Lattices for Space/Belief and Extrusion/Utterance

Stefan Haar
INRIA/LSV, CNRS & ENS

Camilo Rueda
Pontificia Universidad Javeriana Cali, Colombia

Salim Perchy
INRIA and LIX École Polytechnique de Paris

Frank Valencia
CNRS and LIX École Polytechnique de Paris

The notion of constraint system (cs) is central to *declarative programming* formalisms such as concurrent constraint programming (ccp) and constraint logic programming (clp). Constraint systems are often represented as lattices: their elements, called constraints, represent partial information and their order corresponds to entailment. Recently a notion of (n-agent) spatial cs was introduced to represent information in concurrent constraint programming for spatially distributed systems of multi-agent systems. From a computational point of view a spatial constraint system can be used to specify partial information holding in a given agent's space (*local information*). From an epistemic point of view a spatial cs can be used to specify information that a given agent considers true (*beliefs*). Spatial constraint systems, however, do not provide a mechanism for specifying the mobility/communication of information from one space to another.

In this paper we enrich spatial constraint systems with operators to specify information and processes moving from a space to another. We shall investigate the properties of this new family of constraint systems and illustrate their applications. From a computational point of view the new operators provide for process/information *extrusion,* a central concept in formalisms for *mobile communication*. From an epistemic point of view extrusion corresponds to a notion we shall call *utterance*; a piece of information that an agent communicate to others but that may be inconsistent with the agent's beliefs. Utterances can then be used to express hoaxes or intentional lies which are common place in social media.

1 Introduction

Motivation. Epistemic, mobile and spatial behaviour is commonplace in today's distributed systems. The intrinsic *epistemic* nature of these systems arises from social behavior. We have multiple agents (users) sharing *beliefs*, *opinions* and even intentional *lies* (hoaxes) on social networks. Being able to model this epistemic flow in a distributed environment would be an important contribution to the assessment of safety risks. As for the spatial and mobile behaviour, compelling examples are provided by apps and data moving across (possibly nested) spaces defined by friend circles, groups, and shared folders in social networks and cloud storage. A solid understanding of the notion of *space* and *spatial mobility* is relevant in any model of today's distributed systems.

Declarative formalisms of concurrency theory such as process calculi for *concurrent constraint programming* (ccp) [SRP91] were designed to give explicit access to the concept of partial information and, as such, had close ties with logic [PSSS93, MPSS95]. This makes them ideal for the incorporation of epistemic and spatial concepts by expanding the logical connections to include *multi-agent modal logic* [Kri63]. In fact, the sccp calculus [KPPV12] extends ccp with the ability to define local computational spaces where agents can store epistemic information and run processes.

To appear in EPTCS.

The Problem. Despite being able to express meaningful epistemic and spatial phenomena such as belief, knowledge, local and global information, the sccp calculus does not provide a mechanism to intentionally *extrude* information or processes from local spaces. Such a mechanism would allow sccp to express the transfer of epistemic information from one space into another. To our knowledge, *spatial mobility* can at best be expressed indirectly in sccp or any other ccp process calculus.

Constraint Systems. The notion of *constraint system (cs)* is central to ccp and other declarative formalisms such as (concurrent) constraint logic programming (clp). All ccp calculi are parametric in a cs that specifies partial information upon which programs (processes) may act. A cs is often represented as a *complete lattice* $(Con, \sqsubseteq)$. The elements of *Con*, the *constraints*, represent partial information and we shall think of them as being *assertions*. 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.

Constraint systems provide the domains and operation upon which the semantic foundations of ccp calculi are built. As such, 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, and existential quantification and local variables correspond to a cylindric operation on the set of constraints [SRP91].

Similarly, the notion of computational space and the epistemic notion of belief in sccp [KPPV12] corresponds to a family of self-maps $[\cdot]_i$, called *space functions*, on the elements of the constraint system. From a computational point of view $[c]_i$ means c holds within the space of agent i, and from an epistemic point of view it means that agent i considers c to be true. Both intuitions convey the idea of c being local (subjective) to agent i.

It is therefore natural to assume that a mechanism for extrusion in ccp ought to have a corresponding semantic concept in constraint systems. Furthermore, by incorporating extrusion directly in constraint systems, the concept may become available not only to sccp but also to other declarative constraint-based formalisms.

Goal. Our goal in this paper is then to investigate algebraic operations in the constraint system that provide the semantic foundations for extrusion. From a computational point of view, the new operations will allow us to specify mobile behaviour as constraints. From a logic point of view, they will allow us to specify epistemic concepts such as utterances, opinions, and intentional lies.

Contributions. In this paper we generalize the underlying theory of spatial constraint systems by adding *extrusion* functions to their structure. These functions provide for the specification of spatial mobility and epistemic concepts such as utterance and lies. Our main contributions can be summarised and structured as follows.

- *Extrusion as the right inverse of space.* We shall first introduce a family of self-maps $\uparrow_i$, called *extrusion functions*. Computationally, $\uparrow_i$ can be used to intentionally extrude information from within a space $[\cdot]_i$. Epistemically, $\uparrow_i$ can be used to express *utterances* by agent i. We shall put forward the notion of extrusion/utterance as the *right inverse* of space/belief. Under this interpretation we obtain
$$[c \sqcup \uparrow_i e]_i = [c]_i \sqcup e.$$

This equation illustrates the extrusion of e from the space of agent i and it is reminiscent of the characteristic property for intentional mobility in concurrent calculi [CG98]. By building upon concepts of Heyting Algebra, we will illustrate meaningful spatial and epistemic behaviours. In particular, *program mobility* and *intentional lie (hoax)*, i.e., the utterance of a statement by a given agent that is inconsistent with their beliefs.

- *The Extrusion Problem.* We consider the problem of deriving the corresponding extrusion functions $\uparrow_i$ given a cs with space functions $[\cdot]_i$. We will give canonical constructions of extrusion functions as well as impossibility results for their existence for surjective space functions that satisfy certain limit conditions such as Scott-continuity and meet-completeness.
- *Properties of Extrusion.* We will also investigate distinctive properties of space and extrusion functions. We will show that space functions that admit extrusion are necessarily space consistent: $[false]_i = false$. This corresponds to the Consistency Axiom of Epistemic (Doxastic) logic that states that no agent believes the false statement. We shall show that extrusion functions are *order embedding*, and that injective spaces are *order automorphisms* (hence they preserve all limits). We shall also identify necessary and sufficient conditions under which space and extrusion form a *Galois connection*: Namely a correspondance of the form $[c]_i \sqsubseteq d \Leftrightarrow c \sqsubseteq \uparrow_i d$.
- *Application: A logic of Belief and Utterance.* As an application of the above-mentioned contributions we show how to derive extrusion for a previously-defined instance of spatial constraint systems, namely, Kripke cs [KPPV12]. We also derive the semantics for a logic of belief with reverse modalities by interpreting its formulae as elements in the Kripe cs with extrusion. We can then show how express instances of epistemic notions such as utterances and lies directly in the syntax of this logic. We conclude by showing that belief and utterance in this logic also form a Galois connection. Roughly speaking, this connection allows us to reduce the implication *of* belief from/to implication *by* utterance.

Note: This paper has been submitted for publication at PPDP 2015. The current version has been submitted for presentation-only at ICE 2015. The full version can be dowloaded from:

`https://www.dropbox.com/s/02nj63q2f5an1dm/mccp-ice.pdf?dl=0`

References

[CG98] Luca Cardelli and Andrew D Gordon. Mobile ambients. In *Foundations of Software Science and Computation Structures*, pages 140–155. Springer, 1998.

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

[Kri63] Saul A Kripke. Semantical analysis of modal logic i normal modal propositional calculi. *Mathematical Logic Quarterly*, 9(5-6):67–96, 1963.

[MPSS95] Nax P Mendler, Prakash Panangaden, Philip J Scott, and RAG Seely. A logical view of concurrent constraint programming. *Nord. J. Comput.*, 2(2):181–220, 1995.

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

[SRP91] Vijay A Saraswat, Martin Rinard, and Prakash Panangaden. The semantic foundations of concurrent constraint programming. In *Proceedings of the 18th ACM SIGPLAN-SIGACT symposium on Principles of programming languages*, pages 333–352. ACM, 1991.