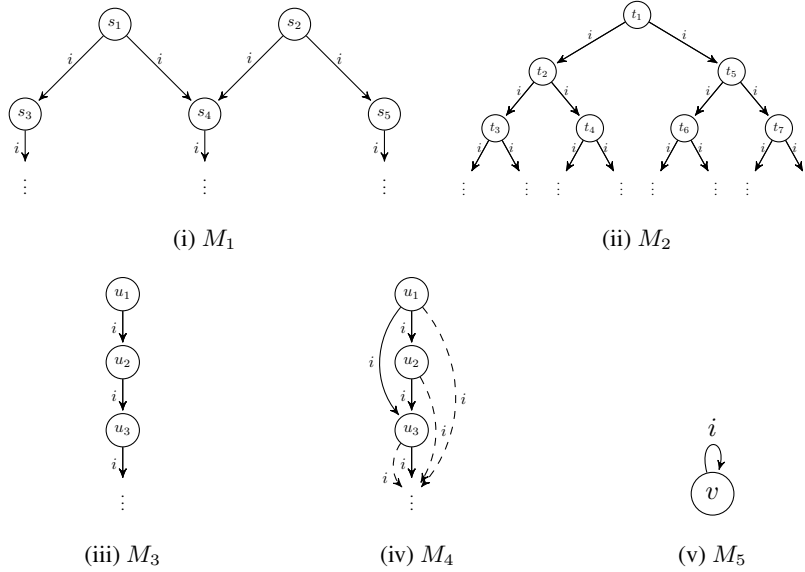


Figure 1: Accessibility relations for an agent i . In each sub-figure we omit the corresponding KS M_k from the edges and draw $s \xrightarrow{i} t$ whenever $s \xrightarrow{i}_{M_k} t$.

of states as $\blacktriangleright_i(M, S) = \bigcup_{s \in S} \blacktriangleright_i(M, s)$. Furthermore, we shall write $s \xrightarrow{i}_M t$ to mean that s uniquely determines t w.r.t. \xrightarrow{i}_M . Analogously, we define $\triangleright_i(M, s) \stackrel{\text{def}}{=} \{(M, t) \mid s \xrightarrow{i}_M t\}$.

Proposition 4.10. *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. The proof proceeds as follows :

- Assume that $s \in S_M$ is determinant w.r.t. \xrightarrow{i}_M . Let us proceed by contradiction and assume that there exists an $S' \subseteq S_M$ 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 exist 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 $\blacktriangleright_i(M, s) \subseteq \blacktriangleright_i(M, S')$.
- 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 $S'' = \{s_1, s_2, \dots\}$. Notice that $\blacktriangleright_i(M, s) \subseteq \blacktriangleright_i(M, S'')$. Applying the assumption, we conclude $s \in S''$, a contradiction. □

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

Theorem 4.11 (Completeness). *Let $[\cdot]_i$ be a space function of a Kripke SCS $\mathbf{K}(S)$. 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.

- 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 we conclude $d \subseteq [c]_i$. It remains to prove $d \supseteq [c]$. Suppose $d \not\supseteq [c]$. Since $d \subseteq [c]$ we have $d \subset [c]$. Then there must be an (M', s') , with $M' \in \mathcal{S}$, such that $(M', s') \notin d$ and $(M', s') \in [c]$. But if $(M', s') \in [c]_i$ then from Definition 4.3 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 4.10 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.
- 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 4.10 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,

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 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}^D to denote the class of KS's whose accessibility relations are determinant-complete. It follows from Theorem 4.11 that $\mathcal{S} = \mathcal{M}^D$ is the largest class for which space functions of a Kripke $\mathbf{K}(\mathcal{S})$ have right inverses.

4.3. Deriving Maximum Right-Inverse

Let $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ be the Kripke SCS. The Axiom of Choice and Theorem 4.11 tell us that each $[\cdot]_i$ has a right inverse (extrusion function) if and only if $\mathcal{S} \subseteq \mathcal{M}^D$. We are interested, however, in explicit constructions of the right inverses.

Remark 4.12. *Recall, from Remark 4.4, that any Kripke SCS $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ is ordered by reversed inclusion (i.e., $\sqsubseteq = \supseteq$). Thus, for example, saying that some f is the least function w.r.t. \sqsubseteq satisfying certain conditions is equivalent to saying that f is the maximum 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 4.5), we can apply Proposition 2.10.2 to obtain the following canonical maximum right-inverse construction. Recall that $[c]_i^{-1} = \{d \mid c = [d]_i\}$ denotes the pre-image of c under $[\cdot]_i$.

Definition 4.13 (Maximum Right Inverse). *Let $\mathbf{K}(\mathcal{S}) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ be a Kripke SCS over $\mathcal{S} \subseteq \mathcal{M}^D$. We define \uparrow_i^M as the following self-map on Con : $\uparrow_i^M : c \mapsto \bigsqcup [c]_i^{-1}$.* □

It follows from Proposition 2.10.2 that \uparrow_i^M is a right inverse of $[\cdot]_i$, and furthermore, from its definition it is clear that \uparrow_i^M is the maximum right inverse of $[\cdot]_i$ w.r.t. \sqsubseteq .

4.4. Summary

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 4.11). In the rest of the paper \mathcal{M}^D is used to denote the class of KS's whose accessibility relations are determinant-complete; \mathcal{M}^D is thus the largest class for which space functions of a Kripke SCS have right inverses. We shall also assume Kripke SCS's $\mathbf{K}(S) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ with $S \subseteq \mathcal{M}^D$.

We also derived \uparrow_i^M as the maximum right inverse of $[\cdot]_i$ w.r.t. the underlying Kripke SCS order. As shown next, however, \uparrow_i^M is not necessarily *normal* in the sense of Definition 3.5. In the next section we shall derive and classify normal right inverses.

5. 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 in Section 6 the Kripke semantics of several inverse modalities in the literature corresponds to normal right inverses of space functions. The notion of indeterminacy and multiple determinacy introduced in Definition 5.1 will play a central role.

Let us first extend the terminology in Definition 4.7.

Definition 5.1 (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 undetermined (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} . Additionally, we shall say that t is uniquely determined if it is determined by only one state in S w.r.t. \mathcal{R} .*

(As in Definition 4.7, we shall often omit “w.r.t. \mathcal{R} ” when no confusion can occur.) \square

The following statement and Theorem 3.6 lead us to conclude that the presence of undetermined/initial states or multiple-determined states causes \uparrow_i^M not to be normal.

Proposition 5.2. Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ and \uparrow_i^M as in Definition 4.13. Let $\text{nd}(\mathcal{S}) = \{(M, t) \mid M \in \mathcal{S} \text{ \& } t \text{ is undetermined w.r.t. } \xrightarrow{i}_M\}$ and $\text{md}(\mathcal{S}) = \{(M, t) \mid M \in \mathcal{S} \text{ \& } t \text{ is multiply determined w.r.t. } \xrightarrow{i}_M\}$:

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

Proof.

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

Assume $\text{nd}(\mathcal{S}) \neq \emptyset$. Take c to be the complement of $\text{nd}(\mathcal{S})$ in Con . Using Equation 5 we have (1) $[c]_i = \text{true}$ and from the assumption (2) $c \neq \text{true}$ (recall that true is the set of all pointed KS, see Remark 4.4). From (1) and (2) we have $\uparrow_i^M(\text{true}) = \bigsqcup [c]_i^{-1} \neq \text{true}$.

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

Suppose $\text{md}(\mathcal{S}) \neq \emptyset$. Take $t \in \text{md}(\mathcal{S})$, 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^M is a right inverse of $[\cdot]_i$. Because of this any constraint $e = [\uparrow_i^M(e)]_i = \{(M, s'') \mid \blacktriangleright_i(M, s'') \subseteq \uparrow_i^M(e)\}$, therefore $(M, t) \in \uparrow_i^M(c) \cap \uparrow_i^M(d)$. Now, $\uparrow_i^M(c \cap d) = \uparrow_i^M(\text{false}) = \text{false}$ because Proposition 2.10 and $\uparrow_i^M(\text{false}) = \bigsqcup [\text{false}]_i^{-1}$ (recall that false is the empty set, see Remark 4.4). We therefore conclude $\uparrow_i^M(c \sqcup d) = \uparrow_i^M(c \cap d) \neq \uparrow_i^M(c) \cap \uparrow_i^M(d) = \uparrow_i^M(c) \sqcup \uparrow_i^M(d)$.

□

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

Lemma 5.3. Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ be the Kripke SCS over $\mathcal{S} \subseteq \mathcal{M}^D$. Suppose that f is a normal right inverse of $[\cdot]_i$. Then for every $M \in \mathcal{S}$, $c \in \text{Con}$:

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. $true \subseteq f(true)$.

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 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 $\blacktriangleright_i(M, s) \subseteq f(c)$ as wanted.
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 if there exists s, s' with $s \neq s'$ such that $s \xrightarrow{i}_M t$ and $s' \xrightarrow{i}_M t$. From Corollary 3.7 we know if $c \subseteq d$ then $f(c) \subseteq f(d)$. Then it follows that $false \subseteq d$ and $f(false) \subseteq f(d)$. Thus, it suffices to prove that $(M, t) \in f(false)$. From the assumption that f is a normal right inverse then by Theorem 3.6 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 5.3 (1) $s \xrightarrow{i}_M t$ and $s' \xrightarrow{i}_M t$ we know $\{(M, t)\} \subseteq f(\{(M, s)\}) \cap f(\{(M, s')\})$. Therefore by Corollary 3.7, $\{(M, t)\} \subseteq f(\{(M, s)\} \cap \{(M, s')\})$ which is the same as $\{(M, t)\} \subseteq f(false)$. Then, we know $f(false) \subseteq f(c)$ therefore $\{(M, t)\} \subseteq f(c)$ as wanted.
3. $true \subseteq f(true)$. Trivial from the assumption that f is a normal-right inverse.

□

As stated in the above lemma, for 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) where t is uniquely determined w.r.t. \xrightarrow{i}_M by some s with $(M, s) \in c$.

5.1. Deriving Maximum Normal Right Inverse

Lemma 5.3 above tells us what 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 5.3 gives us a *complete* set of conditions for normal right inverses. In fact, the least self-map f w.r.t. \sqsubseteq , i.e., the maximum one w.r.t. the lattice order \sqsubseteq , satisfying Conditions 1,2 and 3 in Lemma 5.3 is indeed a normal right inverse. We shall call such a function the *maximum normal right inverse* \uparrow_i^{MN} .

Definition 5.4 (Maximum Normal-Right Inverse). *Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ be a Kripke SCS over $\mathcal{S} \subseteq \mathcal{M}^D$. We define the maximum normal right inverse for agent i , \uparrow_i^{MN} as the following self-map on Con :*

$$\uparrow_i^{\text{MN}}(c) \stackrel{\text{def}}{=} \begin{cases} \text{true} & \text{if } c = \text{true} \\ \{(M, t) \mid t \text{ is determined w.r.t. } \xrightarrow{i}_M \text{ \& } \forall s : s \xrightarrow{i}_M t, (M, s) \in c\} & \end{cases} \quad (6)$$

(Recall that $s \xrightarrow{i}_M t$ means that s uniquely determines t w.r.t. \xrightarrow{i}_M .) \square

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

Theorem 5.5. *Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ and \uparrow_i^{MN} as in Definition 5.4.*

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

Proof.

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

To prove it is a right inverse we prove $[\uparrow_i^{\text{MN}}(c)]_i = c$. From Theorem 3.6 to prove that is a normal map it suffices to show that $\uparrow_i^{\text{MN}}(\text{true}) = \text{true}$ and $\uparrow_i^{\text{MN}}(c \sqcup d) = \uparrow_i^{\text{MN}}(c) \sqcup \uparrow_i^{\text{MN}}(d)$ (equivalent to proving $\uparrow_i^{\text{MN}}(c \cap d) = \uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d)$).

- $[\uparrow_i^{\text{MN}}(c)]_i = c$. Suppose $(M, s) \in [\uparrow_i^{\text{MN}}(c)]_i$. Then $\blacktriangleright_i(M, s) \subseteq \uparrow_i^{\text{MN}}(c)$ from Definition 4.3. Since s is determinant w.r.t. \xrightarrow{i}_M , there exists t s.t. $s \xrightarrow{i}_M t$ and $(M, t) \in \uparrow_i^{\text{MN}}(c)$. Therefore by definition of \uparrow_i^{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 $\blacktriangleright_i(M, s) \subseteq \uparrow_i^{\text{MN}}(c)$, and consequently $(M, s) \in [\uparrow_i^{\text{MN}}(c)]_i$.

- $\uparrow_i^{\text{MN}}(\text{true}) = \text{true}$. By definition of \uparrow_i^{MN} .
- $\uparrow_i^{\text{MN}}(c \cap d) = \uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d)$. We first prove monotonicity (i.e. if $c \subseteq d$ then $\uparrow_i^{\text{MN}}(c) \subseteq \uparrow_i^{\text{MN}}(d)$). For this, suppose $c \subseteq d$. If $(M, t) \in \uparrow_i^{\text{MN}}(c)$ then by definition of \uparrow_i^{MN} t is determined w.r.t. \xrightarrow{i}_M . If there exists a state s with $(M, s) \in c$ s.t. $s \xrightarrow{i}_M t$ then $(M, s) \in d$, thus $(M, t) \in \uparrow_i^{\text{MN}}(d)$. If such state does not exist, then t is multiply determined w.r.t. \xrightarrow{i}_M and consequently $(M, t) \in \uparrow_i^{\text{MN}}(d)$. We now proceed with the proof.
 - * $\uparrow_i^{\text{MN}}(c \cap d) \subseteq \uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d)$. Because $c \cap d \subseteq c$ and $c \cap d \subseteq d$. Then $\uparrow_i^{\text{MN}}(c \cap d) \subseteq \uparrow_i^{\text{MN}}(c)$ and $\uparrow_i^{\text{MN}}(c \cap d) \subseteq \uparrow_i^{\text{MN}}(d)$ by monotonicity of \uparrow_i^{MN} . Thus $\uparrow_i^{\text{MN}}(c \cap d) \subseteq \uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d)$.
 - * $\uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d) \subseteq \uparrow_i^{\text{MN}}(c \cap d)$. Suppose $(M, t) \in \uparrow_i^{\text{MN}}(c) \cap \uparrow_i^{\text{MN}}(d)$. Then from the definition of \uparrow_i^{MN} , t is determined w.r.t. \xrightarrow{i}_M of \uparrow_i^{MN} . If there exists s s.t. $s \xrightarrow{i}_M t$, then $(M, s) \in c$ and $(M, s) \in d$. Suppose there exists such state s . Then $(M, s) \in c \cap d$ and $(M, t) \in \uparrow_i^{\text{MN}}(c \cap d)$. If not, then t is multiply determined w.r.t. \xrightarrow{i}_M hence $(M, t) \in \uparrow_i^{\text{MN}}(c \cap d)$ by definition of \uparrow_i^{MN} .
- For every normal right inverse f of $[\cdot]_i$, we have $f \sqsubseteq \uparrow_i^{\text{MN}}$.

Suppose f is a normal right inverse. We then need to prove $\uparrow_i^{\text{MN}}(c) \subseteq f(c)$ for every c . Keep in mind that \sqsubseteq is \supseteq in $\mathbf{K}(\mathcal{S})$. It suffices to prove that if $(M, t) \in \uparrow_i^{\text{MN}}(c)$ then $(M, t) \in f(c)$. Take any $(M, t) \in \uparrow_i^{\text{MN}}(c)$. 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 $s \xrightarrow{i}_M t$ for some $(M, s) \in c$.

Suppose t is uniquely determined w.r.t. \xrightarrow{i}_M . Therefore, by Condition 1 of Lemma 5.3 (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 5.3 $(M, t) \in f(c)$.

□

Notice that $\uparrow_i^{\text{MN}}(c)$ excludes indetermined states if $c \neq \text{true}$. 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 5.6 (A Normal-Right Inverse). *Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq)$ be a Kripke SCS over $\mathcal{S} \subseteq \mathcal{M}^D$. Define $\uparrow_i^{\text{N}} : \text{Con} \rightarrow \text{Con}$ as $\uparrow_i^{\text{N}}(c) \stackrel{\text{def}}{=} \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in c\}$.* \square

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

Theorem 5.7. *Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ and \uparrow_i^{N} as in Definition 5.6. The self-map \uparrow_i^{N} is a normal right inverse of $[\cdot]_i$.*

Proof. To prove that \uparrow_i^{N} is a right inverse we prove $[\uparrow_i^{\text{N}}(c)]_i = c$. To prove that it is a normal map, by Theorem 3.6, it suffices to prove $\uparrow_i^{\text{N}}(\text{true}) = \text{true}$ and $\uparrow_i^{\text{N}}(c \sqcup d) = \uparrow_i^{\text{N}}(c) \sqcup \uparrow_i^{\text{N}}(d)$ (equivalent to proving $\uparrow_i^{\text{N}}(c \cap d) = \uparrow_i^{\text{N}}(c) \cap \uparrow_i^{\text{N}}(d)$).

- $[\uparrow_i^{\text{N}}(c)]_i = c$. Suppose $(M, s) \in [\uparrow_i^{\text{N}}(c)]_i$. Then $\blacktriangleright_i(M, s) \subseteq \uparrow_i^{\text{N}}(c)$ from Definition 4.3. As s is determinant w.r.t. \xrightarrow{i}_M , there exists t s.t. $s \xrightarrow{i}_M t$ and $(M, t) \in \uparrow_i^{\text{N}}(c)$. Therefore by definition of \uparrow_i^{N} , $(M, s) \in c$. 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 $\blacktriangleright_i(M, s) \subseteq \uparrow_i^{\text{N}}(c)$, and consequently $(M, s) \in [\uparrow_i^{\text{N}}(c)]_i$.
- $\uparrow_i^{\text{N}}(\text{true}) = \text{true}$. We have $\uparrow_i^{\text{N}}(\text{true}) = \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in \text{true}\}$. Since $(M, s) \in \text{true}$ trivially holds, $(M, t) \in \uparrow_i^{\text{N}}(\text{true})$ for any (M, t) . Thus $\uparrow_i^{\text{N}}(\text{true}) = \text{true}$.
- $\uparrow_i^{\text{N}}(c \cap d) = \uparrow_i^{\text{N}}(c) \cap \uparrow_i^{\text{N}}(d)$. We first prove monotonicity (i.e. if $c \subseteq d$ then $\uparrow_i^{\text{N}}(c) \subseteq \uparrow_i^{\text{N}}(d)$). For this, suppose $c \subseteq d$ and $(M, t) \in \uparrow_i^{\text{N}}(c)$. If there exists a state s with $(M, s) \in c$ s.t. $s \xrightarrow{i}_M t$ then $(M, s) \in d$, thus $(M, t) \in \uparrow_i^{\text{N}}(d)$. If such state does not exist, then $(M, t) \in \uparrow_i^{\text{N}}(d)$. We now proceed with the proof.
 - $\uparrow_i^{\text{N}}(c \cap d) \subseteq \uparrow_i^{\text{N}}(c) \cap \uparrow_i^{\text{N}}(d)$. We know $c \cap d \subseteq c$ and $c \cap d \subseteq d$. Then, by monotonicity of \uparrow_i^{N} , $\uparrow_i^{\text{N}}(c \cap d) \subseteq \uparrow_i^{\text{N}}(c)$ and $\uparrow_i^{\text{N}}(c \cap d) \subseteq \uparrow_i^{\text{N}}(d)$. Therefore

$$\uparrow_i^N(c \cap d) \subseteq \uparrow_i^N(c) \cap \uparrow_i^N(d).$$

- $\uparrow_i^N(c) \cap \uparrow_i^N(d) \subseteq \uparrow_i^N(c \cap d)$. Suppose $(M, t) \in \uparrow_i^N(c) \cap \uparrow_i^N(d)$. 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 $(M, t) \in \uparrow_i^N(c \cap d)$. If this is not the case, then t is not determined w.r.t. \xrightarrow{i}_M by any other state, hence $(M, t) \in \uparrow_i^N(c \cap d)$ by definition of \uparrow_i^N .

□

5.2. Deriving Minimal Normal-Right Inverses

In Definition 5.6 we included in $\uparrow_i^N(c)$ 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 $\uparrow_i^N(c)$ and obtain a *minimal* normal right inverse.

We shall see that not adding *all* uniquely determined 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 selection must obey certain conditions to guarantee that the resulting right inverse self-map is still normal.

Remark 5.8. *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 $\triangleright_i(M, s)$ denotes the set of all (M, t) such that s uniquely determines t w.r.t. \xrightarrow{i}_M (see Notation 4.9). Notice that $\triangleright_i(M, s) \neq \emptyset$ since we are assuming that each (M, s) must be determining w.r.t. \xrightarrow{i}_M .*

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

Definition 5.9 (Minimal Normal-Right Inverse). *Let $\mathbf{K}(S) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$*

be a Kripke SCS over $\mathcal{S} \subseteq \mathcal{M}^D$. Define $\uparrow_{i,\mathfrak{s}}^{\text{mN}} : \text{Con} \rightarrow \text{Con}$ as

$$\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c) \stackrel{\text{def}}{=} \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in c\} \cup \quad (7)$$

$$\bigcup_{(M,s) \notin c} \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s))\} \quad (8)$$

where \mathfrak{s} is a family $\{\mathfrak{s}_{\bar{c}}\}_{c \in \text{Con}}$ such that (1) $\mathfrak{s}_{\bar{c}}$ is a selector for $\{\triangleright_i(M, s)\}_{(M,s) \notin c}$ and (2) if $c \subseteq d$ then $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s)) = \mathfrak{s}_{\bar{d}}(\triangleright_i(M, s))$ for every $(M, s) \notin d$. \square

From the above definition of $\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ it is easy to see that :

- (7) includes the indetermined states, the multiply determined states and the uniquely determined states by any state s such that $(M, s) \in c$, and
- (8) includes uniquely determined states by any s' such that $(M, s') \notin c$ but it excludes uniquely determined states selected by the choice function $\mathfrak{s}_{\bar{c}}$ in Definition 5.9 (1).

In the following example we will illustrate that if we were to include these selected states, $\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ would not be a right inverse of $[\cdot]_i$.

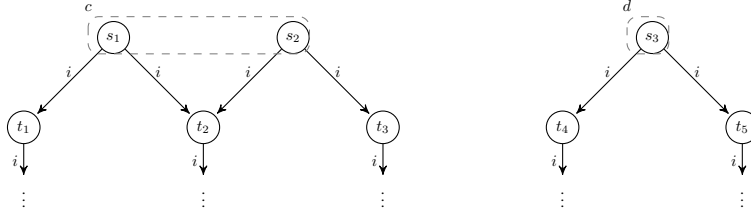


Figure 2: Accessibility relations corresponding to the KS M for an agent i .

Example 5.10. Consider Figure 2 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 $\mathfrak{s}_{\bar{d}}(\triangleright_i(M, s_1)) = (M, t_1)$, $\mathfrak{s}_{\bar{d}}(\triangleright_i(M, s_2)) = (M, t_3)$ and $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s_3)) = (M, t_5)$. From Definition 5.9 we obtain

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

Let us now consider $h(c) = \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c) \cup \mathfrak{s}_{\bar{c}}(\triangleright_i(M, s_3))$ to be $\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ plus the pointed KS from $\triangleright_i(M, s_3)$ 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 for $[\cdot]_i$.
From Definition 4.3 we would have

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

□

Notice that the selection of elements that are not included in $\uparrow_{i, \mathfrak{s}}^{\text{mN}}(c)$ need to obey Condition (2) in Definition 5.9. This condition is needed for the normality of $\uparrow_{i, \mathfrak{s}}^{\text{mN}}(c)$. We illustrate this in the following example.

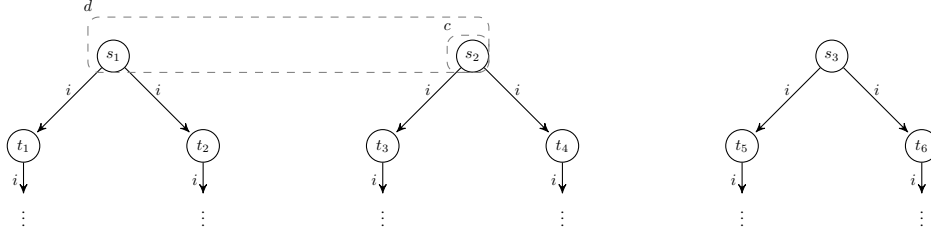


Figure 3: Accessibility relations corresponding to the KS M for an agent i .

Example 5.11. Consider Figure 3 with $c = \{(M, s_2)\}$ 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 $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s_1)) = (M, t_1) = \mathfrak{s}_{\overline{c \cap d}}(\triangleright_i(M, s_1))$, $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s_3)) = (M, t_5) = \mathfrak{s}_{\overline{c \cap d}}(\triangleright_i(M, s_3))$, and $\mathfrak{s}_{\bar{d}}(\triangleright_i(M, s_3)) = (M, t_6)$.

We have $c \subseteq d$ but $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s_3)) \neq \mathfrak{s}_{\bar{d}}(\triangleright_i(M, s_3))$ hence \mathfrak{s} does not satisfy Condition (2) in Definition 5.9.

Notice that if we were to drop Condition (2) in Definition 5.9 we would obtain the following.

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

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

$$\uparrow_{i, \mathfrak{s}}^{\text{mN}}(c) \cap \uparrow_{i, \mathfrak{s}}^{\text{mN}}(d) \neq \uparrow_{i, \mathfrak{s}}^{\text{mN}}(c \cap d).$$

Then $\uparrow_{i,s}^{\text{mN}}$ would not be a normal right inverse of $[\cdot]_i$ (Theorem 3.6) \square

Now we prove that the function $\uparrow_{i,s}^{\text{mN}}(\cdot)$ presented in Definition 5.9 is a normal right inverse of the space function $[\cdot]_i$.

Theorem 5.12. *Let $\mathbf{K}(\mathcal{S}) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ and $\uparrow_{i,s}^{\text{mN}}$ as in Definition 5.9. The self-map $\uparrow_{i,s}^{\text{mN}}$ is a normal right inverse of $[\cdot]_i$.*

Proof. To prove that $\uparrow_{i,s}^{\text{mN}}$ is a right inverse we prove $[\uparrow_{i,s}^{\text{mN}}(c)]_i = c$. To prove that it is a normal map, by Theorem 3.6, it suffices to prove $\uparrow_{i,s}^{\text{mN}}(\text{true}) = \text{true}$ and $\uparrow_{i,s}^{\text{mN}}(c \sqcup d) = \uparrow_{i,s}^{\text{mN}}(c) \sqcup \uparrow_{i,s}^{\text{mN}}(d)$ (equivalent to proving $\uparrow_{i,s}^{\text{mN}}(c \cap d) = \uparrow_{i,s}^{\text{mN}}(c) \cap \uparrow_{i,s}^{\text{mN}}(d)$).

- $[\uparrow_{i,s}^{\text{mN}}(c)]_i = c$. The proof of this equality is similar to corresponding one in the proof of Theorem 5.7. Suppose $(M, s) \in [\uparrow_{i,s}^{\text{mN}}(c)]_i$. Then $\blacktriangleright_i(M, s) \subseteq \uparrow_{i,s}^{\text{mN}}(c)$ from Definition 4.3. Since every state from $\mathbf{K}(\mathcal{S})$ is determinant w.r.t. \xrightarrow{i}_M , there must be t s.t. $s \xrightarrow{i}_M t$ and $(M, t) \in \uparrow_{i,s}^{\text{mN}}(c)$. Therefore by definition of $\uparrow_{i,s}^{\text{mN}}$, $(M, s) \in c$. 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 $\blacktriangleright_i(M, s) \subseteq \uparrow_{i,s}^{\text{mN}}(c)$, and consequently $(M, s) \in [\uparrow_{i,s}^{\text{mN}}(c)]_i$ by Definition 4.3.
- $\uparrow_{i,s}^{\text{mN}}(\text{true}) = \text{true}$. The proof of this equality is also similar to corresponding one in the proof of Theorem 5.7. Recall that true is the set of all pointed KS's (M, s) (with $M \in \mathcal{S}$) while false is the empty set. Thus $\emptyset = \bigcup_{(M,s) \notin \text{true}} \blacktriangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{c}}(\blacktriangleright_i(M, s))\}$. From Definition 5.9 we obtain $\uparrow_{i,s}^{\text{mN}}(\text{true}) = \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in \text{true}\} = \text{true}$.
- $\uparrow_{i,s}^{\text{mN}}(c \cap d) = \uparrow_{i,s}^{\text{mN}}(c) \cap \uparrow_{i,s}^{\text{mN}}(d)$. We first prove monotonicity of $\uparrow_{i,s}^{\text{mN}}$: If $c \subseteq d$ then $\uparrow_{i,s}^{\text{mN}}(c) \subseteq \uparrow_{i,s}^{\text{mN}}(d)$. Suppose $c \subseteq d$ and $(M, t) \in \uparrow_{i,s}^{\text{mN}}(c)$. We want to prove that $(M, t) \in \uparrow_{i,s}^{\text{mN}}(d)$. If t is indetermined or multiply determined w.r.t. \xrightarrow{i}_M then trivially, from Definition 5.9, (M, t) is included $\uparrow_{i,s}^{\text{mN}}(d)$. Otherwise, t is uniquely determined and thus we must have an s such that $s \xrightarrow{i}_M t$. If $(M, s) \in c$ then since $c \subseteq d$, we have $(M, s) \in d$ and thus $(M, t) \in \uparrow_{i,s}^{\text{mN}}(d)$ by Definition 5.9. Otherwise $(M, s) \notin c$ and since $(M, t) \in \uparrow_{i,s}^{\text{mN}}(c)$ we conclude from Definition 5.9 that

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

If $(M, s) \in d$, then $(M, t) \in \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in d\}$ and $(M, t) \in \uparrow_{i,s}^{\text{mN}}(d)$. Otherwise $(M, s) \notin d$ and then from Condition (2) in Definition 5.9 we conclude that $(M, t) \in \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{d}}(\triangleright_i(M, s))\}$. Hence from Definition 5.9 we conclude $(M, t) \in \uparrow_{i,s}^{\text{mN}}(d)$ as wanted.

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

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

And

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

From Definition 5.9 we have $\uparrow_{i,s}^{\text{mN}}(c \cap d) = B_1 \cup B_2$. Using Definition 5.9 and distributive set laws we obtain $\uparrow_{i,s}^{\text{mN}}(c) \cap \uparrow_{i,s}^{\text{mN}}(d) = (A'_1 \cup A'_2) \cap (A''_1 \cup A''_3) = \bigcup_{i \in I} A_i$ 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 5.9 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 $\uparrow_{i,s}^{\text{mN}}(c) \cap \uparrow_{i,s}^{\text{mN}}(d) \subseteq \uparrow_{i,s}^{\text{mN}}(c \cap d)$ as wanted. □

We shall show that the family of normal right inverses in Definition 5.9 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 5.13. *Let $\mathbf{K}(S) = (Con, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ and $\uparrow_{i,s}^{mN}$ as in Definition 5.9. Then $\uparrow_{i,s}^{mN}$ 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 $g(c) \sqsubseteq \uparrow_{i,s}^{mN}(c)$ for every $c \in Con$. Recall that $g(c) \sqsubseteq \uparrow_{i,s}^{mN}(c)$ is equivalent to $\uparrow_{i,s}^{mN}(c) \subseteq g(c)$ (Remark 4.12).

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

Since $(M, t) \notin \uparrow_{i,s}^{mN}(c)$, we can use Lemma 5.3 and Definition 5.9 to show that

$$(M, s) \notin c \text{ and that } (M, t) = \mathfrak{s}_{\bar{c}}(\triangleright_i(M, s)) \quad (9)$$

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

From Equation 9 and Definition 5.9 it follows that $\triangleright_i(M, s) \setminus \{(M, t)\}$ is included in $\uparrow_{i,s}^{mN}(c)$. But $\uparrow_{i,s}^{mN}(c) \subseteq g(c)$ and $(M, t) \in g(c)$ thus

$$\triangleright_i(M, s) \subseteq g(c). \quad (10)$$

From Lemma 5.3(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. $\xrightarrow{i} M$. Consequently, using Equation 10 we conclude

$$\blacktriangleright_i(M, s) \subseteq g(c). \quad (11)$$

Since we assumed that g is a right inverse of $[\cdot]_i$ we have $[g(c)]_i = c$. We can use Equation 11 and Definition 4.3 to conclude $(M, s) \in [g(c)]_i$. Hence $(M, s) \in c$, a contradiction with Equation 9. \square

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 $\uparrow_{i,\mathfrak{s}}^{\text{mN}}$.

Theorem 5.14. *Let $\mathbf{K}(S) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ as in Definition 5.9. Suppose that g is a normal right inverse of the space function $[\cdot]_i$. Then there exists a minimal normal right inverse $\uparrow_{i,\mathfrak{s}}^{\text{mN}}$ as in Definition 5.9 such that $\uparrow_{i,\mathfrak{s}}^{\text{mN}} \sqsubseteq g$.*

Proof. Given a normal right inverse g of $[\cdot]_i$ we will show the existence of a minimal right inverse $\uparrow_{i,\mathfrak{s}}^{\text{mN}}$ of $[\cdot]_i$ for some \mathfrak{s} such that for every $c \in \text{Con}$ if $(M, t) \in g(c)$ then $(M, t) \in \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ (see Remark 4.12).

From Theorem 5.12, Lemma 5.3 and Definition 5.9, $\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ 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 $(M, t) \in \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$. More precisely, it suffices to prove that there exists a family of selectors \mathfrak{s} such that the following holds for any $c \in \text{Con}$:

$$\text{If } (M, t) \in g'(c) \text{ then } (M, t) \in \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c) \quad (12)$$

where $g'(c) \stackrel{\text{def}}{=} g(c) \cap \bigcup_{(M,s) \notin c} \triangleright_i(M, s)$.

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

$$\text{If } (M, s) \notin c \text{ then } \triangleright_i(M, s) \setminus g'(c) \neq \emptyset. \quad (13)$$

This property follows from the fact that g is normal and a right inverse. We prove something stronger: If $(M, s) \notin c$ then $g'(c) \subset \triangleright_i(M, s)$. Notice that if $(M, s) \notin c$ then $g'(c) \subseteq \triangleright_i(M, s)$ since the sets $\triangleright_i(M, s')$ of states uniquely-determined by s' are mutually exclusive. Then by way of contradiction suppose $(M, s) \notin c$ and $g'(c) = \triangleright_i(M, s)$. Since $g'(c) \subseteq g(c)$ we would have $\triangleright_i(M, s) \subseteq g(c)$. From Lemma 5.3(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 $\triangleright_i(M, s) \subseteq g(c)$ we then conclude

$\triangleright_i(M, s) \subseteq g(c)$. But g is a right inverse of $[\cdot]_i$ thus $[g(c)]_i = c$. Using Definition 4.3 we would then conclude $(M, s) \in [g(c)]_i$. Hence $(M, s) \in c$, a contradiction with the assumption in Equation 13.

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

$$\text{If } c \subseteq d \text{ and } (M, s) \notin d \text{ then } \triangleright_i(M, s) \setminus g'(d) \subseteq \triangleright_i(M, s) \setminus g'(c) \quad (14)$$

To prove this property, suppose that $c \subseteq d$, $(M, s) \notin d$, and $(M, t) \in \triangleright_i(M, s) \setminus g'(d)$. Thus $(M, t) \in \triangleright_i(M, s)$ and $(M, t) \notin g'(d)$. Since $(M, s) \notin d$ we have $(M, t) \in \bigcup_{(M, s) \notin d} \triangleright_i(M, s)$. But $(M, t) \notin g'(d)$, and we must have $(M, t) \notin g(d)$. Since g is normal it is monotone (Corollary 3.7), consequently $c \subseteq d$ and $(M, t) \notin g(d)$ implies $(M, t) \notin g(c)$. Hence $(M, t) \notin g'(c)$. It follows that $(M, t) \in \triangleright_i(M, s) \setminus g'(c)$ as wanted.

To conclude the proof we now identify a minimal normal right inverse $\uparrow_{i, \mathfrak{s}}^{\text{mN}}$ that satisfies Equation 12. Let \mathfrak{s} to be a family of choice functions $\{\mathfrak{s}_{\bar{c}}\}_{c \in \text{Con}}$ such that (I) $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s)) \in \triangleright_i(M, s) \setminus g'(c)$ and (II) $\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s)) = \mathfrak{s}_{\bar{d}}(\triangleright_i(M, s))$ for every $(M, s) \notin d$ whenever $c \subseteq d$. Notice that because of Equation 13 condition (I) can be fulfilled. Similarly, with the help of Equations 13 and 14, condition (II) can also be fulfilled. Clearly \mathfrak{s} satisfies the selection conditions in Definition 5.9. Thus we have

$$\uparrow_{i, \mathfrak{s}}^{\text{mN}}$$

$$\begin{aligned} \uparrow_{i, \mathfrak{s}}^{\text{mN}}(c) = & \{(M, t) \mid \forall s : s \xrightarrow{i}_M t, (M, s) \in c\} \cup \\ & \bigcup_{(M, s) \notin c} \triangleright_i(M, s) \setminus \{\mathfrak{s}_{\bar{c}}(\triangleright_i(M, s))\} \end{aligned}$$

Because of (I), this minimal right inverse $\uparrow_{i, \mathfrak{s}}^{\text{mN}}$ is easily seen to satisfy Equation 12 which concludes the proof.

□

5.3. Summary

In this section we 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 maximum normal right inverse \uparrow_i^{MN} , a normal right inverse \uparrow_i^{N} and the family of *all* minimal right inverses, denoted as $\{\uparrow_{i,\mathfrak{s}}^{\text{mN}}\}$. The following corollary summarizes our classification.

Corollary 5.15 (Taxonomy). *Fix a Kripke SCS $\mathbf{K}(S) = (\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$ with $S \subseteq \mathcal{M}^D$. Then*

1. $\uparrow_{i,\mathfrak{s}}^{\text{mN}} \sqsubseteq \uparrow_i^{\text{N}} \sqsubseteq \uparrow_i^{\text{MN}}$ for every \mathfrak{s} as in Definition 5.9, and
2. for every normal right inverse g of $[\cdot]_i$, there exists \mathfrak{s} as in Definition 5.9 such that

$$\uparrow_{i,\mathfrak{s}}^{\text{mN}} \sqsubseteq g \sqsubseteq \uparrow_i^{\text{MN}} \quad (\text{i.e., } \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c) \supseteq g(c) \supseteq \uparrow_i^{\text{MN}}(c) \text{ for every } c \in \text{Con.})$$

where \uparrow_i^{MN} , \uparrow_i^{N} and $\uparrow_{i,\mathfrak{s}}^{\text{mN}}$ are given as in Definitions 5.4, 5.6 and 5.9 w.r.t. $\mathbf{K}(S)$.

The upper and lower bounds in Corollary 5.15(2) are useful to prove whether some e is in a given normal-right inverse g . Thus if $e \in \uparrow_i^{\text{MN}}(c)$ then $e \in g(c)$ and if $e \notin \uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ for every \mathfrak{s} then $e \notin g(c)$. We shall use these properties in the next section.

The following table summarizes the right inverses presented in Corollary 5.15 .

<i>Maximum Right Inverse</i>		
\uparrow_i^{M}	The maximum right inverse for the space function $[\cdot]_i$ wrt $\sqsubseteq (\supseteq)$. $\uparrow_i^{\text{M}}(c)$ is the minimum set (wrt \sqsubseteq) of pointed Kripke structures such that $[\uparrow_i^{\text{M}}(c)]_i = c$. The function \uparrow_i^{M} is not normal (Definition 3.5) as shown in Proposition 5.2.	Definition 4.13.
<i>Normal Right Inverses</i>		
\uparrow_i^{MN}	The maximum normal right inverse of $[\cdot]_i$ wrt $\sqsubseteq (\supseteq)$. The function \uparrow_i^{MN} is normal and $\uparrow_i^{\text{MN}}(c)$ is the minimum set (wrt \sqsubseteq) of pointed Kripke structures such that $[\uparrow_i^{\text{MN}}(c)]_i = c$.	Definition 5.4.
\uparrow_i^{N}	A normal right inverse of $[\cdot]_i$. The function \uparrow_i^{N} is normal and the set $\uparrow_i^{\text{N}}(c)$ is given by $\uparrow_i^{\text{MN}}(c)$ plus those pointed Kripke structures (M, t) such that t is indetermined w.r.t. $\xrightarrow{i} M$.	Definition 5.6.
$\uparrow_{i,\mathfrak{s}}^{\text{mN}}$	A minimal normal right inverse of $[\cdot]_i$ wrt $\sqsubseteq (\supseteq)$. The function $\uparrow_{i,\mathfrak{s}}^{\text{mN}}$ is normal and $\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)$ is a maximal set (wrt \sqsubseteq) of pointed Kripke structures, determined by a choice function \mathfrak{s}_c , such that $[\uparrow_{i,\mathfrak{s}}^{\text{mN}}(c)]_i = c$.	Definition 5.9.

6. 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 we recall the notion of modal language.

Definition 6.1 (Modal Language). *Let Φ be a set of primitive propositions. The modal language $\mathcal{L}_n(\Phi)$ is given by the following grammar:*

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

where $p \in \Phi$ and $i \in \{1, \dots, n\}$. 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 $\neg\text{ff}$. \square

We say that a pointed KS (M, s) satisfies ϕ iff $(M, s) \models \phi$ where \models is defined inductively as follows:

$$\begin{aligned} (M, s) &\models p \text{ iff } \pi_M(s)(p) = 1 \\ (M, s) &\models \phi \wedge \psi \text{ iff } (M, s) \models \phi \text{ and } (M, s) \models \psi \\ (M, s) &\models \neg\phi \text{ iff } (M, s) \not\models \phi \\ (M, s) &\models \Box_i \phi \text{ iff } (M, t) \models \phi \text{ for every } t \text{ such that } s \xrightarrow{i} t. \end{aligned}$$

Bisimilarity is a central equivalence in concurrency theory [24].

Definition 6.2 (Bisimilarity ([24])). *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 $s \xrightarrow{i} s'$ then there exists t' s.t. $t \xrightarrow{i} t'$ and $((M, s'), (N, t')) \in \mathcal{B}$. We say that (M, s) and (N, t) are (strongly) bisimilar, written $(M, s) \sim_s (N, t)$ if there exists a bisimulation \mathcal{B} such that $((M, s), (N, t)) \in \mathcal{B}$. \square*

Bisimilarity-invariance. The well-known result of *bisimilarity-invariance* of modal satisfiability [13] implies that if $(M, s) \sim_s (N, t)$ 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 6.3 (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 $\mathbf{C}[\![\phi]\!]_i$ inductively defined as follows:*

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

□

Remark 6.4. *One can verify that for any Kripke SCS $\mathbf{K}(\mathcal{S}_n(\Phi))$, the Heyting negation $\sim c$ (Def. 3.1) 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 $\mathbf{C}[\![\phi]\!]_i$ 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 $\mathbf{C}[\![\phi]\!] = \text{true}$.

Notice that from Proposition 4.5 and Theorem 3.6, each space function $[\cdot]_i$ of $\mathbf{K}(\mathcal{S}_n(\Phi))$ is a normal self-map. From Definitions 3.5 and 6.3 we can derive the following standard property stating that \Box_i is a normal modal operator:

1. (*Necessitation*) If ϕ is valid in $\mathcal{S}_n(\Phi)$ then $\Box_i\phi$ is valid in $\mathcal{S}_n(\Phi)$, and
2. (*Distribution*) $\Box_i(\phi \Rightarrow \psi) \Rightarrow (\Box_i\phi \Rightarrow \Box_i\psi)$ is valid in $\mathcal{S}_n(\Phi)$.

6.1. 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 [12], in modal logics for concurrency they represent backward moves [10], in epistemic logic they correspond to utterances [9].

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 6.5 (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:*

$$\phi, \psi, \dots := p \mid \phi \wedge \psi \mid \neg\phi \mid \Box_i\phi \mid \Box_i^{-1}\phi$$

where $p \in \Phi$ and $i \in \{1, \dots, n\}$. □

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:

$$\Box_i\Box_i^{-1}\phi \Leftrightarrow \phi \quad \text{valid in } \mathcal{S}_n(\Phi). \quad (15)$$

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 6.3. Assuming that such a right inverse exists, we can then interpret the reverse modality in \mathbf{C} as

$$\mathbf{C}[\Box_i^{-1}\phi] = \uparrow_i(\mathbf{C}[\phi]). \quad (16)$$

Since each \uparrow_i is a right inverse of $[\cdot]_i$, it is easy to verify that the interpretation satisfies the requirement in Equation 15. Furthermore, from Theorem 4.11 we can conclude that for each $M \in \mathcal{S}_n(\Phi)$, \xrightarrow{i}_M must necessarily be determinant-complete.

6.2. Normal Inverses Modalities

We can choose \uparrow_i in Equation 16 from the set $\{\uparrow_i^N, \uparrow_i^{MN}, \uparrow_i^M, \uparrow_{i,s}^{mN}\}$ of right inverse constructions in Section 4.3. Assuming that \uparrow_i is a normal self-map (e.g., either \uparrow_i^N , \uparrow_i^{MN} , or $\uparrow_{i,s}^{mN}$), we can show from Definition 3.5 and Equation 16 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 $\mathcal{S}_n(\Phi)$.

6.3. Inconsistency Invariance

We can conclude from Proposition 2.10(1) that since we assumed a right inverse of $[\cdot]_i$, we should have

$$\neg\Box_i ff \text{ valid in } \mathcal{S}_n(\Phi). \quad (17)$$

Indeed using the fact that $[\cdot]_i$ is a normal self-map with an inverse \uparrow_i and Theorem 3.6, we can verify the following:

$$\mathbf{C}[\Box_i ff] = \mathbf{C}[\Box_i(ff \wedge \Box_i^{-1} ff)] = \mathbf{C}[\Box_i(ff) \wedge \Box_i(\Box_i^{-1} ff)] = \mathbf{C}[\Box_i(ff) \wedge ff] = \mathbf{C}[ff].$$

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 [13] where $\neg\Box_i ff$ is not an axiom cannot be extended with a reverse modality satisfying Equation 15 (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. 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 .

6.4. 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 not 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 next as an application of our taxonomy of normal right inverses in Section 5.

Let us suppose that the chosen right inverse \uparrow_i in Equation 16 is *any* normal self-map whatsoever. Now take v and s_4 as in Figure 4. Suppose that $\pi_{M_5}(v) = \pi_{M_1}(s_i)$ for every s_i in the states of M_1 . Clearly $(M_1, s_4) \sim_s (M_5, v)$. Since s_4 is multiply determined then from Definition 5.4 $(m, s_4) \in \uparrow_i^{\text{MN}}(\text{false})$. Using Corollary 5.15(2) we obtain $(m, s_4) \in \uparrow_i(\text{false})$, and thus $(M_1, s_4) \models \Box_i^{-1}ff$.

Since v is uniquely determined, applying Definition 5.9 we conclude that $(M_5, v) \notin \uparrow_{i, \mathfrak{s}}^{\text{MN}}(\text{false})$ for any \mathfrak{s} . From Corollary 5.15(2) it follows that $(M_5, v) \notin \uparrow_i(\text{false})$ and thus $(M_5, v) \not\models \Box_i^{-1}ff$.

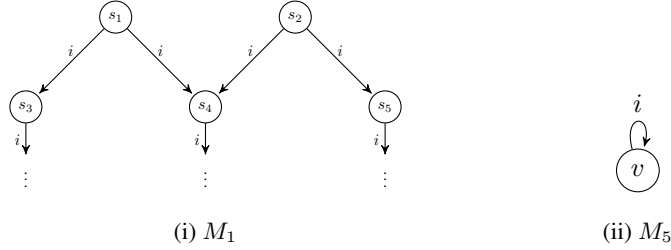


Figure 4: Accessibility relations for an agent i . In each sub-figure we omit the corresponding KS M_k from the edges and draw $s \xrightarrow{i} t$ whenever $s \xrightarrow{i}_{M_k} t$.

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

6.5. 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 (\circ) and the *henceforth/always* operator (\square), 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 .

Let us first consider the modality \Box_1 to be the next operator $\Box_1 = \circ$. 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 4.11 there exists a right inverse of $[\cdot]_i$. If we apply Equation 16 taking as the inverse $\uparrow_1 = \uparrow_1^M$, i.e., the maximum right inverse of $[\cdot]_1$, we obtain $\Box_1^{-1} = \ominus$, which corresponds to a past modality known in the literature as the *strong* previous operator [12]. Formally, the operator \ominus is given by $(M, t) \models \ominus \phi$ 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_i^N , we obtain $\Box_1^{-1} = \tilde{\ominus}$ the past modality known as *weak* previous operator [12]. The formal definition of the operator $\tilde{\ominus}$ is given by $(M, t) \models \tilde{\ominus} \phi$ 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 $(M, s) \not\models \ominus \phi$ and $(M, s) \models \tilde{\ominus} \phi$ for any ϕ . It follows that \ominus is not a normal operator, since $\ominus tt$ is not valid in $\mathcal{S}_2(\Phi)$ but tt is.

Now consider the modality \Box_2 to be the always operator $\Box_2 = \square$. 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 $s_0 \xrightarrow{1}_M s_1 \xrightarrow{1}_M \dots$. 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 4.11 tell us that there is no right inverse \uparrow_2 of $[\cdot]_i$ that can give us an operator \square_2^{-1} satisfying Equation 15 .

By analogy to the above-mentioned past operators, one may think that the past operator *it-has-always-been* \boxminus [8] may provide a reverse modality for \square in the sense of Equation 15. The operator is given by $(M, t) \models \boxminus\phi$ iff $(M, s) \models \phi$ for every s such that $s \xrightarrow{2}_M t$. Clearly $\square \boxminus \phi \Rightarrow \phi$ is valid in $\mathcal{S}_2(\Phi)$ but $\phi \Rightarrow \square \boxminus \phi$ is not.

7. Concluding Remarks and Related Work

We gave an algebraic counterpart of the notion of normality from modal logic: A self-map is normal if and only if it preserves finite suprema. We then studied the existence and derivation of right inverses (extrusion) 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 maximum 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.

Most of the related work was discussed in the previous sections. The constraint systems in this paper can be seen as a modal extension of geometric logic [15]. Modal logic has also been studied from an algebraic perspective by using modal extensions of boolean and Heyting algebras in [25, 20, 26]. These works, however, do not address issues related to inverse modalities. Inverse modalities have been used in temporal, epistemic and logic for concurrency. In [8] the authors discuss inverse temporal and epistemic modalities from a proof theory perspective. The work in [10, 27, 28] uses modal logic with reverse modalities for specifying true concurrency and [29, 30] 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.

Residuation theory [31] can be used for elegant characterizations of minimal information and inverse operators. In fact it has been used in the context of soft concurrent constraint programming, e.g. to allow for non-idempotent tell/post operations [32]. Soft CCP requires a richer notion of constraint system based on residuation theory. As future work, it is natural to consider spatial extensions of the constraint systems from [32] towards a more general framework of spatial soft CCP.

In the present paper we used spatial constraint systems (SCS's) as an abstract representation of modal logics. This helped us to prove new expressiveness results for these logics. As future work we plan to use SCS's as an abstract representation of process calculi. We expect to use this representation to prove expressiveness results similar to those in [33, 34] which are inspired by the close connection between processes and formulae.

Index

- \sqsubseteq , entailment, order relation, 9
- \sqcup, \sqcup , join, LUB, supremum, 9
- \sqcap, \sqcap , meet, GLB, infimum, 9
- false* (top), all (possibly inconsistent) information, 9
- true* (bottom), empty information, 9
- directed set, 9
- compact element, 9
- $\mathcal{C}, (\text{Con}, \sqsubseteq)$, constraint system, CS, 10
- Con*, set of constraints, 10
- Φ , set of primitive propositions, 11
- $\mathbf{B}(\Phi)$, boolean constraint system, 11
- $\mathcal{A}(\cdot)$, set of assignments, 11
- $\mathcal{L}_0(\cdot)$, propositional language, 11
- $\mathbf{B}[\cdot]$, boolean interpretation, 11
- $(\text{Con}, \sqsubseteq, [\cdot]_1, \dots, [\cdot]_n)$, spatial constraint system, scs, 12
- $[\cdot]$, space function, 12
- S.1, first space axiom, 12
- S.2, second space axiom, 12
- E.1, extrusion axiom, 14
- $[\cdot]^{-1}$, pre-image under $[\cdot]$, 14
- \rightarrow , Heyting implication, 16
- \sim , Heyting negation, 16
- constraint frame, 16
- $m(\cdot)$, normal map, 18
- (M, s) , pointed Kripke structure, 21
- π , states interpretation, 21
- $\blacktriangleright_i(M, s)$, M states accessible by i from s , 21
- S_M and π_M , set of states and interpretation function of M , 21
- $\mathbf{K}(\mathcal{S}_n(\cdot))$, Kripke SCS, 22
- $\mathcal{S}_n(\cdot)$, set of Kripke structures, 22
- $[c]_i(\cdot)$, space function for Kripke structures, 22
- Δ , set of pointed Kripke structures, 22
- \mathcal{R} , accessibility relation of a Kripke structure, 23
- $\triangleright_i(M, s)$, M states uniquely determined by i from s , 23
- $\blacktriangleright_i(M, S)$, M states determined by i from states in S , 23
- \mathcal{R} , accessibility relation of a Kripke structure, 23
- \mathcal{M}^D , class of Kripke structures whose \mathcal{R} are determinant-complete, 26
- \uparrow_i^M , maximum right inverse for space function $[\cdot]_i$ of agent i , 26
- $\text{nd}(\cdot)$, set of indetermined states, 27
- $\text{md}(\cdot)$, set of multiply determined states, 27

\uparrow_i^{MN} , maximum normal-right inverse for space function $[\cdot]_i$ of agent i , 30	ff , constant false, 42
\uparrow_i^{N} , a normal-right inverse for space function $[\cdot]_i$ of agent i , 32	\models , satisfiability relation, 42
$\uparrow_{i,s}^{\text{mN}}$, minimal normal-right inverse for space function $[\cdot]_i$ of agent i , 33	$\wedge, \vee, \Rightarrow, \Leftrightarrow$, logical operators, 42
\mathfrak{s} , family of choice functions/selectors, 33	\Box , box modality/modality, 42
$\{\uparrow_{i,s}^{\text{mN}}\}$, family of minimal right inverses for the space function $[\cdot]_i$ of agent i , 40	\sim_s , bisimilarity relation, 42
$\mathcal{L}_n(\cdot)$, modal language, 42	\mathcal{B} , bisimulation relation, 42
tt , constant true, 42	$\mathbf{C}[\phi]$, Kripke SCS interpretation of ϕ , 43
	\Box^{-1} , reverse modality, 44
	\bigcirc , next operator in linear-time temporal logic, 47
	\Box , always/henceforth operator in linear-time temporal logic, 47
	\ominus , strong previous operator, 47
	\boxminus , it-has-always-been operator, 48

References

- [1] S. Knight, C. Palamidessi, P. Panangaden, F. D. Valencia, Spatial and epistemic modalities in constraint-based process calculi, in: CONCUR 2012, Vol. 7454 of LNCS, Springer, 2012, pp. 317–332.
- [2] V. A. Saraswat, M. Rinard, P. Panangaden, Semantic foundations of concurrent constraint programming, in: POPL’91, ACM, 1991, pp. 333–352.
- [3] F. S. Boer, A. Di Pierro, C. Palamidessi, Nondeterminism and infinite computations in constraint programming, Theoretical Computer Science (1995) 37–78.
- [4] F. Fages, P. Ruet, S. Soliman, Linear concurrent constraint programming: Operational and phase semantics, Information and Computation (2001) 14–41.
- [5] J. Réty, Distributed concurrent constraint programming, Fundamenta Informaticae (1998) 323–346.

- [6] M. Nielsen, C. Palamidessi, F. D. Valencia, Temporal concurrent constraint programming: Denotation, logic and applications, *Nord. J. Comput.* 9 (1) (2002) 145–188.
- [7] S. Popkorn, *First steps in modal logic*, 1st Edition, Cambridge University Press, 1994.
- [8] M. Ryan, P.-Y. Schobbens, Counterfactuals and updates as inverse modalities, *Journal of Logic, Language and Information* (1997) 123–146.
- [9] S. Haar, S. Perchy, C. Rueda, F. D. Valencia, An algebraic view of space/belief and extrusion/utterance for concurrency/epistemic logic, in: *PPDP 2015*, ACM, 2015, pp. 161–172.
- [10] I. Phillips, I. Ulidowski, A logic with reverse modalities for history-preserving bisimulations, in: *EXPRESS’11*, Vol. 64 of *EPTCS*, 2011, pp. 104–118.
- [11] R. Fagin, J. Y. Halpern, Y. Moses, M. Y. Vardi, *Reasoning about knowledge*, 4th Edition, MIT press Cambridge, 1995.
- [12] A. Pnueli, Z. Manna, *The temporal logic of reactive and concurrent systems*, Springer, 1992.
- [13] M. Hennessy, R. Milner, Algebraic laws for nondeterminism and concurrency, *Journal of the ACM (JACM)* 32 (1) (1985) 137–161.
- [14] J. Hintikka, *Knowledge and belief*, Cornell Univeristy Press, 1962.
- [15] S. Vickers, *Topology via logic*, 1st Edition, Cambridge University Press, 1996.
- [16] M. Guzman, S. Perchy, C. Rueda, F. Valencia, Deriving Inverse Operators for Modal Logic, in: *Theoretical Aspects of Computing – ICTAC 2016*, Vol. 9965 of *Theoretical Aspects of Computing – ICTAC 2016*, Springer, 2016, pp. 214–232.
- [17] S. Haar, S. Perchy, C. Rueda, F. Valencia, An Algebraic View of Space/Belief and Extrusion/Utterance for Concurrency/Epistemic Logic, in: *17th International Symposium on Principles and Practice of Declarative Programming (PPDP 2015)*, ACM SIGPLAN, 2015, pp. 161–172.

- [18] M. Guzman, S. Haar, S. Perchy, C. Rueda, F. 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) (2017) 107–133.
- [19] S. Abramsky, A. Jung, Domain theory, in: S. Abramsky, et al. (Eds.), *Handbook of Logic in Computer Science*, Vol. 3, Oxford University Press, 1994, pp. 1–168.
- [20] P. Blackburn, M. De Rijke, Y. Venema, *Modal Logic*, 1st Edition, Cambridge University Press, 2002.
- [21] A. Aristizábal, F. Bonchi, C. Palamidessi, L. Pino, 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*, Springer, 2011, pp. 138–152.
- [22] A. N. Prior, *Past, present and future*, Vol. 154, Oxford University Press, 1967.
- [23] S. Knight, *The epistemic view of concurrency theory*, Ph.D. thesis, École Polytechnique, Palaiseau, France (2013).
- [24] R. Milner, *Communication and concurrency*, Vol. 84, Prentice hall New York etc., 1989.
- [25] D. Macnab, Modal operators on heyting algebras, *Algebra Universalis* 12 (1) (1981) 5–29.
- [26] A. Chagrov, M. Zakharyashev, *Modal logic*, volume 35 of oxford logic guides (1997).
- [27] R. De Nicola, G. L. Ferrari, Observational logics and concurrency models, in: *FSTTCS'90*, Vol. 472 of LNCS, Springer, 1990, pp. 301–315.
- [28] U. Goltz, R. Kuiper, W. Penczek, Propositional temporal logics and equivalences, in: *CONCUR'92*, Vol. 630 of LNCS, Springer, 1992, pp. 222–236.
- [29] R. De Nicola, U. Montanari, F. Vaandrager, Back and forth bisimulations, in: *CONCUR '90*, Vol. 458 of LNCS, Springer, 1990, pp. 152–165.

- [30] R. De Nicola, F. Vaandrager, Three logics for branching bisimulation, *Journal of the ACM (JACM)* 42 (2) (1995) 458–487.
- [31] T. Blyth, M. Janowitz, *Residuation Theory*, International Series of Monographs in Pure and Applied Mathematics, Vol. 102, Pergamon Press, Oxford, 1972.
- [32] F. Gadducci, F. Santini, L. F. Pino, F. D. Valencia, A labelled semantics for soft concurrent constraint programming, in: *International Conference on Coordination Languages and Models*, Springer, 2015, pp. 133–149.
- [33] C. Palamidessi, V. Saraswat, F. D. Valencia, B. Victor, On the Expressiveness of Linearity vs Persistence in the Asynchronous Pi-Calculus, in: *Twenty First Annual IEEE Symposium on Logic in Computer Science (LICS)*, IEEE Computer Society, Seattle, United States, 2006, pp. 59–68.
- [34] D. Cacciagrano, F. Corradini, J. Aranda, F. Valencia, Linearity, Persistence and Testing Semantics in the Asynchronous Pi-Calculus, in: R. Amadio, T. Hildebrandt (Eds.), *14th International Workshop on Expressiveness in Concurrency (EXPRESS'07)*, Lisbon, Portugal, 2007, pp. 58–71.