The Logic in Computer Science Column

by

Yuri Gurevich

Microsoft Research
One Microsoft Way, Redmond WA 98052, USA
gurevich@microsoft.com

Selected Papers from the 1^{st} Workshop "Logic, Language, and Information"

Guido Sciavicco
Dept. of Information and Communication Engineering
University of Murcia, Spain
guido@um.es
Alfredo Burrieza
Dept. of Phylosophy
university of Málaga, Spain
burrieza@uma.es

This issue's Logic Column in Computer Science features five selected papers from the 1^{st} Workshop "Logic, Language, and Information", held in Málaga (Spain) from the November 3^{rd} to November 5^{th}, 2014, and co-edited by Prof. Guido Sciavicco (University of Murcia) and Prof. Alfredo Burrieza (University of Málaga).

The workshop was held as the foundational act of the recently created Logic, Language and Information Research Unit (UILLI-AT), which features research groups from the University of Málaga and Sevilla, under the control of the institute Andalucia Tech. External researchers are also involved in the unit, and the basic areas of interests are logic, linguistics, computer science, and cognitive science, particularly focused on information management and representation. There were 16 contributions and 3 invited talks at the workshop.

Although the contributions to the workshop were focused on very different areas, the common denominator to all of them was logic. Among the areas that have been discussed during this event, we mention: formal analysis of concepts, information extraction models for metabolic networks, interval temporal logics, preference change logics in the context of social networks, automatic proof systems in databases, formal methods for analysis of spoken language, epistemic logics for collective awareness, bio-informatics analysis methods, transformation of programs, belief revision methods, abductive processes, and mereology in the context of temporal reasoning.

The selected papers, that underwent a full review process, are the following ones. *Distributed Explicit Knowledge and Collective Awareness*, by Alfredo Burrieza and Claudia Fernández: in this paper the authors present an approach to model the communication among a group of agents with limited knowledge resources; this leads to the introduction of the concept of collective awareness, which allows one to transform implicit distrbuited knowledge into explicit one. *A Logical Approach for Direct-Optimal Basis of Implications*, by Estrella Rodríguez-Lorenzo, Pablo Cordero, Manuel Enciso and Angel Mora: here the authors consider the problem of formal concept analysis, and in particular the problem of analyzing data by means of sets of implications; optimization is taken care of by means of Simplification Logic, presented by the authors, and shown to be equivalent to Armstrong axiomatization. *Abductive Reasoning in Dynamic Epistemic Logic - Generation and Selection of Hypothesis*, by Ismael Delgado-Arróniz: the paper aims to represent abductive reasoning in the context of epistemic logic; in particular, it is focused on highlighting the role of experience as a tool to select the best explaining hypothesis, and several methods are presented. *LCC-Program Transformers through Brzozowski's Equations*, by Enrique Sarrión-Morillo: an alternative to classic Kleene translation based on equational methods for program transformation is presented, aimed to dealing with the Logic of Change and Communication; this alternative method is studied in terms of complexity, formulas' length, and simplicity of implementation. And, finally, *Mereology and Temporal Structures*, by Pedro González Núñez: is this paper a semantic structure based on Kamp frames is presented as the theoretical basis to deal with mereological concepts such as *part of* in a spatio-temporal context; a first-order multi-modal language is described with a non-classical semantics to work with such structures.

Distributed explicit knowledge and collective awareness

Alfredo Burrieza Claudia Fernández-Fernández
Universidad de Málaga, Andalucía Tech
Departamento de Filosofía, España
burrieza@uma.es, clau5anj@gmail.com

Abstract

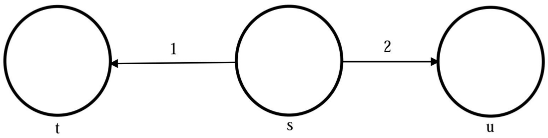
Our goal is to model communication in a group among agents with limited resources. Therefore we redefine the concept of distributed knowledge in order to distinguish between implicit and explicit knowledge. The most useful tool to do this is the concept of awareness, as a way of limiting and selecting what the agents really know. In this sense we propose a collective awareness that ensures full explicit communication and shows the dynamic aspects of the information exchange in a group. Depending on the definition of this collective awareness we are able to model the various ways in which the agents behave during communication.

1 Introduction

The standard definition of distributed knowledge (see [2]) intends to capture the knowledge flow between the agents in a group. The group is treated as another agent (the 'wise man') and acquires some knowledge that no individual agent on his own could posses. This new knowledge is derived from the information exchange between the agents in the group. Consider the standard epistemic language with the distributed knowledge operator D_G (where G is a set of agents). Let $\mathcal{M} = (S, R_1, \ldots, R_n, V)$ be any Kripke model (more details below) and $\models$ be the usual satisfaction relation, defined on the model. Then we have that for any $s \in S$:

$$\mathcal{M}, s \models D_G\varphi \quad \text{iff} \quad \mathcal{M}, t \models \varphi, \text{for all } t \text{ such that } (s,t) \in \textstyle\bigcap_{i \in G} R_i \qquad (D1)$$

This definition gives rise to strange cases in which we can model a group knowledge that has been established without a justified information exchange between the agents in the group, we could call these type of situations 'mysterious knowledge'. Imagine a simple case: Agent 1 knows that the movie X is not shown at at 17:00h, agent 2 knows it, too; but both (as a group), after communication, know that the movie is shown at 17:00h (p). How is this possible? Consider the following model $\mathcal{M}_{\text{movie}} = (S, R_1, R_2, V)$, where:



- $S = \{s, t, u\}$; $R_1 = \{(s, s), (s, t)\}$; $R_2 = \{(s, s), (s, u)\}$; $V(q) = \varnothing$ for all atom q of the language.

We have that $\mathcal{M}, s \models D_G p$, i.e., the group G (where $G = \{1, 2\}$) knows p at s, according to the definition above, in $(D1)$ (since $\bigcap_{i \in G} R_i = \varnothing$). But p is no logical consequence of the combined knowledge of the agents at s, since this knowledge is consistent and $\neg p$ is part of it. This is a case of 'mysterious knowledge' where we cannot justify coherently how the group acquires their knowledge. In our example we use a frame with no special properties on the accessibility relations. In [5] the authors present another case of mysterious knowledge where the accessibility relations are equivalence relations.

To avoid this kind of situations, we can use a different distributed knowledge definition that considers the logical consequences of the knowledge of the group members, as in [3]. But, although this situation could be fixed, the information flow happens in an ideal context. The communication that is being modelled focuses on implicit knowledge and belief. It does not take into account agents with limited reasoning, real agents. Our goal is to approach the distributed knowledge of a group whose members have limited reasoning resources.

When a group of agents establishes communication they exchange different kinds of information: knowledge, beliefs, doubts, mistakes, etc. If we focus only on the knowledge, then it should be clear that this knowledge is always explicit. If the agents have limited reasoning resources then we need an appropriate tool that distinguishes between what is implicit or explicit. Fagin et al., in [2], refer to the awareness of the agent as a way of limiting their knowledge (see also [1]).

The information exchange between the agents in a group modifies each individual awareness. Therefore the explicit knowledge of the group, seen as the knowledge of the wise man, depends on the awareness of the group. We could then consider a 'collective awareness' that will be applied to the explicit knowledge of the group in the same way that the individual awareness is applied to the explicit knowledge of the agent. In other words, if we have $K_i^e\varphi \equiv K_i\varphi \wedge A_i\varphi$, where $K_i^e\varphi$ means 'agent i explicitly knows φ', $K_i\varphi$ means 'agent i implicitly knows φ' and $A_i\varphi$ means 'agent i is aware of φ'; we can establish in a natural way that for any group of agents G we have: $D_G^e\varphi \equiv D_G\varphi \wedge A_G\varphi$, where $D_G^e\varphi$ means 'φ is distributed explicit knowledge among the agents in G', $D_G\varphi$ means 'φ is distributed implicit knowledge among the agents in G' and $A_G\varphi$ means 'group G is aware of φ'.

Our aim is to present the minimum conditions under which we can define the collective awareness, that is, under which expressions such as 'group G is aware of

φ' make sense. We want to analyze the results of combining this notion with two different senses of D_G in order to describe the different concepts of explicit group knowledge: ($D1$) previously defined and ($D2$) presented later.

Group knowledge attempts to reflect how the knowledge is gained after an information exchange between the members of the group. This can be thought of as a new agent representing the group (the wise man) who possesses the information resulting from the exchange. Since the information being exchanged is explicit knowledge, the awareness of the wise man needs to be the result of the interaction of the individual awarenesses. In a natural way this knowledge exchange needs to have an impact on the collective awareness of the group.

On the formal account we can reduce the collective awareness to the intersection of all the information of the individual awarenesses (*pure awareness intersection*). Nevertheless, being less strict about this matter we can suppose that the collective awareness will contain *at least* this intersection (*awareness intersection*) (AI). This less strict version enables us to reflect the dynamic aspects of communication, the fact that after the information exchange the individual awarenesses acquire the new shared information. The way in which we represent this notion is by allowing the collective awareness of the wise man to include the additional information that results from communication and that does not belong to, or cannot be a result of, the awareness intersection.

On the other hand, the new information of the collective awareness, that does not belong to any individual, has its origin in the communication itself. Then we can state the *principle of limited collective awareness* (PLCA), according to which the content of the collective awareness can only be generated by the interaction of the information of the individual awarenesses.

In general, the agents can communicate everything they know or not, depending on the context. Thus, there will be some information that they will not communicate to the others, but which nevertheless belongs to their individual awareness. We can consider both these type of models and, furthermore, distinguish between those cases where all the information they communicate is knowledge (*full rational communication*) and where not necessarily all of it is knowledge (*partial rational communication*).

2 Epistemic logic with distributed knowledge and collective awareness

Consider a countable set of propositional letters $\mathcal{P}$ and a finite set of agents $\mathcal{A}g = \{1, \ldots, n\}$, the language $\mathcal{L}^{DA^c}$ of epistemic logic with distributed knowledge and collective awareness is given by the following definition:

$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \to \varphi \mid K_i\varphi \mid K_i^e\varphi \mid D_G\varphi \mid D_G^e\varphi \mid A_i\varphi \mid A_G\varphi$

(where $p \in \mathcal{P}$, $i \in G \subseteq \mathcal{A}g$)

A frame is a tuple $\mathcal{F} = (S, R_1, \ldots, R_n, \mathcal{A}_1, \ldots, \mathcal{A}_n, \mathcal{A}_G)$, where:

1. S is a non-empty set of sates (also called 'worlds').
2. $R_i \subseteq S \times S$ for all $1 \leq i \leq n$. Each R_i is an accessibility relation for agent i.
3. $\mathcal{A}_i \colon S \longrightarrow 2^{\mathcal{L}^{DA^c}}$ for all $1 \leq i \leq n$.
4. $\mathcal{A}_G \colon S \longrightarrow 2^{\mathcal{L}^{DA^c}}$, where $\mathcal{A}_G$ satisfies: for any $s \in S$,

   (AI): $\bigcap_{i \in G} \mathcal{A}_i(s) \subseteq \mathcal{A}_G(s)$,

   (PLCA): $\mathcal{A}_G(s) \subseteq For(ATOM(\bigcup_{i \in G} \mathcal{A}_i(s)))$.

   where

   $For(ATOM(\bigcup_{i \in G} \mathcal{A}_i(s)))$ is the set of formulas generated by the atoms that appear in the formulas of $\bigcup_{i \in G} \mathcal{A}_i(s)$ (*the awareness set in s*).

Note that we did not specify any special properties for the individual awareness on item 3. It is wellknown [2] that $\mathcal{A}_i$ can have different properties (closed under subformulas, generated by a subset of primitive propositions, etc.). Analogous considerations can be expressed regarding collective awareness and depending on the properties of the individual awarenesses. Note also that the two general conditions of 'awareness intersection' and 'limited collective awareness', included on item 4, are the minimum intuitive requirements we impose on the collective awareness frames.

(AI) says that $\mathcal{A}_G(s)$ contains at least the intersection of individual awarenesses (before communication) and it can be expanded with new information (after communication). This new information represents the modifications of the individual awarenesses after communication. This mechanism works similarly to the D_G operator. This operator collects what the agents know after communication, but the model can only reflect, as a picture, what the agents know before communication. In this regard we are dealing with static models.

(PLCA) says that after communication, the collective awareness cannot have more information than the information contained by the set of formulas generated by the atoms of the awareness set. For instance, if no agent in the group has notice about the trigeminus, it is impossible that the information 'the trigeminus is a nerve' can appear in the collective awareness after communication.

A model is a tuple $\mathcal{M} = \{\mathcal{F}, V\}$, where $\mathcal{F}$ is a frame and V is a valuation function $V \colon \mathcal{P} \longrightarrow 2^S$ such that V associates every $p \in \mathcal{P}$ with a subset of S, intuitively the states in which p is true. In addition, a satisfaction relation $\models$ between models and formulas in $\mathcal{L}^{DA^c}$ can be defined. We write $\mathcal{M}, s \models \varphi$ to mean that the formula φ is true at (satisfied in) state s in $\mathcal{M}$ and it can be inductively defined as follows:

$$
\begin{array}{lll}
\mathcal{M}, s \models p & \text{iff} & s \in V(p) \qquad \text{(for each } p \in \mathcal{P}) \\
\mathcal{M}, s \models \neg\varphi & \text{iff} & \mathcal{M}, s \not\models \varphi \\
\mathcal{M}, s \models \varphi \wedge \psi & \text{iff} & \mathcal{M}, s \models \varphi \text{ and } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models \varphi \rightarrow \psi & \text{iff} & \mathcal{M}, s \not\models \varphi \text{ or } \mathcal{M}, s \models \psi \\
\mathcal{M}, s \models K_i\varphi & \text{iff} & \text{for all } t \text{ such that } (s,t) \in R_i\text{: } \mathcal{M}, s \models \varphi \\
\mathcal{M}, s \models A_i\varphi & \text{iff} & \varphi \in \mathcal{A}_i(s) \\
\mathcal{M}, s \models K_i^e\varphi & \text{iff} & \mathcal{M}, s \models K_i\varphi \text{ and } \mathcal{M}, s \models A_i\varphi \\
\mathcal{M}, s \models A_G\varphi & \text{iff} & \varphi \in \mathcal{A}_G(s)
\end{array}
$$

We can extend the satisfaction relation with both the following alternatives for distributed knowledge, ($D1$) introduced before and ($D2$) below (see [3]):

$$\mathcal{M}, s \models D_G\varphi \quad \text{iff} \quad \mathcal{M}, t \models \varphi, \text{ for all } t \text{ such that } (s,t) \in \textstyle\bigcap_{i \in G} R_i \qquad \text{(D1)}$$

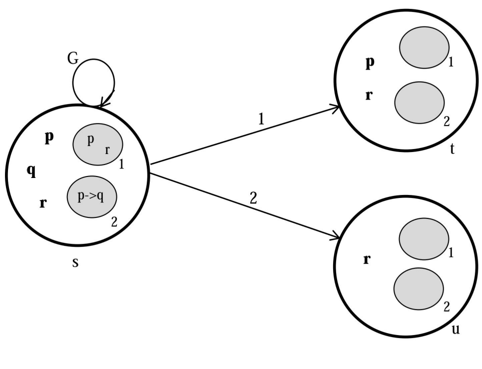
$$\mathcal{M}, s \models D_G\varphi \quad \text{iff} \quad \{\psi \in \mathcal{L}^{A^c} \mid \mathcal{M}, s \models K_i\psi \text{ for some } i \in G\} \Vdash \varphi \qquad \text{(D2)}$$

We also have:

$$\mathcal{M}, s \models D_G^e\varphi \quad \text{iff} \quad \mathcal{M}, s \models D_G\varphi \text{ and } \mathcal{M}, s \models A_G\varphi$$

Note that in ($D2$) we use $\mathcal{L}^{A^c}$, i.e., the language resulting from $\mathcal{L}^{DA^c}$ by dropping D_G and D_G^e, avoiding this way circularity in the definition, as pointed out in [3]. On the other hand, the symbol $\Vdash$ is a relation of *logical consequence* between a set of formulas and one formula. In general, $\Phi \Vdash \varphi$ means that for every model $\mathcal{M}$ and every state s in $\mathcal{M}$, if all formulas in Φ are satisfied in s, φ is also satisfied in s. In addition, the notions of *satisfiability* and *validity* are defined as usual.

Example 1. *In the following model $\mathcal{M} = (S, R_1, R_2, \mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_G, V)$ we take s as the actual state and $G = \{1,2\}$. We define the model only attending to atoms p, q, r:*



- $S = \{s, t, u\}$.
- $R_1 = \{(s,s),(s,t)\}$; $R_2 = \{(s,s),(s,u)\}$.
- $\mathcal{A}_1(s) = \{p, r\}$; $\mathcal{A}_2(s) = \{p \rightarrow q\}$; $\mathcal{A}_1(t) = \mathcal{A}_2(t) = \mathcal{A}_1(u) = \mathcal{A}_2(u) = \varnothing$.
- *$\mathcal{A}_G(s)$ is defined below in different ways; $\mathcal{A}_G(t) = \mathcal{A}_G(u) = \varnothing$.*

- $V(p) = \{s, t\}$; $V(q) = \{s\}$; $V(r) = S$.

Before communication we have:

- $\mathcal{M}, s \models K_1 p \qquad \mathcal{M}, s \models K_1 r \qquad \mathcal{M}, s \models K_2(p \to q)$

The agents can interchange information. And what happens after communication? We can contemplate several possibilities. In all of them we have the same result using (D1) or (D2):

1. • $\mathcal{A}_G(s) = \bigcap_{i \in G} \mathcal{A}_i(s) = \emptyset$ *(there is no communication at all)*
   The distributed implicit knowledge is infinite ($\mathcal{M}, s \models D_G p, \mathcal{M}, s \models D_G r, \ldots$), and the distributed explicit knowledge does not increase (it was empty and remains empty).
2. • $\mathcal{A}_G(s) = \{r\}$
   1 does not speak about all his knowledge. 2 does not speak at all. We will focus on the distributed explicit knowledge. As a consequence: $\mathcal{M}, s \models D_G^e r$ (only!).
3. • $\mathcal{A}_G(s) = \{p, q, r, p \to q\}$
   1 speaks about all his knowledge. 2 speaks about all his knowledge. In particular they can conclude q, since $\mathcal{M}, s \models D_G q$ and, as $q \in \mathcal{A}(s)$, we obtain $\mathcal{M}, s \models D_G^e q$.
4. • $\mathcal{A}_G(s) = \{p, p \to q\}$
   1 speaks about part of his knowledge. 2 speaks about all his knowledge. Although they communicate p and $p \to q$ they cannot conclude q despite all (they may lack Modus Ponens). Indeed, though $\mathcal{M}, s \models D_G q$ we have $\mathcal{M}, s \not\models D_G^e q$, since $q \notin \mathcal{A}(s)$.

3 Models with rational information flow

We say that there is 'rational information flow' in a group whenever the collective awareness of a group acquires knowledge from the individual agents after communication. We can define two versions of rational information flow: (*i*) all the acquired information needs to be knowledge (strong version), or (*ii*) at least part of the acquired information is knowledge (weak version). We can also specify two ways in which the information flows: either all explicit knowledge is acquired or only part of it.

We are interested in collecting classes of structures with rational communication flow and in defining the concept of explicit distributed knowledge in relation to this property. Note that the minimum conditions (AI) and (PLCA) do not commit themselves to neither of these versions. There is also no guarantee that those intersections cannot be empty. However, if there is a real knowledge exchange between the agents in the group those sets can never be empty. The rational communication flow models are of interest because they ensure fruitful knowledge exchange, where the agents have really learned new information.

In what follows we will use the following notation: We will call $KS_G(\mathcal{M}, s)$ and $KS^e_G(\mathcal{M}, s)$ respectively *implicit knowledge set* and *explicit knowledge set* of a group of agents G in a state s of a model $\mathcal{M}$, defined as follows:

$$KS_G(\mathcal{M}, s) = \{\psi \in \mathcal{L}^{DA^c} \mid \mathcal{M}, s \models K_i\psi \text{ for some } i \in G\}$$

$$KS^e_G(\mathcal{M}, s) = \{\psi \in \mathcal{L}^{DA^c} \mid \mathcal{M}, s \models K^e_i\psi \text{ for some } i \in G\}$$

Consider the following four possibilities for A_G that reflect different forms of communication:

$$\mathcal{A}_G(s) = \bigcap_{i\in G} \mathcal{A}_i(s) \tag{$A1$}$$

$$\mathcal{A}_G(s) \subseteq \{\psi \in \mathcal{L}^{DA^c} \mid KS^e_G(\mathcal{M}, s) \Vdash \psi\} \tag{$A2$}$$

$$\{\psi \in \mathcal{L}^{DA^c} \mid KS^e_G(\mathcal{M}, s) \Vdash \psi\} \subseteq \mathcal{A}_G(s) \tag{$A3$}$$

$$\{\psi \in \mathcal{L}^{DA^c} \mid KS^e_G(\mathcal{M}, s) \Vdash \psi\} \cap \mathcal{A}_G(s) \neq \varnothing \tag{$A4$}$$

Combining these notions of $\mathcal{A}_G$ with ($D1$) and ($D2$) we are able to model many ways of knowledge transfer between the agents.

If we assume ($A1$), then two different things may happen: either there is no communication at all, or everything the agents communicate is already known by them. If we assume ($A2$), ($A3$) or ($A4$) there can be information that the group explicitly knows without the need that any of their members do. The group acquires this knowledge after deriving it from the knowledge of their members. On the other hand, regarding the specific case of ($A2$), the collective awareness is only 'rational'; that is, it only contains the logical consequences of the explicit knowledge of their members.

In the case of ($A3$) and ($A4$) the collective awareness has a 'rational core', $\{\psi \in \mathcal{L}^{DA^c} \mid KS^e_G(\mathcal{M}, s) \Vdash \psi\}$, standing for the information that can be derived from the explicit knowledge set. Regarding ($A3$) the agents communicate all their knowledge. But the collective awareness is not necessarily reduced to its rational core. This strikes us more intuitive since the awareness can contain inconsistent information. Assuming ($A4$), there can be members of the group that do not communicate all their knowledge. This has a direct impact on the explicit knowledge of the group which does not contain everything its members really know.

4 Classes of models and full explicit communication

The *Principle of Full Communication* establishes that whenever φ is considered group knowledge, it should be possible for the members of the group to establish φ through communication. It is argued by Van der Hoek et al., in [5], that group knowledge should comply with this principle. They formulate it as follows:

$$\mathcal{M}, s \models D_G\varphi \text{ implies } KS_G(\mathcal{M}, s) \Vdash \varphi$$

The authors use the language $\mathcal{L}^D$ (resulting from $\mathcal{L}^{DA^c}$ by dropping the explicit epistemic and awareness operators). A dissertation about the class of models that comply with this principle using (D1) can be found in [4]. Since we want to deal with real agents whose reasoning resources are limited, the knowledge that is being established through communication needs to be explicit. Hence we can establish the principle of *full explicit communication*:

$$\mathcal{M}, s \models D^e_G\varphi \text{ implies } KS^e_G(\mathcal{M}, s) \Vdash \varphi$$

Full explicit communication can be studied combining the definitions of distributed implicit knowledge, $(D1)$ or $(D2)$, and the conditions on the collective awareness, $(A1)$-$(A4)$ above. The result of this combination is given by the following propositions:

Proposition 1. *In the following classes of models distributed explicit knowledge does not comply with the principle of full explicit communication:*

1. *The class of models that satisfies* $(D1)$ *and either* $(A1)$ *or* $(A3)$ *or* $(A4)$*.*
2. *The class of models that satisfies* $(D2)$ *and either* $(A1)$ *or* $(A3)$ *or* $(A4)$*.*

Proof. We will prove in item 2 the case $(D2)$ and $(A4)$. Let $G = \{1, 2\}$ and consider the model $(S, R_1, R_2, \mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_G, V)$, where:
$S = \{s\}$; $R_1 = \{(s, s)\}$; $R_2 = \emptyset$; $\mathcal{A}_1(s) = \{p, q\}$; $\mathcal{A}_2(s) = \emptyset$; $V(p) = \{s\}$; $V(\varphi) = \emptyset$ for all atom φ in $\mathcal{P}$ distinct of p. Assume also that $\mathcal{A}_G(s) = \{p, q\}$ which satisfies (A4), because $KS^e_G(\mathcal{M}, s) = \{p\}$. Now, we have that $\mathcal{M}, s \models D_Gq$ (since $\mathcal{M}, s \models K_2q$) and as $q \in \mathcal{A}_G(s)$, we obtain $\mathcal{M}, s \models D^e_Gq$, but $\{p\} = KS^e_G(\mathcal{M}, s) \nVdash q$. □

The following result is immediate:

Proposition 2. *In the following classes of models distributed knowledge complies with the principle of full explicit communication:*

1. *The class of models that satisfies* $(D1)$ *and* $(A2)$*.*
2. *The class of models that satisfies* $(D2)$ *and* $(A2)$*.*

5 Conclusions and future work

We have seen that collective awareness is an adequate concept for modeling communication with information exchange in a group of agents. This notion reflects, in a static way, the dynamics of communication allowing changes and integrating new information. But there are still many unexplored areas in this field, such as: (i) Exploring more classes of models that comply with the principle of full explicit communication and defining formal systems to deal with this concept syntactically. (ii) Redefining explicit distributed knowledge specifying the type of information that collective awareness can contain. (*iii*) Analyzing the concepts of distributed explicit knowledge and collective awareness from the perspective of Dynamic Epistemic Logic (DEL) and Public Announcement Logic (PAL).

Acknowledgements. This paper was partially supported by the Spanish Research Project TIN2012-39353-C04-01.

References

[1] R. Fagin, J.Y. Halpern. Belief, Awareness, and Limited Reasoning. *Artificial Intelligence*, 34: 39–76, 1988.

[2] R. Fagin, J.Y. Halpern, Y. Moses, M.Y. Vardi. *Reasoning About Knowledge.* The MIT Press, 1995.

[3] J. Gerbrandy. Distributed Knowledge. *Twendial'98: Formal Semantics and Pragmatics of Dialogue. (Joris Hulstijn and Anton Nijholt, eds.)*, TWLT 13: 111–124, 1998.

[4] F. Roelofsen. Distributed knowledge. *Journal of Applied Non-Classical Logics*, 17:2, 255–273, 2007.

[5] W. van der Hoek, B. van Linder, John-Jules Meyer. Group knowledge is not always distributed (neither is it always implicit). *Mathematical Social Sciences*, 38: 215–240, 1999.

A LOGICAL APPROACH FOR DIRECT-OPTIMAL BASIS OF IMPLICATIONS

Estrella Rodríguez-Lorenzo, Pablo Cordero,
Manuel Enciso, Angel Mora
Universidad de Málaga, Andalucía Tech- Spain
estrellarodlor,amora@ctima.uma.es, pcordero,enciso@uma.es

Abstract

In Formal Concept Analysis, knowledge extracted from a data set is represented in two alternative ways: concept lattices and sets of implications. The sets of implications are optimized under different criteria linked to several properties. In this paper the optimization task is strongly based on the Simplification Logic. Specifically, we present a review of how minimal sets of implications (basis) with different properties can be calculated with a logical style. Therefore, different techniques to manipulate them are outlined. Our logic-based approach property fits with the logic programming paradigm and, thus, a Prolog implementation to calculate direct basis from a set of implications is also sketched .

1 Introduction and background

Formal Concept Analysis (FCA) is an useful tool for mining information from a dataset. FCA has been used in different areas: Artificial Intelligence, Databases, Software Engineering, Data Mining, and recently in the Semantic Web.

In this section, we summarize the main concepts regarding FCA. For a more detailed explanation, we refer the reader to [7]. Data are represented through binary tables, named formal contexts $\mathbb{K} := (G, M, I)$, in which a set of objects G and a set of attributes M are related via the binary relation I. From $\mathbb{K}$, two mappings are defined:

- $(\ )' \colon 2^G \to 2^M$ where $A' = \{m \in M \mid g\ I\ m \text{ for all } g \in A\}$ for all $A \subseteq G$.

- $(\ )' \colon 2^M \to 2^G$ where $B' = \{g \in G \mid g\ I\ m \text{ for all } m \in B\}$ for all $B \subseteq M$.

In FCA several automated method have been introduced to extract knowledge from formal concepts. This knowledge is extracted in the shape of *concepts* and they can be represented in the so called *Concept Lattice*. A concept is a pair $\langle A, B\rangle \in 2^G \times 2^M$ such that $A' = B$ and $B' = A$ (i.e. a set of objects that are precisely characterized by a set of attributes) and an order relation is established providing a hierarchy in the concept set.

In this paper, we deal with an alternative way to represent this knowledge, that is the set of *implications*. An attribute implication is an expression $A \to B$ where A and B are sets of attributes. A formal context satisfies $A \to B$ if every object that has all the attributes in A also has all the attributes in B. In other terms, $A \to B$ holds (is valid) in $\mathbb{K}$ whenever $A' \subseteq B'$.

The set of all valid implications in a context, called *full implicational system*, gathers the same knowledge as its corresponding concept lattice. However, the first -alternative- approach provides an interesting advantage: since it satisfies the Armstrong's axioms [**?**], some subsets can be considered as representatives of the full implicational systems. Thus, an *implicational system* (briefly IS) for $\mathbb{K}$ is a set Σ of implications satisfing that the valid implications on $\mathbb{K}$ are those that can be derived from Σ using Armstrong's axioms. That is, the implicational system Σ represents all the knowledge obtained from $\mathbb{K}$ in a shorten way. Implicational System knowledge representation is strongly related with two issues:

1. Do Armstrong's axioms can be used efficiently?
2. Since several implicational systems can represent the same knowledge, there is an optimal one?

The first question use to be addressed with indirect methods based on the semantics, avoiding a direct syntactic manipulation provided by Armstrong's inference system. An alternative way that remains faithful to the logic point of view stems from the Simplification Logic. In Section 2 we present this logic that allows an efficient reasoning with implications.

The second question is addressed by characterizing those implicational systems fulfilling some minimality criteria. Such Implicational Systems are usually called *basis*. Among the different basis definitions, the Duquenne-Guigues basis [8], also called Stem basis, has been widely accepted in FCA. This basis is minimal in the number of implications. Nevertheless, as it is shown in [6], Duquenne-Guigues bases tend to have redundant attributes and therefore, an equivalent one having the same cardinality but less attributes can be provided. In that paper, we also propose a method to obtain a basis with minimal size in the left-hand side of the implications.

Other well-known property used to define another kind of bases is *directness*, i.e., a single traversal of the implicational system is enough to compute the closure of an given set of attributes. A basis fulfilling this property is named *direct basis*. This property is usually accompanied by some minimality criteria. We are particularly interested in those ones with minimum size (number of attributes). In [2, 3, 11] several methods to calculate the direct-optimal basis are introduced, where minimality and directness have been joined in the same notion of basis.

Here, our main issue is how to calculate such a direct-optimal basis, providing a tool for a very efficient computation of attribute closures. Our method to calculate the direct-optimal basis [11] is based on Simplification Logic [5], $\mathbf{SL}_{\text{FD}}$, a sound and complete inference system for Implicational Systems. Our logic is strongly based on the Simplification Rule, which describes the redundancy removal of attributes. This method based on $\mathbf{SL}_{\text{FD}}$ is more efficient than previous methods appeared in the literature. To prove this fact, we have developed an illustrative empirical test using Prolog.

2 Simplification Logic and closures

Armstrong's Axioms [1] is the former system introduced to manage implications in a logical style. In this section, we briefly present Simplification Logic ($\mathbf{SL}_{\text{FD}}$ for short), which is an equivalent logic that arises from the idea of simplifying the set of implications by efficiently removing redundant attributes [10]. [1]

The language: Given a non-empty finite alphabet S (namely attributes set), the language of $\mathbf{SL}_{\text{FD}}$ is $\mathcal{L}_S = \{A \to B \mid A, B \subseteq S\}$.

In order to distinguish between language and metalanguage, inside implications, AB means $A \cup B$ and A-B denotes the set difference $A \smallsetminus B$.

Semantics: A context $\mathbb{K}$ is said to be a model for $A \to B \in \mathcal{L}_S$, denoted by $\mathbb{K} \models A \to B$, if this implication holds in the context. For an IS Σ, $\mathbb{K} \models \Sigma$ means $\mathbb{K} \models A \to B$ for all $A \to B \in \Sigma$. If Σ_1 and Σ_2 are implicational systems, $\Sigma_1 \equiv \Sigma_2$ denotes the equivalence of the two sets of implications (i.e. $\mathbb{K} \models \Sigma_1$ iff $\mathbb{K} \models \Sigma_2$ for all context $\mathbb{K}$).

Syntactic derivations: Reflexivity as axiom scheme and the following inference rules named fragmentation, composition and simplification are considered in $\mathbf{SL}_{\text{FD}}$.

$$\texttt{[Ref]}\ \frac{}{A \to A} \qquad \texttt{[Frag]}\ \frac{A \to BC}{A \to B} \qquad \texttt{[Comp]}\ \frac{A \to B,\ C \to D}{AC \to BD} \qquad \texttt{[Simp]}\ \frac{A \to B,\ C \to D}{A(C\text{-}B) \to D}$$

[1] See [9] for a more detailed presentation of the Simplification Logic and its advantadges.

Given a set of implications Σ and an implication $A \to B$, $\Sigma \vdash A \to B$ denotes that $A \to B$ can be derived from Σ by using the axiomatic system in a standard way. If any implication valid in a formal context $\mathbb{K}$ can be derived from Σ and vice versa, then Σ is called an implicational system (IS) for $\mathbb{K}$.

The main advantage of $\mathbf{SL}_{\text{FD}}$ is that its inferences rules induce equivalence relations among sets of implications. Moreover, these equivalencies are enough to compute all the derivations (see [9] for further details and proofs).

Theorem 2.1 ([9])**.** *In* $\mathbf{SL}_{\text{FD}}$ *logic, the following equivalences hold:*

1. *Fragmentation Equivalency* **[FrEq]**: $\{A \to B\} \equiv \{A \to B\text{-}A\}$
2. *Composition Equivalency* **[CoEq]**: $\{A \to B, A \to C\} \equiv \{A \to BC\}$
3. *Simplification Equivalency* **[SiEq]**: *If* $A \cap B = \emptyset$ *and* $A \subseteq C$ *then*

$$\{A \to B, C \to D\} \equiv \{A \to B, C\text{-}B \to D\text{-}B\}$$

Note that these equivalencies (read from left to right) remove redundant information, approaching our main spirit when creating $\mathbf{SL}_{\text{FD}}$.

Definition Let $\Sigma \subseteq \mathcal{L}_S$ be an IS and $X \subseteq S$. The closure of X wrt Σ is the largest subset of S, denoted X^+_Σ, such that $\Sigma \vdash X \to X^+_\Sigma$.

3 Direct-Optimal basis

A mainstream topic in FCA is the study of different properties to be fulfilled by implicational systems. As we have mentioned in the introduction, our goal is the minimization of the computation of attribute closure computations. In [3], Bertet and Monjardet present a survey concerning implicational systems and basis. They show the equality among five basis presented in different works. They also study the properties they satisfy, including directness and minimality. The conclude that all the presented bases are equivalent to the so called direct-optimal basis.

The formal definition of these properties (minimality, optimality and directness) is the following:

Definition An IS Σ is said to be:

- *minimal* if $\Sigma \smallsetminus \{A \to B\} \not\equiv \Sigma$ for all $A \to B \in \Sigma$,
- *minimum* if $\Sigma' \equiv \Sigma$ implies $|\Sigma| \leq |\Sigma'|$, for all IS Σ',
- *optimal* if $\Sigma' \equiv \Sigma$ implies $||\Sigma|| \leq ||\Sigma'||$, for all IS Σ',

where $|\Sigma|$ is the cardinal of Σ and $||\Sigma||$ denotes its size, i.e. $||\Sigma|| = \sum_{A\to B\in\Sigma}(|A| + |B|)$.

An IS is said to be a basis if it is minimal. We are looking for bases satisfying this property because the less cardinal of the IS, the better the performance of closure computation. Moreover, to reduce the cost of the computation of closures, we demand for another criterion: directness. It ensures that the closure computation only needs one traversal of the IS.

Although closure is a linear task, the search for fast and easy closure methods is a hot topic because several problems are addressed by exhaustively computing closures. Thus, many of the classical algorithms in FCA are solved by intensively computing the closure of a set of attributes. A significant reduction in the performance of closure methods is relevant when a huge -sometimes exponential-number of closures are executed to solve the original problem.

For this reason, we have paid attention to the notion of direct-optimal basis [2, 3], introduced as follows:

Definition Let S be a set of attributes, an IS Σ is said to be *direct* if, for all $X \subseteq S$:

$$X^+_\Sigma = X \cup \{b \in B \mid A \subseteq X \text{ and } A \to B \in \Sigma\}$$

Moreover, Σ is said to be *direct-optimal* if it is direct and, for any direct IS Σ', $\Sigma' \equiv \Sigma$ implies $||\Sigma|| \leq ||\Sigma'||$.

In other words, Σ is said to be direct-optimal if it is direct and it is optimal among all the equivalent direct ISs. In [3], the existence and the unicity for a direct-optimal basis equivalent to a given one was proved.

In the following section, we are introducing a method to calculate the direct-optimal basis for any IS proposed in [11] and its improved version proposed in [12].

4 Computing direct-optimal basis

This section deals with the integration of the techniques proposed by Bertet et al. [2–4] and the Simplification Logic proposed by Cordero et al. [5]. First, we developed a function to get the direct-optimal basis whose first step is the narrowing of the implications (see [11]). To this end, in this paper we use *reduced* ISs.

Definition An IS Σ is *reduced* if $B \neq \emptyset$ and $A \cap B = \emptyset$ for all $A \to B \in \Sigma$.

Obviously, an arbitrary IS Σ can be turned into a reduced equivalent one Σ_r by applying **[FrEq]**, and by removing implications of the form $A \to \emptyset$. The method proposed here to get a direct optimal basis begins with this transformation, preserving reduceness in further steps.

In [3, 4] the authors apply two completely separated stages, first is focussed in directness and, later, an optimization stage is carried out by removing redundant implications. In our method, we have introduced a new inference rule covering in juts one step both properties: directness and minimality. The kernel of the new method is the so named Strong Simplification:

$$\texttt{[sSimp]} \quad \text{If } B \cap C \neq \emptyset \text{ and } D \nsubseteq A \cup B, \ \frac{A \to B, C \to D}{A(C\text{-}B) \to D\text{-}(AB)} \tag{1}$$

The exhaustively application of this rule to a reduced IS renders an equivalent direct and reduced implicational system, direct-reduced IS in the following. As we proved in [12], the implicational system Σ_{dr} generated from an IS Σ is defined as the smallest one containing Σ which is closed for `[sSimp]`.

As a final step, the three first equivalencies from Theorem 2.1 are used to remove redundant information preserving directness. The implicational system generated in this way by applying these equivalences is named *simplified* IS.

To conclude, the method turns the direct-reduced implicational system obtained in previous stages into an equivalent simplified-direct-reduced one Σ_{sdr} [12]. Indeed, Σ_{sdr} is exactly the direct-optimal basis.

Although the direct-optimal basis is unique, the cost of its computation varies depending on the proposed method. So, to efficiently solve the original problem demanding closure computation, a reduction in the computation of the basis is demanded. The exponential cost of this process is due to the generation of the direct implicational system. In [12], we reduce the input of this stage. The reduction step described above is substituted by simplification, providing a greater reduction in the redundancy. Now, simplification is achieved by applying all the four equivalences in Theorem 2.1 to remove all the redundant attributes in the implications.

Finally, we include here a Prolog implementation[2] based on $\mathbf{SL}_{\text{FD}}$ to calculate the direct-optimal basis from a IS. Due to the fact that our methods are based on logic, Prolog prototypes can be developed in a more direct way.

The input of the Prolog program is a set of attributes S and a set of implications Σ over the attributes in S. The output is the direct optimal basis equivalent to this set of implications. The main predicate of the method developed is `directoptimalSL`. There are three main operations in the method: the first one, `applySL` predicate, executes the four equivalences of $\mathbf{SL}_{\text{FD}}$ the second one,

[2] Available at `http://www.lcc.uma.es/~enciso/do2Simp.zip`

`sSimp`, is a predicate which applies exhaustively the [sSimp] rule (1) to any pair of implications to obtain a direct IS. The last one `applySimplification` renders the simplified-direct-reduced basis: Σ_{sir}. When an implication is added in one step of this execution, the flag `fixpoint` takes the value `false` in order to repeat the method again until the fixpoint is reached. An sketch of this process is showed here.

```
directoptimalSL(Input,Output):-
     ...
     fixPoint_Non,
     applySL,
     applysSimp,
     applySimplification,!.
applySL:-
    siEq, rSiEq, CoEq,FrEq,
    applySL.
applySL.
applysSimp:-
   read2implications(implication(A,B),
                     implication(C,D)),
    sSimp(implication(A,B),implication(C,D),
    fail.
applysSimp.
applySimplification:-
    siEq, CoEq,FrEq,
    applySimplification.
applySimplification.
sSimp(implication(A,B),implication(C,D),
  implication(ACminusB,DminusAB)):-
     union(A,C,AC), difference(AC,B,ACminusB),
     union(A,B,AB), difference(D,AB,DminusAB),
     fixpoint(false),!.
```

Example In this example, we will compute the direct basis of the following set of implications stored in a file called `ganter.txt`:

```
implication([a],[b,c]).
implication([d],[b]).
implication([c],[b]).
implication([a,b,c,d],[e,g]).
implication([a,b,c,e],[d,g]).
```

We call the Prolog predicate:

```
directoptimalSL('ganter.txt','Outganter.txt').

-> Equivalences:     CoEq + SiEq
       implication([c],[b]) +
implication([a,b,c,e],[d,g]) |---
        implication([a,c,e],[d,g])  added
....
***    A DIRECT IS
implication([c], [b]).
implication([d], [b]).
implication([a, b, c, e], [d, g]).
implication([a, b, c, d], [e, g]).
implication([a], [b, c]).
implication([a, c, e], [d, g]).
implication([a, c, d], [e, g]).
implication([a, e], [d, g]).
implication([a, d], [e, g]).

*** BEGIN   Simplification  **
->Equivalences:  SiEq
   implication([c],[b]) +
   implication([a,b,c,e],[d,g]) |---
   implication([a,b,c,e],[d,g])  removed
   implication([a,c,e],[d,g]) yet exist
...
** OUTPUT: DIRECT OPTIMAL BASIS
implication([c], [b]).
implication([d], [b]).
implication([a], [b, c]).
implication([a, e], [d, g]).
implication([a, d], [e, g]).
```

5 Conclusions

In this work, we have outlined two methods to calculate the direct-optimal basis. We apply such methods to reduce the cost in closure computations. In FCA, several methods exhaustively calculate closures of attribute sets and this kind of basis allows their computation in just one traverse of the Implicational System. Prolog has been used as an useful tool to quickly develop prototypes of these methods. Due to space limitations a comparison is omitted and it could be the goal of an extended work. The development of an integrated tool with all the algorithms to manipulate implications in the direct-optimal issue is a future work.

Acknowledgment

This work has been partially supported by grants TIN11-28084 and TIN2014-59471-P of the Science and Innovation Ministry of Spain.

References

[1] W W. Armstrong, *Dependency structures of data base relationships*, Proc. IFIP Congress. North Holland, Amsterdam: 580–583, 1974.

[2] K. Bertet, M. Nebut, *Efficient algorithms on the Moore family associated to an implicational system*, DMTCS, 6(2): 315–338, 2004.

[3] K. Bertet, B. Monjardet, *The multiple facets of the canonical direct unit implicational basis*, Theor. Comput. Sci., 411(22-24): 2155–2166, 2010.

[4] K. Bertet, *Some Algorithmical Aspects Using the Canonical Direct Implicationnal Basis*, CLA:101–114, 2006.

[5] P Cordero, A. Mora, M. Enciso, I.PéÂŐrez de Guzmán, *SLFD Logic: Elimination of Data Redundancy in Knowledge Representation*, LNCS, 2527: 141–150, 2002.

[6] P. Cordero, M. Enciso, A. Mora, M. Ojeda-Aciego, *Computing Left-Minimal Direct Basis of implications*. CLA: 293–298, 2013.

[7] B. Ganter, *Two basic algorithms in concept analysis*, Technische Hochschule, Darmstadt, 1984.

[8] J.L. Guigues and V. Duquenne, *Familles minimales d'implications informatives résultant d'un tableau de données binaires*, Math. Sci. Humaines: 95, 5–18, 1986.

[9] A. Mora, M. Enciso, P. Cordero, and I. Fortes, *Closure via functional dependence simplification*, I. J.of Computer Mathematics, 89(4): 510–526, 2012.

[10] A. Mora, M. Enciso, P. Cordero, and I. Pérez de Guzmán, *An Efficient Preprocessing Transformation for Functional Dependencies Sets Based on the Substitution Paradigm*, LNCS, 3040: 136–146, 2004.

[11] E. Rodríguez-Lorenzo, K. Bertet, P.Cordero, M.Enciso, and A. Mora, *The Direct-optimal basis via reductions*, CLA, 145–156, 2014.

[12] E. Rodríguez-Lorenzo, K. Bertet, P.Cordero, M.Enciso, A. Mora, and M. Ojeda-Aciego, *From Implicational Systems to Direct-Optimal bases: A Logic-based Approach*, Applied Mathematics & Information Sciences, 2L: 305–317, 2015.

Abductive Reasoning in Dynamic Epistemic Logic - Generation and Selection of Hypothesis

Ismael D. Arroniz
Universidad de Sevilla, Andaluca Tech
ismdelarr@gmail.com

Abstract

We propose to represent abductive reasoning in a dynamic epistemic logic framework. The framework emphasizes the role of experience, defined as the result of a dynamic process over an agent™s information, in the generation of hypotheses and the selecting the best one. We collect several insights of different contexts in logic, such as IBE or AKM model, and we introduce two extra criteria. Besides the one based on experience, a traditional criterion using the idea of minimality, a pragmatic one is also explored. We also introduce a method to combine different orders.

1 Introduction

Abduction is one of the most important non-monotonic reasoning processes. Traditionally described as the reasoning that goes from facts to their causes, in the process of looking for explanations. Originally studied by Charles S. Peirce we can present abduction with his words:

The surprising fact, C, is observed.
But if A were true, C would be a matter of course.
Hence, there is reason to suspect that A is true.

Abductive reasoning has been useful in fields such as philosophy of science and cognitive science. Abductive reasoning is also useful in Artificial Intelligence, where it has been applied to diagnosis and natural language understanding tasks [11, 1, 8, 7].

Recent advances using Dynamic Epistemic Logic as framework[13, 10, 12] allow us to study explicitly the actions involved in the abductive process, understand as a process that involves an agent™s information. In this paper, we try to add some components to dynamic epistemic logic in order to describe abductive reasoning step by step. We focus on the semantics for space limit reasons. This work is organisation as follows: We present a semantic model that allows us to study the generation and selection of hypotheses based on experience in Section 2. Then in Section 3 we present our basic definitions for the study of abductive reasoning under this new framework. Section 4 presents a method to select explanations using several criteria. Section 5 recalls other proposals that present abductive reasoning like a belief change process. We finish in Section 6 with a summary of our proposal and further lines of inquiry for future work.

2 Semantic model

Epistemic logic and its possible worlds semantics are a powerful framework that allows us to represent an agent™s information not only about propositional facts, but also to the agent™s information. Other proposals also point to the importance of experience in abductive reasoning [6][11]. In epistemic terms, we understand the experience as the result of a dynamic process of an agent™s information. In this process, the agent will change not only their knowledge but also their beliefs. We try to formalize this notion in the epistemic framework. Moreover, in order to approximate us to a more realistic scenario we use a non-omniscient agent. Logical omniscience, useful in some applications, is an unrealistic idealization in some others. Most of the proposals to solve this problem focus on weakening the properties of the agent™s information (usually by distinguishing between implicit and explicit information). We use a framework based on [14] for representing implicit and explicit beliefs that combines a framework for representing implicit and explicit information with plausibility models for representing beliefs. We add some specific components for the purpose of our research.

Definition 2.1 (Best explanation model)**.** *A best explanation model* BE *is a possible words model* $M = \langle W, \leq, V, A, S, \preceq, C \rangle$ *where* $M = \langle W, \leq, V \rangle$ *is a plausibility model presented in [2] and where:*

- $A : W \to \wp(\mathcal{L})$ *is the acknowledgement set function, indicating the formulas the agent has acknowledged as true at each possible world.*
- $S \subseteq \mathcal{L}$ *is a finite set of formula of the language. Any element will be considerated an explanation (hypothesis).*
- $\preceq \subseteq (S \times S)$ *is a locally well-preorder* [1] *priority relation over* S

[1] For more details consult [2]

$\leq$ relation allow us to talk about the priority that an agent gives to each explanation. If $t \leq u$, u is at least as priorly as t.

- *$C : \mathcal{L} \longrightarrow \mathbb{N}$ is the cost function, assigned a natural number for any formula of the language. We all know that not every explanation is verifiable with the same cost, whether economic or simplicity reasons. In some cases, it™s simpler, faster or cheaper to discard some possibilities that are not priorities for what the agent knows or believes but for their easy verification.*

In this framework, we can define the notions of implicit and explicit knowledge and beliefs. The agent knows φ implicitly if and only if φ is true in all the epistemically indistinguishable worlds. The agent knows φ explicitly if, in addition, she acknowledges it as true in all these worlds. The agent believes φ implicitly if and only if φ is true in the most plausible worlds and the agent believes φ explicitly if, in addition, she acknowledges it as true in these *best* worlds.

Implicit knowledge:	$K_{Im}\varphi := [\sim]\varphi$
Explicit knowledge:	$K_{Ex}\varphi := [\sim](\varphi \wedge A\varphi)$
Implicit belief:	$B_{Im}\varphi := \langle\leq\rangle[\leq]\varphi$
Explicit belief:	$B_{Ex}\varphi := \langle\leq\rangle[\leq](\varphi \wedge A\varphi)$

3 Abductive problem and abductive solution

When an agent observes a surprising fact, there is no element of uncertainty about it. We use a public announcement definition [3] modified for our non-omniscient logic to represent this action on the model.

Definition 3.1 (Observation). *Given a* BE *model $M = \langle W, \leq, V, A, S, \preceq, C\rangle$ and a formula χ of propositional language $\chi \in \mathcal{L}_p$*[2]*, the observation operation $\chi!$ produces a model $M_{\chi!} = \langle W', \leq', V', A', S, \preceq, C\rangle$ where:*

$W' := \{w \in W \mid (M, w) \Vdash \chi\}$
$\leq' := \leq \cap (W' \times W')$ *and*
For all $w \in W'$, $V'(w) := V(w)$ and $A'(w) := A(w) \cup \{\chi\}$

We eliminate all the worlds, where χ is false. Moreover, χ is added to A, a function of acknowledgment. The agent™s knowledge about χ becomes explicit. In epistemic logic terms, we can say that an abductive problem is generated

[2]Formulas of propositional langugage $\mathcal{L}_p$ are given by:

$$\varphi ::= p \mid \neg\varphi \mid \varphi \vee \psi$$

when there exists a formula that was not explicitly known prior to the agent™s observation.

Definition 3.2 (Abductive problem)**.**

$$\chi \textit{ is an abductive problem iff } \chi \in \mathcal{L}_p \textit{ and } (M_{\chi!}, w) \Vdash K_{Ex}\chi \textit{ and } (M, w) \nVdash K_{Ex}\chi$$

Knowing that $(M_{\chi!}, w)$ represents the sub-model obtained when χ is observed, we define an abductive problem as a fact that is not known explicitly in the first instance but it is before the observation and can be expressed in the propositional language.[3]

After the observation, the agent tries to find what they could have been able to infer from the observation that raised the problem. We define this process as an action called Generation in our semantic model. We propose that an abductive solution is a formula that, together with the background theory (including knowledge and beliefs), entails the surprising observation.

Definition 3.3 (Generation)**.** *Given a model $M = \langle W, \leq, V, A, S, \preceq, C\rangle$ y χ, and an abductive problem in (M, w), the generation operation $\chi?$ produces a model $M_{\chi?} = \langle W, \leq, V, A, S', \preceq', C\rangle$ where:*

$$S' = \{\mathtt{DNF}(\psi) \mid \psi \to \chi \in A(w)\} \text{ for all } w \in W$$

S is the explanation set where we generate all the possible candidates of explanations; the antecedents of conditionals that have χ as consequent in the acknowledgement function A We use DNF[4] as a method to standardise and identify any formula for its syntactic form. All candidates generated are sorted based on the background theory and, therefore, on what we call experience.

$\preceq'$ is defined as follows:

$$\psi_1 \preceq' \psi_2 \text{ iff } \begin{cases} (M, w) \Vdash K_{Ex}(\psi_2 \to \chi), \text{or} \\ (M, w) \nVdash K_{Ex}(\psi_2 \to \chi) \text{ and exists} \\ w_1 \in MM(\psi_1 \to \chi), w_2 \in \text{MM}(\psi_2 \to \chi) \text{ such that } w_1 \leq w_2 \end{cases}$$

Where M is the set of world where the implication is acknowledge

$$M(\beta_1 \to \beta_2) = \{w \in W \mid \beta_1 \to \beta_2 \in A(w)\}$$

[3]With this restriction, we avoid technical problems generated by more expressive languages.

[4]$\mathtt{DNF}(\psi)$ is a disjunctive normal form that validates the following relation:

$$\vdash \psi \leftrightarrow \mathtt{DNF}(\psi)$$

and MM is the maximum set of worlds in M

$$\mathrm{MM}(\beta_1 \rightarrow \beta_2) = \{w \in M(\beta_1 \rightarrow \beta_2) \mid \text{ for all } u \in M(\beta_1 \rightarrow \beta_2), u \leq w\}$$

In words, if an hypothesis is part of a implication that an agent knows explicitly, she puts this hypothesis at the top in the priority order. If not, an hypothesis will be more priorly that another if the most plausible world where the agent is acknowledged of the implication that contain the hypothesis as antecedent is more plausible that the maximum plausible world where the agent is acknowledged of the implication that contain the second hypothesis as antecedent.

4 Selecting the best explanation

Some authors argue that abduction is just the Generation step, and together with Selection is part of a more complex process called Inference to the Best Explanation (IBE)[9]. We think the name is not important because everything is part of an explanatory inference, and avoiding this debate, we make significant advances in the study of the selection stage. Traditionally, the criteria to select the best explanation was syntactic, referring to the complexity of the formula. Now we try to formalize a more pragmatic approach. In addition to the experiential criteria, we consider it appropriate to add more ways to sort explanations. According to this, we introduce two criteria more than the experiential approach with the requirement of a method to combine different criteria. We combine them using social-choice techniques[5], and we apply a hierarchy to these criteria. At the top, we consider the experience as the best way to prioritize explanations. We use $\preceq'$ relation detailed in Definition 3.3. At an inferior level, we consider a logic order $\prec_{log}$ that follows a syntactic criterion.

$$\psi_1 \prec_{log} \psi_2$$

if any of the following cases is true:

- ψ_2 is a formula α, being α an atomic formula *(a)*
- ψ_2 is a formula β and ψ_1 is not a formula α, being β the negation of an atomic formula *($\neg a$)*
- ψ_2 is a formula γ and ψ_1 is not a formula α nor β, being γ the disjunction of literals ($\neg A \vee B$)
- ψ_2 is a formula δ and ψ_1 is not a formula α nor β neither γ, being δ the conjunction of literals ($\neg A \wedge B$)

- ψ_2 is a formula ϵ and ψ_1 is not a formula ϵ, being ϵ the disjunction of conjunctions $(A \wedge \neg B) \vee C$

Because all possible explanation generated in Definition 3.3 is in its disjunctive normal form (DNF) we know their syntactic structure is α, β, γ, δ or ϵ kind. Finally, we consider the context criteria $\preceq_C$ that tells us about the contextual differences between explanations. We assign a numerical value to any hypothesis that defines the difficulty of the verification. Using the arithmetic symbol $\geqslant$ we state an order based on $\preceq_C$ the component of the best explanation model BE

$$\psi_1 \preceq_C \psi_2 \text{ syss } C(\psi_1) \geqslant C(\psi_2)$$

In some cases, we need to prioritize explanations based on their practicality. Any of these orders act only when the superior order is not definitive enough to order explanations, resolving the draw. In order to represent this notion in a semantic model, we describe an action that combines three orders with a final priority relation $\preceq_f$:

Definition 4.1 (Selection). *Given a model $M = \langle W, \leq, V, A, S, \preceq, C \rangle$ the operation selection $\Rsh$ produces a model $M_{\Rsh} = \langle W, \leq, V, A, S, \preceq_f, C \rangle$ where:*
$\psi_1 \preceq_f \psi_2$ iff any of the following cases is true:

- $\psi_1 \prec \psi_2$
- *$\psi_1 \precsim \psi_2$ and $\psi_1 \prec_{log} \psi_2$*
- *$\psi_1 \precsim \psi_2$ and $\psi_1 \precsim_{log} \psi_2$ y $\psi_1 \preceq_C \psi_2$*

It is desirable that only one explanation is at maximum. Cases where the final order result with two different explanations at the top may also occur. In this situation, there is need to find some other criterion that distinguishes them. Our method is perfectly applicable to different approaches in the hierarchy of the various orders.

5 Belief revision

At the belief revision stage, we focus on the agent™s information changes. Once an agent establishes an explanation as a definitive best explanation, the agent modifies their beliefs and, therefore, the information they hold about the situation. As noted above, many studies have linked belief revision with abduction [4]. bductive reasoning does not guarantee that the hypothesis is correct. For that reason it cannot eliminate such possible worlds where the best explanation is not true, it just gives them a higher order of plausibility. In our model, we represent it with an action of belief revision, based on other works [2].

Definition 5.1 (Belief Revision)**.** *Let $M = \langle W, \leq, V, A, S, \preceq, C\rangle$ be a model and a let χ be a formula, the belief change $\Uparrow \chi$ operation produces a model $M_{\Uparrow\chi} = \langle W, \leq', V, A, S, \preceq, C\rangle$ differing from M in:*
$w \leq' u$ iff any of the following cases is true:

- $(M, u) \Vdash \chi \wedge A\chi$ *and* $w \leq u$
- $(M, w) \Vdash \neg(\chi \wedge A\chi)$ *and* $w \leq u$
- $(M, w) \Vdash \neg(\chi \wedge A\chi)$ *and* $(M, u) \Vdash \chi \wedge A\chi$ *and* $w \sim u$

A world u will be at least as plausible as a world w, if and only if they already are of that order and u satisfies χ, or they already are of that order and w satisfies $\neg\chi$ or they are comparable, w satisfies $\neg\chi$ and u satisfies χ. This operation preserves the properties of the plausibility relation and hence preserves plausibility models.

6 Summary and further work

From some existing tools of dynamic epistemic logic, we have argued that experience plays a key role in abductive reasoning. We have extended the language of a non-omniscient epistemic logic with some components that collect some intuitions of deliberation about what explanations come to a surprising fact and which one is the priority. In addition, our proposal emphasizes more than other perspectives, the role of experience, understood as the prior information the agent holds. We have also added a contextual component to the abduction. To order the explanations, we have taken into account the practicality and efficiency of verification.

As possible extensions of our proposal, we consider the fact that abductive problems so far have been limited to phenomena and not epistemic problems. It would be interesting to study how some basic intuitions about abduction would be applicable to problems of the type $K\varphi \wedge \neg\varphi$. *I know φ, however $\neg\varphi$ is the case.* Another interesting research line is to consider a multi-agent system, where concepts such as common knowledge or distributed knowledge, as well as public and private announcements, are taken into account. This would be interesting to study under an experience approach.

References

[1] ALISEDA, A. Abduction as Epistemic Change: A Peircean Model in Artificial Intelligence, *P. A. Flach y A. C. Kakas, eds., Abduction and Induction, Kluwer, Dordrecht, 2000*, pages 45-58

[2] A. BALTAG and S. SMETS. A qualitative theory of dynamic interactive belief revision. *In G. Bonanno, W. van der Hoek, and M. Wooldridge, editors, Logic and the Foundations of Game and Decision Theory (LOFT7), volume 3 of Texts in Logic and Games,* pages 13 60.

[3] A. Baltag, L. S. Moss, and S. Solecki. The logic of public announcements, common knowledge and private suspicions. *Technical Report SEN-R9922, CWI, Amsterdam, 1999.*

[4] BOUTILLIER, C. BECHER, V. Abduction as belief revision, *Artificial Intelligence 77, 1, 1995* pages 43-94

[5] CHEVALEYRE, Y. ENDRISS,U. LANG,J., MAUDET,N. A Short Introduction to Computational Social Choice, *SOFSEM 2007: Theory and Practice of Computer Science, Springer*, pages 51-69

[6] HOFFMANN, M. Hay una lgica de la abduccin?, *Analoga Filosfica, M©xico 12/1 (1998)*, pages 41-56

[7] KAKAS, A.C.; KOWALSKI, R.A.; TONI, F. (1993). Abductive Logic Programming. *Journal of Logic and Computation Vol. II,6. 1992* pages 719“770

[8] KUIPERS, T. Abduction aiming at empirical progress of even truth aproximation leading to a challenge for computational modelling, *Foundations of Science, Vol. IV, 1999.* pages 307-323

[9] LIPTON, P. Inference to the best explanation *W.H. Newton-Smith (eds.) A Companion to the Philosophy of Science (Blackwell, 2000)* pages 184-193.

[10] NEPOMUCENO, ., SOLER-TOSCANO, F., VELZQUEZ-QUESADA, F. An epistemic and dynamic approach to abductive reasoning: selecting the best explanation *Logic Journal of the IGPL, Vol. XXI,6, 2013* pages 943-961

[11] SOLER TOSCANO, F. Razonamiento abductivo en lgica clsica, Cuadernos de Lgica, epistemologa y lenguaje, *College Publications, 2012*

[12] SOLER-TOSCANO, F., VELZQUEZ-QUESADA, F. Generation and Selection of Abductive Explanations for Non-Omniscient Agents, *Journal of Logic, Language and Information, Vol. XXIII, 2, 2014*, pages 141-168

[13] VELZQUEZ-QUESADA, F., SOLER-TOSCANO, F., NEPOMUCENO, . An epistemic and dynamic approach to abductive reasoning: Abductive problem and abductive solution, *Journal of Applied Logic, Vol. XI, 4, 2013*, pages 505“522

[14] VELZQUEZ-QUESADA, F. Dynamic Epistemic Logic for Implicit and Explicit Beliefs, *Journal of Logic, Language and Information vol. XXIII,2* pages 107-140, 2014

LCC-PROGRAM TRANSFORMERS THROUGH BRZOZOWSKI'S EQUATIONS

Enrique Sarrión-Morillo. Universidad de Sevilla, Andalucía Tech.
Dpto. Filosofía y Lógica y Filosofía de la Ciencia. esarrion@us.es

Abstract

The original work about Logic of Communication and Change uses Kleene's translation from finite automata to regular expressions in program transformers. It appears in the axioms set that accomplish to reduce LCC to PDL. This work presents an elegant matrix treatment of Brzozowski's equational method for program transformers. The two alternatives generate equivalent formulas although the obtained ones through Brzozowski's method are usually much smaller; moreover this method possess computational advantages because its complexity in typical cases (but not the worst) is polinomial, whereas in the original LCC paper is always exponential.

Keywords: Brzozowski's equational method, propositional dynamic logic, logic of communication and change, program transformer.

1 Introduction

The *Logic of Communication and Change* (LCC) is a powerful logic system [10] consisting of a Propositional Dynamic Logic [6] (PDL) interpreted epistemically and the action models machinery [3, 2] for representing the knowledge about actions, allows to model diverse epistemic actions and also factual changes.

Such as other logical frameworks, LCC formulas are interpreted over epistemic models: an *epistemic model* M is a triple $(W, \langle R_a \rangle_{a \in \mathsf{Ag}}, V)$ where $W \neq \emptyset$ is a set of worlds, $R_a \subseteq (W \times W)$ is an epistemic relation for each agent $a \in \mathsf{Ag}$ and $V : \mathsf{Var} \to \wp(W)$ is an atomic evaluation[1].

Meanwhile, the action models (relational structures too), are used for representing the knowledge about actions in the system: if $\mathcal{L}$ be a language built upon Var and Ag that can be interpreted over epistemic models, then an $\mathcal{L}$ *action model*[2]

[1]From now on, Ag is a finite set of agents and Var is a set of propositional variables.

[2]The language $\mathcal{L}$ is just a parameter.

U is a tuple $(\mathsf{E}, \langle \mathsf{R}_a \rangle_{a \in \mathsf{Ag}}, \mathsf{pre}, \mathsf{sub})$ where $\mathsf{E} = \{\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}\}$ is a *finite* set of actions, $\mathsf{R}_a \subseteq (\mathsf{E} \times \mathsf{E})$ is a relation for each $a \in \mathsf{Ag}$, $\mathsf{pre} : \mathsf{E} \to \mathcal{L}$ is a precondition map assigning a formula $\mathsf{pre}(\mathsf{e}) \in \mathcal{L}$ to each action $\mathsf{e} \in \mathsf{E}$, and $\mathsf{sub} : (\mathsf{E} \times \mathsf{Var}) \to \mathcal{L}$ is a postcondition map assigning a formula $\mathsf{sub}(\mathsf{e}, p) \in \mathcal{L}$ to each atom $p \in \mathsf{Var}$ at each action $\mathsf{e} \in \mathsf{E}$. With respect to the postcondition map, it is required that $\mathsf{sub}(\mathsf{e}, p) \neq p$ only for a finite number of atoms p. From now on, all action models are assumed to be $\mathcal{L}_{\mathsf{LCC}}$ action models.

In order to obtain the $\mathcal{L}_{\mathsf{LCC}}$ language, formulas and programs, respectively, are defined simultaneously with the notion of an $\mathcal{L}_{\mathsf{LCC}}$ action model (i.e. an action model using $\mathcal{L}_{\mathsf{LCC}}$ for its precondition and postcondition functions):

$$\begin{array}{rcl} \varphi & ::= & \top \mid p \mid \neg\varphi \mid \varphi_1 \wedge \varphi_2 \mid [\pi]\varphi \mid [\mathsf{U}, \mathsf{e}]\varphi \\ \pi & ::= & a \mid ?\varphi \mid \pi_1; \pi_2 \mid \pi_1 \cup \pi_2 \mid \pi^* \end{array}$$

where $p \in \mathsf{Var}$, $a \in \mathsf{Ag}$, U is an $\mathcal{L}_{\mathsf{LCC}}$ action model and e an action in this model.

We establish semantics of $\mathcal{L}_{\mathsf{LCC}}$ throught $||\cdot||^M$ function, that collects the worlds of a given epistemic model M in which a given $\mathcal{L}_{\mathsf{LCC}}$ formula holds or the the pairs of worlds related by a given $\mathcal{L}_{\mathsf{LCC}}$ program: let $M = (W, \langle R_a \rangle_{a \in \mathsf{Ag}}, V)$ be an epistemic model and $\mathsf{U} = (\mathsf{E}, \langle \mathsf{R}_a \rangle_{a \in \mathsf{Ag}}, \mathsf{pre}, \mathsf{sub})$ an action model. The function $||\cdot||^M$, returning both those worlds in W in which an $\mathcal{L}_{\mathsf{LCC}}$ formula holds and those pairs in $W \times W$ in which an $\mathcal{L}_{\mathsf{LCC}}$ program holds, is given by

$$\begin{array}{rlrl} ||\top||^M & := W & ||a||^M & := R_a \\ ||p||^M & := V(p) & ||?\varphi||^M & := \mathsf{Id}_{||\varphi||^M} \\ ||\neg\varphi||^M & := W \setminus ||\varphi||^M & ||\pi_1; \pi_2||^M & := ||\pi_1||^M \circ ||\pi_2||^M \\ ||\varphi_1 \wedge \varphi_2||^M & := ||\varphi_1||^M \cap ||\varphi_2||^M & ||\pi_1 \cup \pi_2||^M & := ||\pi_1||^M \cup ||\pi_2||^M \\ ||[\pi]\varphi||^M & := \{w \in W \mid \forall v((w, v) \in ||\pi||^M \Rightarrow v \in ||\varphi||^M)\} & ||\pi^*||^M & := (||\pi||^M)^* \\ ||[\mathsf{U}, \mathsf{e}]\varphi||^M & := \{w \in W \mid w \in ||\mathsf{pre}(\mathsf{e})||^M \Rightarrow (w, \mathsf{e}) \in ||\varphi||^{M \otimes \mathsf{U}}\} & & \end{array}$$

where $\circ$ and * are the composition and the reflexive transitive closure operator, respectively. Notice two special cases for *test*: $||?\bot||^M = \emptyset$ and $||?\top||^M = \mathsf{Id}_W$.

A key feature of this logic is that it characterises the effect of an action model's execution via *reduction axioms*: valid formulas through which it is possible to rewrite a formula with update modalities as an equivalent one without them, thus reducing LCC to PDL and hence providing a compositional analysis for a wide range of informational events. In order to obtain an correct and complete axiom system, is necessary to introduce the program transformer functions: let $\mathsf{U} = (\mathsf{E}, \langle \mathsf{R}_a \rangle_{a \in \mathsf{Ag}}, \mathsf{pre}, \mathsf{sub})$ be an action model with $\mathsf{E} = \{\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}\}$. The *program transformer* T^{U}_{ij} $(i, j \in \{0, \ldots, n-1\})$ on the set of LCC programs is defined as:

- $T^{\mathsf{U}}_{ij}(a) := \begin{cases} ?\mathsf{pre}(e_i); a & \text{if } \mathsf{e}_i \mathsf{R}_a \mathsf{e}_j \\ ?\bot & \text{otherwise} \end{cases}$
- $T^{\mathsf{U}}_{ij}(?\varphi) := \begin{cases} ?(\mathsf{pre}(\mathsf{e}_i) \wedge [\mathsf{U}, \mathsf{e}_i]\varphi) & \text{if } i = j \\ ?\bot & \text{otherwise} \end{cases}$
- $T^{\mathsf{U}}_{ij}(\pi_1; \pi_2) := \bigcup_{k=0}^{n-1} (T^{\mathsf{U}}_{ik}(\pi_1); T^{\mathsf{U}}_{kj}(\pi_2))$
- $T^{\mathsf{U}}_{ij}(\pi_1 \cup \pi_2) := T^{\mathsf{U}}_{ij}(\pi_1) \cup T^{\mathsf{U}}_{ij}(\pi_2)$
- $T^{\mathsf{U}}_{ij}(\pi^*) := K^{\mathsf{U}}_{ijn}(\pi)$

with K^{U}_{ijn} inductively defined as indicated below:

$$\blacktriangleright\ K^{\mathsf{U}}_{ij0}(\pi) := \begin{cases} ?\top \cup T^{\mathsf{U}}_{ij}(\pi) & \text{if } i = j \\ T^{\mathsf{U}}_{ij}(\pi) & \text{otherwise} \end{cases}$$

$$\blacktriangleright\ K^{\mathsf{U}}_{ij(k+1)}(\pi) = \begin{cases} (K^{\mathsf{U}}_{kkk}(\pi))^* & \text{if } i = k = j \\ (K^{\mathsf{U}}_{kkk}(\pi))^*; K^{\mathsf{U}}_{kjk}(\pi) & \text{if } i = k \neq j \\ K^{\mathsf{U}}_{ikk}(\pi); (K^{\mathsf{U}}_{kkk}(\pi))^* & \text{if } i \neq k = j \\ K^{\mathsf{U}}_{ijk}(\pi) \cup (K^{\mathsf{U}}_{ikk}(\pi); (K^{\mathsf{U}}_{kkk}(\pi))^*; K^{\mathsf{U}}_{kjk}(\pi)) & \text{if } i \neq k \neq j \end{cases}$$

The axiom system for LCC, combines the known axiom system of its PDL fragment ([6]) with *recursion* axioms for its action model fragment:

(*taut*) propositional tautologies

(K) $[\pi](\varphi_1 \to \varphi_2) \to ([\pi]\varphi_1 \to [\pi]\varphi_2)$

(*test*) $[?\varphi_1]\varphi_2 \leftrightarrow (\varphi_1 \to \varphi_2)$

(*seq*) $[\pi_1;\pi_2]\varphi \leftrightarrow [\pi_1][\pi_2]\varphi$

(*choice*) $[\pi_1 \cup \pi_2]\varphi \leftrightarrow [\pi_1]\varphi \wedge [\pi_2]\varphi$

(*mix*) $[\pi^*]\varphi \leftrightarrow \varphi \wedge [\pi][\pi^*]\varphi$

(*ind*) $\varphi \wedge [\pi^*](\varphi \to [\pi]\varphi)) \to [\pi^*]\varphi$

(*MP*) From $\vdash \varphi_1$ and $\vdash \varphi_1 \to \varphi_2$ infer $\vdash \varphi_2$

(*top*) $[\mathsf{U},\mathsf{e}]\top \leftrightarrow \top$

(*atm*) $[\mathsf{U},\mathsf{e}]p \leftrightarrow (\mathsf{pre}(\mathsf{e}) \to \mathsf{sub}(\mathsf{e}, p))$

(*neg*) $[\mathsf{U},\mathsf{e}]\neg\varphi \leftrightarrow (\mathsf{pre}(\mathsf{e}) \to \neg[\mathsf{U},\mathsf{e}]\varphi)$

(*conj*) $[\mathsf{U},\mathsf{e}](\varphi_1 \wedge \varphi_2) \leftrightarrow ([\mathsf{U},\mathsf{e}]\varphi_1 \wedge [\mathsf{U},\mathsf{e}]\varphi_2)$

(*prog*) $[\mathsf{U},\mathsf{e}_i][\pi]\varphi \leftrightarrow \bigwedge_{j=0}^{n-1}[T^{\mathsf{U}}_{ij}(\pi)][\mathsf{U},\mathsf{e}_j]\varphi$

(N_{U}) From $\vdash \varphi$ infer $\vdash [\mathsf{U},\mathsf{e}]\varphi$

(N_π) From $\vdash \varphi$ infer $\vdash [\pi]\varphi$

The crucial reduction axiom is the one characterising the effect of an action model over epistemic PDL programs (*prog*). It is based on the correspondence between action models and finite automata observed in [9]; its main component, the program transformer function T^{U}_{ij}, follows Kleene's translation from finite automata to regular expressions [7]. The present work proposes an alternative definition that uses a matrix treatment of Brzozowski's equational method for obtaining an expression representing the language accepted by a given finite automaton [4, 5]. This alternative definition posses several advantages: first, it have a lower complexity in typical cases (i.e., when the relation differs sufficiently of the Cartesian product), thus allowing more efficient implementations of any LCC-based method; second, the formulas which are obtained are usually much smaller (in the original LCC paper, the size of the transformed formulas of type π^* is always exponential); and third, the matrix treatment presented here is more synthetic, simple and elegant, thus allowing a simpler implementation.

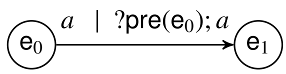
This paper is organized as follows: after this Introduction, Section 2 explains how we can obtain the corresponding expression to each program transformer's type, particularly for Kleene closure through Brzozowski's equational method. Then Section 3 introduces this paper's matrix proposal and discusses the complexity of it. Finally, Section 4 indicates briefly some conclusions.

2 Program transformers through Brzozowski's equations

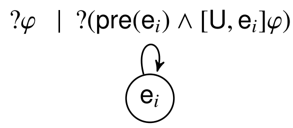
The new definition of program transformer differs mainly, but not only, on the case for the Kleene closure operator. For every program π a matrix $\mu^{\mathsf{U}}(\pi)$, whose cells are LCC programs, is defined. In this matrix, $\mu^{\mathsf{U}}(\pi)[i, j]$ (the cell in the i^{th} row and j^{th} column) corresponds to the transformation (i.e. the path in M) of π from e_i to e_j (i.e. the path in $M \otimes \mathsf{U}$). The matrix $\mu^{\mathsf{U}}(\pi)$ can be interpreted as the adjacency matrix of a labelled directed graph whose nodes are the actions in E and each edge from e_i to e_j is labelled with the transformation of π from e_i to e_j.

Before presenting the formal definitions, we introduce our method in an informal way. In the following paragraphs we introduce examples of action models and label the edges both with an LCC program and its transformation. Labels have two parts separated by a vertical bar, the left part is the LCC program and the right part is its transformation. For example, the label $a \mid ?\mathsf{pre}(\mathsf{e}_0); a$ from e_0 to e_1 in the agents' diagram indicates that $\mu^{\mathsf{U}}(a)[0, 1] = ?\mathsf{pre}(\mathsf{e}_0); a$.

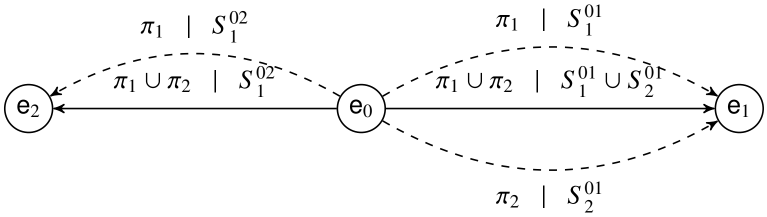
Agents Suppose that U contains an edge from e_i to e_j labelled with agent a. Let w be an state in an epistemic model M. What do we have to test in order to ensure that, after executing $(\mathsf{U}, \mathsf{e}_i)$ over (M, w), an a-path from (w, e_i) to some state (w', e_j) will persist $M \otimes \mathsf{U}$? First, we need to test in M that e_i is executable in w, an then that an a-path exists from w to w'. Trivially, if there is no a-path from e_i to e_j in U, then the transformation of a is $?\bot$, that is, $\mu^{\mathsf{U}}(a)[i, j] = ?\bot$.



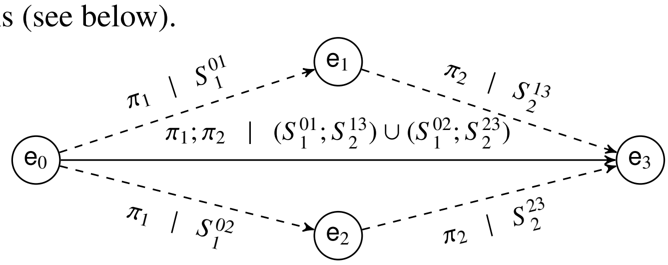
Test The transformation of a test from some e_i to itself is just a test. But the test has two parts, because in order to test $?\varphi$ in (w, e_i) we should first test that $\mathsf{pre}(\mathsf{e}_i)$ is true in w. Then, as the execution of action e_i may change the valuation function, what we test in w is not just φ but rather $[\mathsf{U}, \mathsf{e}_i]\varphi$. The transformation of a test from some state to a different one is always $?\bot$.



Non-deterministic choice The transformation of the choice $\pi_1 \cup \pi_2$ is the choice of the transformations of both programs π_1 and π_2. But, as the choice of some program π with $?\bot$ is equivalent to π, we can simplify some cases. In the diagram, the transformations of some π_1 and π_2 are labelled with dashed lines. When such a line between two nodes does not exist, the transformation should be $?\bot$. Labels of the form S_i^{jk} represent LCC programs corresponding to the transformation of π_i from e_j to e_k. The transformations of $\pi_1 \cup \pi_2$ are shown with continuous lines.

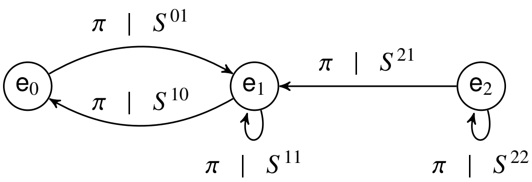


Sequential composition State e_k is reachable from e_i through the concatenation of π_1 and π_2 iff there is some intermediate state e_j that is reachable from e_i through a π_1-path and there is a π_2-path from e_j to e_k. But there can be different intermediate states with these properties (e_1 and e_2 in the example below). So the transformation of the concatenation $\pi_1;\pi_2$ is the choice of all the different possible concatenations (see below).



Up to now, we proceed in a very similar way to that of original program transformer definition, just simplifying some trivial cases like $\pi \cup ?\bot$, which is reduced to π. The main novelty of our transformation is with Kleene closure. We use a method proposed by Brzozowski [4], presented here in a matrix format.

Kleene closure The following graph will be used to illustrate the creation of the transformations of π^* given those of π. This is where our program transformers are substantially different from those in [10].



In the above graph, labels S^{ij} represent the transformations of π from e_i to e_j (when there is no arrow between two —equal or different— nodes, it is assumed that the corresponding program is $?\bot$). In order to find the labels X^{ij} for the transformations of π^*, we follow an equational method first proposed by Brzozowski [4]. Observe, for example, how a π^*-path from e_1 to e_0 might start with S^{10} (an instance of π from e_1 to e_0) and then continue with X^{00} (an instance of π^* from e_0 to e_0), but it might also start with S^{11} (an instance of π from e_1 to e_1) and then continue with X^{10} (an instance of π^* from e_1 to e_0). In this case, these are the only two possibilities, and they can be represented by the following equation:

$$X^{10} = (S^{10};X^{00}) \cup (S^{11};X^{10}) \tag{1}$$

The equations for X^{00} and X^{20} can be obtained in a similar way:

$$X^{00} = ?\mathsf{pre}(\mathsf{e}_0) \cup (S^{01}; X^{10}) \tag{2}$$

$$X^{20} = (S^{22}; X^{20}) \cup (S^{21}; X^{10}) \tag{3}$$

This yields an equation system of LCC programs with X^{00}, X^{10} and X^{20} as its only variables. Observe how, in (2), $?\mathsf{pre}(\mathsf{e}_0)$ indicates that a possible π^*-path from e_0 to e_0 is to do nothing, but the transformation should check whether e_0 is executable at the target state; hence the test $?\mathsf{pre}(\mathsf{e}_0)$.

To solve the above system we proceed by substitution using properties of Kleene algebra [8], such as associative and distributive properties of the operators. First, we can use (2) to replace X^{00} in (1):

$$\begin{aligned}&X^{10} = (S^{10}; (?\mathsf{pre}(\mathsf{e}_0) \cup (S^{01}; X^{10}))) \cup (S^{11}; X^{10}) = (S^{10}; ?\mathsf{pre}(\mathsf{e}_0)) \cup \\ &\cup (S^{10}; S^{01}; X^{10}) \cup (S^{11}; X^{10}) = (S^{10}; ?\mathsf{pre}(\mathsf{e}_0)) \cup (((S^{10}; S^{01}) \cup S^{11}); X^{10}) = \\ &= ((S^{10}; S^{01}) \cup S^{11})^*; S^{10}; ?\mathsf{pre}(\mathsf{e}_0)\end{aligned} \tag{4}$$

The last equality uses Arden's Theorem [1]: $X = B \cup (A; X)$ implies $X = A^*; B$.

Now, we use (4) to substitute X^{10} in (2):

$$X^{00} = ?\mathsf{pre}(\mathsf{e}_0) \cup (S^{01}; ((S^{10}; S^{01}) \cup S^{11})^*; S^{10}; ?\mathsf{pre}(\mathsf{e}_0)) \tag{5}$$

Finally, we substitute X^{10} in (3) and apply Arden's Theorem to solve X^{20}:

$$\begin{aligned}X^{20} &= (S^{22}; X^{20}) \cup (S^{21}; ((S^{10}; S^{01}) \cup S^{11})^*; S^{10}; ?\mathsf{pre}(\mathsf{e}_0)) = \\ &= (S^{22})^*; S^{21}; ((S^{10}; S^{01}) \cup S^{11})^*; S^{10}; ?\mathsf{pre}(\mathsf{e}_0)\end{aligned} \tag{6}$$

After solving these equations, each X^{00}, X^{10} and X^{20} represents a transformed π^*-path from e_0, e_1 and e_2 to e_0, respectively.

By using a matrix calculus similar to that in chapter 3 of [5] we calculate all X^{ij} in parallel and thus avoid repeating the process for each destination node. This method is more synthetic, clear, elegant and allows a simpler implementation that the original LCC paper. The following section will present the formal definition of the matrix calculus; now, we just introduce the basis of the matrix calculus. The equations used above can be represented in the following matrix:

	e_0	e_1	e_2	e_0	e_1	e_2
e_0	$?\perp$	S^{01}	$?\perp$	$?\mathsf{pre}(\mathsf{e}_0)$	$?\perp$	$?\perp$
e_1	S^{10}	S^{11}	$?\perp$	$?\perp$	$?\mathsf{pre}(\mathsf{e}_1)$	$?\perp$
e_2	$?\perp$	S^{21}	S^{22}	$?\perp$	$?\perp$	$?\mathsf{pre}(\mathsf{e}_2)$

The left part contains the π-paths from one node (row) to another one (column). It is an accessibility matrix for the π-graph above. Call $\mu^{\cup}(\pi)[i, j]$ the cell corresponding to row e_i and column e_j in this left part and $A^{\cup}[i, j]$ the cell with the same position at the right part. Observe that $A^{\cup}[i, j] = ?\mathsf{pre}(\mathsf{e}_i)$ if $i = j$ and $?\perp$ otherwise. We may check that the equations for X^{ij} that we created above looking at the π-graph can be created now by:

$$X^{ij} = (\mu^{\cup}(\pi)[i, 0]; X^{0j}) \cup (\mu^{\cup}(\pi)[i, 1]; X^{1j}) \cup (\mu^{\cup}(\pi)[i, 2]; X^{2j}) \cup A^{\cup}[i, j] \tag{7}$$

The greatest advantage of working with matrices is that we can transform several equations at the same time by working in a row. Look at the following matrix, which is the result of applying Arden's Theorem to the e_1 row.

	e_0	e_1	e_2	e_0	e_1	e_2
e_0	$?\bot$	S^{01}	$?\bot$	$?\mathsf{pre}(\mathsf{e}_0)$	$?\bot$	$?\bot$
e_1	$(S^{11})^*; S^{10}$	$?\bot$	$?\bot$	$?\bot$	$(S^{11})^*; ?\mathsf{pre}(\mathsf{e}_1)$	$?\bot$
e_2	$?\bot$	S^{21}	S^{22}	$?\bot$	$?\bot$	$?\mathsf{pre}(\mathsf{e}_2)$

The transformation consists on replacing the position $[\mathsf{e}_1, \mathsf{e}_1]$ in the left part with $?\bot$ and concatenate $(S^{11})^*$ with the others cells in the row.

Not only Arden's Theorem, but also the substitution operation can be done in parallel using matrix operations. Look at the following matrix:

	e_0	e_1	e_2	e_0	e_1	e_2
e_0	$?\bot$	S^{01}	$?\bot$	$?\mathsf{pre}(\mathsf{e}_0)$	$?\bot$	$?\bot$
e_1	$(S^{11})^*; S^{10}$	$?\bot$	$?\bot$	$?\bot$	$(S^{11})^*; ?\mathsf{pre}(\mathsf{e}_1)$	$?\bot$
e_2	$(S^{21}; (S^{11})^*; S^{10})$	$?\bot$	S^{22}	$?\bot$	$(S^{21}; (S^{11})^*; ?\mathsf{pre}(\mathsf{e}_1))$	$?\mathsf{pre}(\mathsf{e}_2)$

The above matrix has been obtained from the previous one by applying one substitution that has converted the left $[\mathsf{e}_2, \mathsf{e}_1]$ position into $?\bot$ and each (left/right) position $[\mathsf{e}_2, \mathsf{e}_i]$ (different to the left $[\mathsf{e}_2, \mathsf{e}_1]$ position) contains now a program with the form $(B; C) \cup D$, where B is always the previous program in the left $[\mathsf{e}_2, \mathsf{e}_1]$ position (always S^{21}, in this case), C is the program in the (resp. left/right) $[\mathsf{e}_1, \mathsf{e}_i]$ position (that is, the program in the same column and the above row) and D is the previous program in the position being modified.

3 A matrix calculus for program transformation

In this section we introduce the formal definitions of our matrix calculus.

Definition 3.1 (Program transformation matrix)**.** Let $\mathsf{U} = (\mathsf{E}, \mathsf{R}, \mathsf{pre}, \mathsf{sub})$ be an action model with $\mathsf{E} = \{\mathsf{e}_0, \ldots, \mathsf{e}_{n-1}\}$. The function $\mu^{\mathsf{U}} : \Pi \to \mathcal{M}_{n\times n}$, with Π the set of LCC programs and $\mathcal{M}_{n\times n}$ the class of n-square matrices, takes an LCC program π and returns a n-square matrix $\mu^{\mathsf{U}}(\pi)$ in which each cell $\mu^{\mathsf{U}}(\pi)[i, j]$ is an LCC program representing the transformation of π from e_i to e_j in the sense of the program transformers $T^{\mathsf{U}}_{ij}(\pi)$ of [10]. The recursive definition of $\mu^{\mathsf{U}}(\pi)$ is:

- *Agents*:
$$\mu^{\mathsf{U}}(a)[i, j] := \begin{cases} ?\mathsf{pre}(\mathsf{e}_i); a & \text{if } \mathsf{e}_i \mathsf{R}_a \mathsf{e}_j \\ ?\bot & \text{otherwise} \end{cases} \tag{8}$$
- *Test*:
$$\mu^{\mathsf{U}}(?\varphi)[i, j] := \begin{cases} ?(\mathsf{pre}(\mathsf{e}_i) \wedge [\mathsf{U}, \mathsf{e}_i]\varphi) & \text{if } i = j \\ ?\bot & \text{otherwise} \end{cases} \tag{9}$$
- *Non-deterministic choice*:
$$\mu^{\mathsf{U}}(\pi_1 \cup \pi_2)[i, j] := \bigoplus \left\{\mu^{\mathsf{U}}(\pi_1)[i, j],\ \mu^{\mathsf{U}}(\pi_2)[i, j]\right\} \tag{10}$$
where $\bigoplus\Gamma$ is the non-deterministic choice of the programs in Γ set after removing occurrences of $?\bot$, that is (being "$\bigcup$" the generalised non-deterministic choice ("$\cup$") of a program non-empty set),

$$\bigoplus\Gamma := \begin{cases} \bigcup(\Gamma \setminus \{?\bot\}) & \text{if } \varnothing \neq \Gamma \neq \{?\bot\} \\ ?\bot & \text{otherwise} \end{cases} \tag{11}$$

- *Sequential composition*:

  $$\mu^{\mathsf{U}}(\pi_1;\pi_2)[i,j] := \bigoplus\left\{\mu^{\mathsf{U}}(\pi_1)[i,k] \odot \mu^{\mathsf{U}}(\pi_2)[k,j] \;\middle|\; 0 \leq k \leq n-1\right\} \tag{12}$$

  where $\sigma \odot \rho$ is the sequential composition of σ and ρ after removing superfluous occurrences of $?\bot$ and $?\top$, that is,

  $$\sigma \odot \rho := \begin{cases} \sigma;\rho & \text{if } \sigma \neq ?\bot \neq \rho \text{ and } \sigma \neq ?\top \neq \rho \\ \sigma & \text{if } \sigma \neq ?\top = \rho \\ \rho & \text{if } \sigma = ?\top \\ ?\bot & \text{otherwise} \end{cases} \tag{13}$$

- *Kleene closure*:

  $$\mu^{\mathsf{U}}(\pi^*) := S_0^{\mathsf{U}}\left(\mu^{\mathsf{U}}(\pi) \mid A^{\mathsf{U}}\right) \tag{14}$$

  where $\mu^{\mathsf{U}}(\pi) \mid A^{\mathsf{U}}$ is the $n \times 2n$ matrix obtained by augmenting $\mu^{\mathsf{U}}(\pi)$ with A^{U}, an $n \times n$ matrix defined as

  $$A^{\mathsf{U}}[i,j] := \begin{cases} ?\mathsf{pre}(\mathsf{e}_i) & \text{if } i = j \\ ?\bot & \text{otherwise} \end{cases} \tag{15}$$

  The function S_k^{U} (with $0 \leq k \leq n$), defined as

  $$S_k^{\mathsf{U}}(M \mid A) := \begin{cases} A & \text{if } k = n \\ S_{k+1}^{\mathsf{U}}(Subs_k(Ard_k(M \mid A))) & \text{otherwise} \end{cases} \tag{16}$$

  receives an argument $M \mid A$ and performs an iterative process applying Arden's Theorem to row k (via function $Ard_k : \mathcal{M}_{n\times 2n} \to \mathcal{M}_{n\times 2n}$) and substituting rows different from k (via function $Subs_k : \mathcal{M}_{n\times 2n} \to \mathcal{M}_{n\times 2n}$) until a $k = n$, then returning the right part of the augmented matrix. The two auxiliary functions, Ard_k and $Subs_k$, are given by

  $$Ard_k(N)[i,j] \;:=\; \begin{cases} N[i,j] & \text{if } i \neq k \\ ?\bot & \text{if } i = k = j \\ N[i,j] & \text{if } i = k \neq j \text{ and } N[k,k] = ?\bot \\ N[k,k]^* \odot N[i,j] & \text{otherwise} \end{cases} \tag{17}$$

  $$Subs_k(N)[i,j] \;:=\; \begin{cases} N[i,j] & \text{if } i = k \\ ?\bot & \text{if } i \neq k = j \\ \bigoplus\{N[i,k] \odot N[k,j], N[i,j]\} & \text{otherwise} \end{cases} \tag{18}$$

Now it is possible to substitute the previous version of the crucial reduction axiom by the following: $[\mathsf{U},\mathsf{e}_i][\pi]\varphi \;\leftrightarrow\; \bigwedge_{\substack{0\leq j\leq n-1 \\ \mu^{\mathsf{U}}(\pi)[i,j]\neq ?\bot}} [\mu^{\mathsf{U}}(\pi)[i,j]][\mathsf{U},\mathsf{e}_j]\varphi$

3.1 Complexity

The program transformers in [10] require exponential time due to the use of Kleene's method [7]; moreover, the size of the transformed formulas of type π^* is also exponential because of the definition of $K^{\cup}_{ijn}$. The advantage of our transformers is that they can be executed in polynomial time in cases different to the worst one[3]; moreover, in typical cases our method generates much smaller expressions.

Let $M = (W, \langle R_a \rangle_{a \in \mathsf{Ag}}, V)$ be an epistemic model with $card(W) = n$. If M is a complete model, then the number of operators in $\mu^{\cup}(\pi^*)[n-1, 0]$ is in the order of 2^{2n} (i.e. our transformers produce an exponential output), which implies that the required time is also exponential. If M is a *chain model*[4], then the number of operators in $\mu^{\cup}(\pi^*)[n-1, 0]$ is in the order of $2n^2$ (i.e. in this case the length of the output is polynomial), thus the required time is also polinomial.

An analysis of the operations involved in computing $\mu^{\cup}(\pi)$ for the different kinds of programs π show that, while Kleene's method forces the program transformers to use an exponential number of operations, our proposal uses only a polynomial number of matrix operations: if π is an agent, a test or a non-deterministic choice, then the whole matrix $\mu^{\cup}(\pi)$ can be computed in $O(n^2)$; but, if π is a sequential composition or a Kleene closure, then $\mu^{\cup}(\pi)$ is computed in $O(n^3)$.

Just, the last is the crucial case; therefore it is analyzed with more detail. With $\mu^{\cup}(\pi)$ given, computing $\mu^{\cup}(\pi^*)$ requires first to build $(\mu^{\cup}(\pi) \mid A^{\cup})$ and then n iterations of $S^{\cup}_k$ (see (14)). Note that the size of $(\mu^{\cup}(\pi) \mid A^{\cup})$ is $n \times 2n$ and to build each cell requires a constant number of operations. So building the initial matrix is in $O(n^2)$. Now, each one of the n calls to $S^{\cup}_k$ is in $O(n^2)$, as Ard_k (see (17)) only changes the cells in row k, $Subs_k$ only the cells in the other rows, and each cell can be modified in constant time. So the n calls to $S^{\cup}_k$ are computed in $O(n^3)$. If the matrices for subprograms are not given and g is the number of subprograms in π, building $\mu^{\cup}(\pi)$ from scratch requires a number of matrix operations in $O(g \cdot n^3)$.

3.2 Possible improvements

The operators "$\oplus$" and "$\odot$" used in the previous definition are versions, respectively, of non-deterministic choice and sequential composition that remove unnecessary occurrences of $?\bot$ and $?\top$; thus returning programs that are (potentially) syntactically shorter but nevertheless semantically equivalent to their PDL counterparts "$\cup$" and ";". The $\odot$ operation's definition can be modified to obtain even simpler expressions. For example, $\sigma \odot \rho$ might be defined as σ if $\sigma \neq ?\top = \rho$ and as ρ if $\sigma = ?\top$. Moreover, the algorithm implementing Ard_k and $Subs_k$ functions can be improved by disregarding the $N[i, j]$ elements with $j < k$ or $j > n+k$ (being $N[i, j]$ a $n \times 2n$ matrix), since those are necessarily equal to $?\bot$. These changes,

[3] The worst cases are *complete* models, i.e. fully connected models.

[4] A chain model is a model with a linear order relation.

despite not lowering the translation's complexity order, would nevertheless make it more efficient.

4 Conclusions

This work presents an alternative definition of the program transformers, which are used in the crucial axiom of the axiom system for LCC. This system allows us to reduce LCC to PDL. The proposal uses a matrix treatment of Brzozowski's equational method in order to obtain a regular expression representing the language accepted by a finite automaton. While Brzozowski's method and that used in the original LCC paper [10] are equivalent, the first is computationally more efficient in typical cases and generates much smaller expressions. Moreover, the matrix treatment presented here is more synthetic, clear and elegant, thus allowing a simpler implementation.

References

[1] D. N. Arden. Delayed-logic and finite-state machines. In *SWCT (FOCS)*, pages 133–151. IEEE Computer Society, 1961.

[2] A. Baltag and L. S. Moss. Logics for epistemic programs. *Synthese*, 139(2):165–224, 2004.

[3] A. Baltag, L. S. Moss, and S. Solecki. The logic of public announcements and common knowledge and private suspicions. In I. Gilboa, editor, *TARK*, pages 43–56, San Francisco, CA, USA, 1998. Morgan Kaufmann.

[4] J. A. Brzozowski. Derivatives of regular expressions. *Journal of the ACM*, 11(4):481–494, 1964.

[5] J. H. Conway. *Regular Algebra and Finite Machines*. Chapman and Hall, 1971.

[6] D. Harel, D. Kozen, and J. Tiuryn. *Dynamic Logic*. MIT Press, Cambridge, MA, 2000.

[7] S. Kleene. Representation of events in nerve nets and finite automata. In C. E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–41. Princeton University Press, Princeton, NJ, 1956.

[8] D. Kozen. On kleene algebras and closed semirings. In B. Rovan, editor, *MFCS*, volume 452 of *Lecture Notes in Computer Science*, pages 26–47. Springer, 1990.

[9] J. van Benthem and B. Kooi. Reduction axioms for epistemic actions. In R. Schmidt, I. Pratt-Hartmann, M. Reynolds, and H. Wansing, editors, *Advances in Modal Logic (Number UMCS-04-09-01 in Technical Report Series)*, pages 197–211. Department of Computer Science, University of Manchester, 2004.

[10] J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.

MEREOLOGY AND TEMPORAL STRUCTURES

Pedro González Núñez
Universidad de Sevilla, Andalucía Tech
pegnunez@gmail.com

Abstract

We will propose a semantics for first order multimodal logics combining necessity and temporal operators, which intends to reflect some philosophical insights brought up when discussing certain topics in ontology. Namely, our proposal will allow us to construe a primarily semantic notion of part, and specifically of temporal and spatial part, which will in turn enable an easy and intuitive way of modeling some metaphysical claims, and also set a solid framework to develop multimodal logics including temporal and non-temporal operators.

1 Introduction

How to best conceive change and the notion of the flow of time are questions that have an illustrious lineage in the history of philosophy. In contemporary literature there are quite a handful of intertwined topics that are deployed when discussing conceptions of time in relation with identity and constitution. We will present a model theory [1] that is tailored to let us easily model some of these topics, which include, but are not limited to, the use of the notion of "temporal part" to explain maintenance of identity through change, and some approach to the idea that there is a notion of "loose" or "vague" identity that is used in ordinary speech that should be differentiated from the proper notion of ("strict") identity".

To achieve this, we are going to make some strong assumptions, necessary to increase the manageability of the philosophical notions we will try to formalize. First, we shall adopt the full law of Leibniz –i.e. both the principle of identity of the indiscernibles and the principle of indiscernibility of the identicals– as a given. This amounts to assuming that our language will be expressive enough for the purposes of distinguishing the objects of our theories. Therefore it can be

[1]The basic structure for the treatment of time is heavily based on the so-called Kamp frames, and in a variation on bundled tree structures found in [7]. More systematic work on the relation between Kamp frames and bundled trees can be found in [12].

understood as a way to limit the scope of our task rather than as an unjustified metaphysical claim.

Secondly, as an application of the aforementioned principles we shall understand that, for purpose of our models, it suffices for an object a to be a part of another object b at some point of time and region of space, that they are undistinguishable in that point and region. Although if understood as a philosophical claim it would require an independent argument to be justified, we can take this to be, like the one above, a methodological constraint we impose to ourselves.

Thirdly, we are going to adopt the following semantical stance: the truth of the attribution of some predicate P to some term t at some state w is to be understood in a different way than the standard treatment. Instead, we shall define the denotation of a predicate symbol as an element of some non-empty set, we shall understand a "property" as the relativisation of the denotation of a predicate-symbol (say, P) to a world (say, w), that is, as the pair $\langle ||P||, w \rangle$. Then, we shall understand the denotation of individual terms as sets of "properties" in the mentioned sense, and say that Pt is true in a world if the pair $\langle ||P||, w \rangle$ is an element of the denotation of t. Thus, with these guiding notions of basing parthood in sameness of "properties", and construing individuals as sets of "properties" [2] we will be able to draw semantical notions of parthood in terms of set-theoretical inclusion.

2 Temporal structures: language and semantics

Definition 1 (Language)**.** The formulas of our basic language are defined by the following grammar:

$\phi ::= Pt \mid \neg\phi \mid \phi \wedge \phi \mid \langle F \rangle\phi \mid \langle P \rangle\phi \mid \langle S \rangle\phi \mid \forall x\phi \mid t_1 = t_2$

where t is either an individual constant or variable.

As can be seen, we are limiting ourselves to monadic predicate-letters.

The symbols of the form $\langle * \rangle$ are to be understood as "possibility" operators. $\langle P \rangle$ is to be read as "at some point in the past"; $\langle F \rangle$ is to be read as "at some point of the future"; $\langle S \rangle$ will be given two alternate lectures depending on the kind of discourse we are representing: we can either speak of alternative or counterfactual histories, in which case it will be understood as "at this time in some other history", or we can speak of states of affairs in different regions of space, in which case it will be read as "in some spatial region". We will write $[*]$ for the dual of any modal operator $\langle * \rangle$. We shall also use the usual truth-functional conectives definable by means of negation and conjunction, and the existential quantifier defined by means of the universal quantifier and negation.

[2] Let us maintain the quotation marks as a reminder that here property has a meaning quite different than in usual formal semantics.

Definition 2 (Parallel histories frame)**.** A parallel history frame is a 5-tuple $F = \langle W, \prec, \approx, U, D \rangle$, where

- W is a non-empty set
- $\prec$ is a binary relation on W such that for all $x, y, z, \in W$
  - $\forall xyz(x \prec y \wedge y \prec z \rightarrow x \prec z)$,
  - $\forall xy \neg(x \prec y \wedge y \prec x)$,
  - $\forall xyz(x \prec y \wedge x \prec z \rightarrow (y \prec z \vee y = z \vee z \prec y))$,
  - $\forall xyz(y \prec x \wedge z \prec x \rightarrow (y \prec z \vee y = z \vee z \prec y))$,
- $\approx$ is an equivalence relation such that for all $x, y, z, w, \in W$
  - $\forall xy(x \approx y \rightarrow \neg x \prec y)$,
  - $\forall xyz(x \prec y \wedge x \approx z \rightarrow \exists w(w \approx y \wedge z \prec w))$,
  - $\forall xy(\neg(x \prec y \vee y \prec x \vee x = y) \rightarrow \exists z((x \prec z \vee z \prec x \vee x = z) \wedge z \approx y))$
- U is a non-empty set such that $W \bigcap U = \emptyset$
- D is a non-empty set such that $D \subseteq \wp(U \times W)$

Additionally, we shall say that $w \sim v$, or informally, that w and v belong to the same history, whenever $w \prec v$, $v \prec w$ or $v = w$

Relation $\prec$ is constructed as an earlier-later relation as usual in Kamp frames [3]. The relation $\approx$, however is a variation on the one used in Kamp frames. The motivation for the variation is achieving a notion of necessity which check for all histories in the model rather than only those with a common past up to the point of evaluation, which is the norm in renderings of "historical" necessity.

Definition 3 (Parallel histories model)**.** A parallel histories model is a tuple $M = \langle W, \prec, \approx, U, D, I \rangle$, where $W, U, D, \prec$ and $\approx$ are as above and I is an interpretation function that assigns elements of D to individual constants, and elements of U to predicate symbols. We shall also have an assignment a which assigns elements of D to the individual variables of our language. Note that we do not impose any restrictions on the atoms that are true in worlds related by $\approx$.

Definition 4 (Semantic rules)**.** The rules that govern the semantic interpretation of our language, denoted by $|| * ||$, are as follows:

- For any constant symbol c, $||c||_{M,w,a} = I(c)$

[3] We follow [6], page 664 in our understanding of Kamp structures.

- For any predicate symbol P, $||P||_{M,w,a} = I(P)$
- For any individual variable x, $||x||_{M,w,a} = a(x)$
- For any term t and predicate-symbol P, $||Pt||_{M,w,a} = 1$ iff $\langle ||P||_{M,w,a}, w\rangle \in ||t||_{M,w,a}$
- For any terms, t_1 and t_2, $||t_1 = t_2||_{M,w,a} = 1$ iff $||t_1||_{M,w,a} = ||t_2||_{M,w,a}$

  For any well-formed formulae φ and ψ:
- $M, w, a \models \neg\varphi$ syss $M, w, a \not\models \neg\varphi$
- $M, w, a \models \varphi \wedge \psi$ syss $M, w, a \models \varphi$ and $M, w, a \models \psi$
- For any variable x $M, w, a \models \forall x\varphi$ iff every assignment a' differing from a in at most the value of $a'(x)$ is such that $M, w, a' \models \varphi$
- $M, w, a \models \langle F\rangle\varphi$ iff exists v such that $w \prec v$ and $M, v, a \models \varphi$
- $M, w, a \models \langle P\rangle\varphi$ iff exists v such that $v \prec w$ and $M, v, a \models \varphi$
- $M, w, a \models \langle S\rangle\varphi$ iff exists v such that $v \approx w$ and $M, v, a \models \varphi$

These semantic rules let us wiev, as stated in the introduction, each individual as the set of "properties" which hold true of it. That is why we have limited ourselves to unary predicates, since we cannot set such an straightforward way of construing the interpretation of individual terms as being the set of interpretations of both the monadic and poliadic relations which are true of them, for the latter would involve the interpretation of some other individual terms. Trying to find a relatively homogeneus treatment for poliadic relations is an objective set for future investigations.

With this we have a semantics that enforces both Leibniz principles precisely due to how we have set the semantics for terms, according to our described aim. Beyond that, the changes in atomic sentences interpretation have no further implications in terms of validities. The most salient feature of the temporal structure is that every “history” is enforced to have the same topology, so to speech, so that for every moment of every history, there is an equivalence class of moments that are “simultaneous” to it. That enables another interpretation for the modality $\langle S\rangle$, as said above: instead of having alternative histories, we can view the structure as modeling simultaneous histories, i.e. histories of different regions of space. This is crucial, given that once having the notion of temporal part we can, with some modifications, either use it to represent counterfactual discourse concerning objects and their temporal parts, or discourse about the interaction of objects and both their temporal and spatial parts within the same history. Let’s then define such notions.

3 Mereology: notions of parthood and some applications

Below we shall introduce new primitive relational symbols for parthood relationships. In order to facilitate the formulation of the semantic rules, we shall introduce an auxiliary definition.

Definition 5 (Reduction of an individual to a world)**.** Given an individual $d \in D$ and a world $v \in W$ we shall define the reduction of d to v as

$$red(d, w_1) = \{\langle u, w_i \rangle \in d \mid w_i = w_1\}$$

With this set-theoretical device, our mereological basic relations can be defined as follows.

Definition 6 (Spatial parthood relation)**.** Let us add to our basic language a binary relation symbol with interfixed notation $\sqsubseteq^S$ to denote the notion '...'is an (improper) spatial part of...". The semantic rule for this symbol is as follows:
$M, w, a \models t_1 \sqsubseteq^S t_2$ iff:

$$\cup\{red(||t_1||_{M,w,a}, v)|v \approx w\} \subseteq \cup\{red(||t_2||_{M,w,a}, v)|v \approx w\}, \text{ and } red(||t_1||_{M,w,a}, w) \neq \emptyset$$

This says that some object a is a spatial part of b at some point of space and time w if, provided that a has some property at w, then the reduction of a to the spatio-temporal locations simultaneous with w is a subset of the respective reduction of b. That is, the relation holds whenever the two individuals are indistiguishable from each other at any state simultanous with the point of evaluation.

Definition 7 (Temporal parthood in counterfactual contexts)**.** Let us add to our basic language a binary relation symbol with interfixed notation $\sqsubseteq^{T1}$ to denote the notion "... is an (improper) temporal part of...". The semantic rule for this symbol is as follows:
$M, w, a \models t_1 \sqsubseteq^{T1} t_2$ iff:

$$\cup\{red(||t_1||_{M,w,a}, v)|v \sim w\} \subseteq \cup\{red(||t_2||_{M,w,a}, v)|v \sim w\}, \text{ and } red(||t_1||_{M,w,a}, w) \neq \emptyset$$

This says that some object a is a temporal part (in a couterfactual context) of b at some point of space and time w if, provided that a has some property at w, then the reduction of a to the states in the history to which w belongs is a subset of the respective reduction of b. That is, the relation holds whenever the two individuals are indistiguishable from each other at any state in the same history as the point of evaluation.

Now, it is noteworthy that while the notion of spatial part as defined is intuitive when the frame is interpreted as an spatio-temporal frame, that intuitive interpretation is lost when we switch to the counterfactual histories interpretation of the semantics. In a similar way, while the notion of some object being a temporal part of another in a given history, whereas in some other history it might not be, is perfectly intelligible, the notion of being a temporal part in a region of space while possibly not being so in another defeats the notion of temporal part under that interpretation: a is a temporal part of b at time t if and only if a is indistinguishable with b at that time, everywhere. While the relation of spatial parthood is variable from time to time, the relation of temporal parthood is constant from time to time and from place to place. Thus, an alternative should be construed to represent the notion of temporal parthood when our language and semantics are given the spatio-temporal interpretation.

Definition 8 (Temporal parthood (in spatio-temporal contexts))**.** We yet again extend our language with a second temporal parthood relation, expressed with the symbol $\sqsubseteq^{T2}$, interpreted as follows:
$M, w, a \models t_1 \sqsubseteq^{T2} t_2$ iff:

$$\{\cup\{red(||t_1||_{M,w,a}, v) \mid v \approx w'\} \mid \cup\{red(||t_1||_{M,w,a}, v)\} \neq \emptyset, w' \sim w\} \subseteq$$

$$\subseteq \{\cup\{red(||t_2||_{M,w,a}, v) \mid v \approx w'\} \mid \cup\{red(||t_1||_{M,w,a}, v)\} \neq \emptyset, w' \sim w\},$$

$$\text{and } red(||t_1||_{M,w,a}, w) \neq \emptyset$$

.

This definition is a bit more complex, but not too much. It says that an object a will in this sense be a temporal part of another object b iff, by considering the unions of reductions of a respect to worlds simultanous to some world w', and collecting the set of all such unions such that they are non-empty and w' is in the history the point of evaluation belongs to, and doing the same with respect to b, it happens that the set of such unions of reductions of a is included in the set of said unions of reductions of b. This means that if a exists at some time, then a is indistingueshable from b in any region and time in which some property hods true of a.

It can be shown that these relations are transitive and reflexive, while for the spatial and temporal in confterfactual context parthood relations their modal nature makes antisymmetry fail. (I will subsequently omit superindexes of our parthood symbols when the matter discussed is equally appliable to the three of them, or the reference to one or another is clear from the context). While $a \sqsubseteq^i b \wedge b \sqsubseteq^i b \rightarrow a = b$ when $i \in \{S, T1\}$ is not valid in every parallel histories frame, a modal version of it is so, different for each of the two problematic parthood relationships:

$$[P](a \sqsubseteq^S b \wedge b \sqsubseteq^S a) \wedge [F](a \sqsubseteq^S b \wedge b \sqsubseteq^S a) \rightarrow a = b$$

$$[S](a \sqsubseteq^{T1} b \wedge b \sqsubseteq^{T1} a) \rightarrow a = b$$

In general, it should be observed that adaptation of mereological axions for S- and T1-parthood should replace occurances of identity with local equality or indiscernibility, which can be defined as reciprocal parthood ($a \sqsubseteq b \wedge b \sqsubseteq a$). This can be understood by examinating the usual definion of "proper parthood" which we should symbolise $\sqsubset$. If we adopt the usual definition $x \sqsubset y =_{def} x \sqsubseteq y \wedge \neg(x = y)$, then we are putting forward a needlessly weak condition: In order to, say, recognise that a is a proper spatial part of b at some time, we must demand that they be distinct objects in our model, but also that they are distinguishable at the time of evaluation, that is at some simultaneous state to the point of evaluation, and mutatis mutandis with the $\sqsubset^{T1}$ relation. To achieve this relative inditinguishability, we should require that $x \sqsubset y =_{def} x \sqsubseteq y \wedge \neg(y \sqsubseteq x)$.

Now, this can allow us to fomally model puzzles such as that of the boat of Theseus, or similar ones, as a case of certain individuals being part of others at different times. This construction allow us to formally approach to notions of "loose identity": we can loosely say that two things are the same if they have the same set of parts, as in the puzzle referred above, if the two are parts of the same entity (e.g. me as a baby and me as an undergraduate can be construed as two different individuals, being what identifies them the fact that the former was indistinguishable from me as a whole when I was a baby and the later was indistinguishable from me some years ago, therefore, they are "the same" in a non-strict sense). Similar relations can be construed from qualitative similarities: loose identity based on similarity would be strict identity of some subsets of the individuals. Of course a study of all the different uses or "A and B are identical/the same" is not the object of this work, but may this serve as a lead to possible analyses of such phenomenon.

4 Possible formal developments

As it stands, we have a modal system that basically combines S5 for the $[S]/\langle S \rangle$ operators with the temporal system of choice. Searching for complete classes of models for a given axiomatisation of the temporal aspect and the different relations of parthood –and it is noteworthy that some stipulations on the parthood relations may impose restrictions not only in the composition of the domain of the appropriate models, but also on the onderlying structure $\langle W, R \rangle$. Another very compelling development possibility is refining the structure of the spatial aspect of our models.

Another line of investigation that can be started from this framework is the embedding of counterpart-like semantics in our frames, by interpreting our modal statements not as being about individuals which happen to be identical to those referenced in the scope of the modality, bus as being about individuals which stand in some "counterparthood" relation with them, relation that in principle need not be an equivalence.

References

[1] Benthem, Johan van (2010) – Modal logic for open minds. Stanford, California, CSLI Publications.

[2] Blackburn, P., de Rijke, M., Venema, Y. (2005) – Modal logic, Cambridge university Press.

[3] Burgess, John P. (1979) – "Logic and Time" in The Journal of Symbolic Logic, Vol. 44, No. 4, pp. 566–582

[4] Fitting, M., Mendelsohn, R. L.(1990) – First-order modal logic. Dordrecht, Kluwer Academic.

[5] Hawley, K., – "Temporal Parts", The Stanford Encyclopedia of Philosophy (Winter 2010 Edition), Edward N. Zalta (ed.), `http://plato.stanford.edu/archives/win2010/entries/temporal-parts/`

[6] Hodkinson, I., Reynolds, M. (2007) – Temporal logic. In Handbook of modal logic (Vol. 11). Amsterdam; Boston: Elseiver.

[7] Nute, D. (1991) – "Historical Necessity and Conditionals" Noûs, Vol. 25, No. 2, pp. 161–175

[8] Oderberg, D. S. (2004) – "Temporal Parts and the Possibility of Change", Philosophy and Phe-nomenological Research Vol. LXIX, No. 3

[9] Steen, M. (2008) – "Chisholm's changing conception of ordinary objects" , Grazer Philosophische Studien 76, pp. 1–56.

[10] , "Mereology", The Stanford Encyclopedia of Philosophy (Fall 2014 Edition), Ed-ward N. Zalta (ed.), forthcoming `http://plato.stanford.edu/archives/fall2014/entries/mereology/`

[11] Yourgrau, P. (1985) – "On the Logic of Indeterministic Time", The Journal of Philosophy, Vol. 82, No. 10, Eighty-Second Annual Meeting AmericanPhilosophical Association, Eastern Division, pp. 548–559

[12] Zanardo, A. (1996) - "Branching-Time Logic with Quantification over Branches: The Point of View of Modal Logic", The Journal of Symbolic Logic, Vol. 61, No. 1, pp. 1–39