Doctopic: Logic, semantics and theory of programming ARTICLE IN PRESS TCS:11604

Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Characterizing right inverses for spatial constraint systems with applications to modal logic ☆

Michell Guzmán [a], Salim Perchy [b], Camilo Rueda [c], Frank D. Valencia [d,c,*]

[a] *INRIA-LIX, École Polytechnique de Paris, France*
[b] *Loria, INRIA Grand-Est, France*
[c] *Pontificia Universidad Javeriana de Cali, Colombia*
[d] *CNRS & LIX, École Polytechnique de Paris, France*

A R T I C L E I N F O

Article history:
Received 22 April 2017
Received in revised form 14 May 2018
Accepted 17 May 2018
Available online xxxx

Keywords:
Constraint systems
Concurrent constraint programming
Concurrency theory
Modal logic
Inverse operators

A B S T R A C T

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 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.

© 2018 Published by Elsevier B.V.

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) [2,3,1,4–6]. 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

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

* Corresponding author.
E-mail address: frank.valencia@gmail.com (F.D. Valencia).

https://doi.org/10.1016/j.tcs.2018.05.022
0304-3975/© 2018 Published by Elsevier B.V.

Please cite this article in press as: M. Guzmán et al., Characterizing right inverses for spatial constraint systems with applications to modal logic, Theoret. Comput. Sci. (2018), https://doi.org/10.1016/j.tcs.2018.05.022

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 $[\cdot]_i : Con \to Con$ on the elements of the constraint system Con that preserve the join operation. These functions are called *space functions*. A CS equipped with space functions is called a *spatial constraint system* (SCS). From a computational point of view, given $c \in Con$, the assertion (constraint) $[c]_i$ specifies that c resides within the space of agent i. From an epistemic point of view, the assertion $[c]_i$ specifies that agent i considers c to be true (i.e. that in the world of agent i assertion c is true). Both intuitions convey the idea of c being local (subjective) to agent i.

The 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 $\uparrow_i : Con \to Con$ such that $[\ \uparrow_i c\ ]_i = c$. The computational interpretation of $\uparrow_i$ is that of a process being able to extrude any c from the space $[\cdot]_i$. 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., $c \sqcup d = \mathit{false}$), 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 $\Box_1, \Box_2, \ldots, \Box_n$, 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 $\xrightarrow{1}_M, \xrightarrow{2}_M, \ldots, \xrightarrow{n}_M$ 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-labeled transition; i.e. $s \xrightarrow{i}_M t$. We shall use $[\![\phi]\!]$ 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 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 $[\![\phi]\!]$ is a constraint in a Kripke SCS. The set $[\![\Box_i\phi]\!]$ can be compositionally obtained, using the space function $[\cdot]_i$ of agent i, as the constraint $[\ [\![\phi]\!]\ ]_i$ in the Kripke SCS. Furthermore, if there exists an extrusion

Download English Version:

https://daneshyari.com/en/article/10225769

Download Persian Version:

https://daneshyari.com/article/10225769

Daneshyari.com