UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
POSGRADO EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN
INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS
TEORÍA DE LA COMPUTACIÓN

PATTERN MODELS: DYNAMIC EPISTEMIC LOGICS FOR DISTRIBUTED SYSTEMS

TESIS
QUE PARA OPTAR POR EL GRADO DE
DOCTOR EN CIENCIA E INGENIERÍA DE LA COMPUTACIÓN

PRESENTA:
M.C.I.C. DIEGO ALEJANDRO VELÁZQUEZ CERVANTES

TUTORES PRINCIPALES
DR. ARMANDO CASTAÑEDA ROJANO, INSTITUTO DE MATEMÁTICAS, UNAM
DR. DAVID ARTURO ROSENBLUETH LAGUETTE, INSTITUTO DE INVESTIGACIONES EN MATEMÁTICAS APLICADAS Y EN SISTEMAS, UNAM

MIEMBRO DEL COMITÉ TUTOR
DR. HANS PIETER VAN DITMARSCH, CENTRE NATIONAL DE LA RECHERCHE SCIENTIFIQUE, INSTITUT DE LA RECHERCHE EN INFORMATIQUE DE TOULOUSE, UNIVERSITÉ DE TOULOUSE

CIUDAD DE MÉXICO, ABRIL, 2024

PROTESTA UNIVERSITARIA DE INTEGRIDAD Y

HONESTIDAD ACADÉMICA Y PROFESIONAL

De conformidad con lo dispuesto en los artículos 87, fracción V, del Estatuto General, 68, primer párrafo, del Reglamento General de Estudios Universitarios y 26, fracción I, y 35 del Reglamento General de Exámenes, me comprometo en todo tiempo a honrar a ala institución y a cumplir con los principios establecidos en el Código de Ética de la Universidad Nacional Autónoma de México, especialmente con los de integridad y honestidad académica.

De acuerdo con lo anterior, manifiesto que el trabajo escrito titulado Pattern Models: Dynamic Epistemic Logics for Distributed Systems que presenté para obtener el grado de Doctor en Ciencia e Ingeniería de la Computación, es original, de mi autoría y lo realicé con el rigor metodológico exigido por mi Programa de Posgrado, citando las fuentes, ideas, textos, imágenes, gráficos u otro tipo de obras empleadas para su desarrollo.

En consecuencia acepto que la falta de cumplimiento de las disposiciones reglamentarias y normativas de la Universidad, en particular las ya referidas en el Código de Ética, llevará a la nulidad e los actos de carácter académico administrativo del proceso de titulación/graduación

Atentamente

Diego Alejandro Velázquez Cervantes, 305022970

To Edith, Bruno and Mónica.

Acknowledgements

I would like to thank to CONAHCYT for the national fellowship that I received during my PhD, to DGAPA for the financial support from PAPIIT projects IN108720 and IN108723, and all Mexicans that made CONAHCYT and DGAPA founding possible. I thank my wife Edith, my son Bruno and my daughter Mónica to whom this thesis is dedicated for all their sacrifices these years. I thank Dr. Armando Castañeda, Dr. Hans van Ditmarsch, and Dr. David Rosenblueth for all the time spent in supervisor duties. I would like to give a special mention to Hans for the continuous support during all my PhD not only in supervisor duties but also in hosting a research visit in the Netherlands. Finally, I thank Dr. Sergio Rajsbaum and Dra. Lourdes González for being members of the jury in my candidacy exam and this thesis defense.

Contents

Abstract

A dynamic epistemic logic extends epistemic logic with the capability to model epistemic change in the system. This thesis presents two dynamic epsitemic logics for analysing distributed-computing models: pattern model logic and parametrized pattern model logic. The first logic describes communication of agents executing the "full-information protocol" and the second allows describing communication executing arbitrary protocols.

We focus on "dynamic-network models". These models are sets of sequences of "communication graphs", namely directed graphs whose nodes are the agents and whose arrows describe successful communication between agents. We can see these models as "adversaries" that decide who communicates with whom in each round of communication.

We start by studying "action models", a general dynamic epistemic logic describing epistemic change by events. Using action models for describing some adversaries has disadvantages, however. In particular, "oblivious" adversaries, the simplest dynamic-network models, require a different action model for each round of communication whose size grows exponentially in the number of rounds. This motivates us to work on a more adequate approach.

Subsequently, we present pattern model logic, that describes communication executing the "full-information protocol". We propose an axiomatization of the logic. We define a procedure to build an infinite sequence of pattern models that describe an adversary and prove that the updates of the epistemic models with the pattern models in such a sequence preserve the indistinguishability relation between configurations. Also, we compare the "update expressivity" of action models and pattern models concluding that both logics are incomparable, namely action models generate updates in epistemic models that cannot be generated using pattern models and vice versa. For this logic, we present a sufficient condition that ensures preserva-

tion of connectivity of the epistemic models through rounds of executions in oblivious adversaries.

Finally, we modify the formalism adding the capability for describing communication using an arbitrary deterministic protocol. We define a modified version of epistemic models and define the semantics of epistemic logic in these structures.

This work gathers results from: "Communication pattern models: An extension of action models for dynamic-network distributed systems" and "Comparing the Update Expressivity of Communication Patterns and Action Models" conference papers, and "Communication Pattern Logic: Epistemic and Topological Views" and "Pattern Models: A Dynamic Epistemic Logic for Distributed Systems" journal papers.

Chapter 1

Introduction

Context. In a distributed system, communication is typically performed either by sending and receiving messages, or by writing to, and reading from, a shared memory. The communication patterns (i.e., who communicated with whom) that can occur may change from model to model. When designing and analyzing distributed systems, it is often the case that authors informally refer to what an agent "knows" after an agent performs some action. Halpern and Moses [24] took the first steps towards establishing a formal connection between distributed systems and epistemic logic in 1984. Roughly, a distributed protocol is studied through an *epistemic model* with each of its states representing a possible *configuration* of the protocol. The epistemic-based approach to distributed systems has been fruitful, as shown in the book by Fagin, Halpern, Moses, and Vardi [20].

An important connection between distributed computing and topology was exhibited in three independent papers by Borowsky and Gafni [10], Herlihy and Shavit [27], and Sacks and Zaharoglou [51] in 1993, and since then this approach has provided useful techniques to show a number of important results in this field. The book by Herlihy, Kozlov, and Rajsbaum [26] provides a comprehensive description of this connection.

Recently, Goubault, Ledent, and Rajsbaum have shown [22] that the epistemic-based approach can be directly connected to the topology-based approach to distributed systems. The topological approach studies a distributed protocol through its topological representation: a geometric object, called *simplicial complex*, where each of its faces is associated with a *configuration* of the protocol. In essence, Goubault, Ledent, and Rajsbaum established [22] a correspondence between the topological description of dis-

tributed protocols and epistemic models.

A second interesting result of these authors is that the communication patterns allowed in the *Iterated Immediate Shapshot* (IIS) distributed model can also be described using epistemic-logic tools from Dynamic Epistemic Logic (DEL): the communication in a distributed model can be modeled with an action model capturing the communication events that can occur, and the *restricted modal product* operator shows how knowledge evolves after agents exchange information in a *communication round*.

We observe that the action models of [22] describing communication in the IIS model have drawbacks: First, such action models are different for each communication round (an ideal representation of communication would not depend on the communication rounds that have been executed so far). In addition, the size of such action models grows exponentially in the number of rounds. Moreover, such action models are structurally isomorphic to the epistemic models we wish to compute. The action models of [22], therefore, not only are not useful for computing the epistemic model resulting from a communication event, but are not a succinct representation of the communication that can happen in the IIS model.

Contributions. We are interested in the following question: in the spirit of the action-model approach to DEL, is it possible to describe the communication in a distributed model in a compact manner? As a first step, we try to salvage the approach of [22], by attempting to find an action model applicable to every communication round for *two* agents with binary inputs in the IIS model. In [46] (See Sect. 3.1.), we exhibit a family of action models with a constant number of events, although each event is labeled with a precondition formula whose size does increase at each communication round. For obtaining these action models, it was crucial to know *in advance* the epistemic model after a communication round. We have not been able to find a similar family for *three or more* agents yet. The case of m-ary inputs for $m \geq 3$ would be even harder to analyze.

The drawbacks of the action models proposed in [22], together with our unsuccessful efforts to find action models for IIS of small size, are motivations for investigating a different approach. We hence propose new dynamic epistemic logics that allow us to easily derive models of small size for a class of message-passing models sometimes called either *dynamic-network models* [7, 19], or *message adversaries* [2]. Roughly speaking, in

a dynamic-network model, the agents execute infinite sequences of communication rounds. In each round, the agents communicate according to a *communication pattern* that specifies who communicates with whom in that round. A proper subclass of dynamic-network models are those known as *oblivious* [19] that are specified with a set of communication patterns that can occur in any round, regardless of the communication patterns that have occurred so far in the execution. The IIS model can be defined as an oblivious dynamic-network model.

Our main contributions are two simple but powerful dynamic epistemic logics: *pattern models* and *parametrized pattern models*. Such logics are based on action models, and allow compact descriptions of oblivious dynamic-network models. First, in Ch. 3, we assume the full-information protocol, usually studied in distributed computing due to its generality to define the pattern models. Next, in Ch. 5, we parametrize the pattern models logic so as to study systems where agents execute arbitrary (deterministic) protocols by modifying the update mechanism considering an arbitrary protocol definition. Such a parametrization, in addition to a systematic construction of pattern models describing an adversary, makes the approach amenable to automated formal verification of distributed systems: Given a set of inputs, a protocol and a number r of rounds, an automated process is able to create an initial epistemic model, update the model r times, and verify system properties.

For the case of oblivious models, the pattern model remains the same all through the execution. Hence, we are able to model communication of oblivious dynamic-network models in constant space. Furthermore, we present a sufficient condition for verifying if the consensus task is unsolvable in a given oblivious adversary by analyzing the initial epistemic model and its corresponding pattern model.

This thesis gathers results from two conferences papers in TARK 2021 [46] and TARK 2023 [16], and two journal papers, one in Journal of Philosophical Logic [14] and another one in The Computer Journal [17].

Structure of this thesis. The rest of this thesis is structured as follows. In Ch. 2 we establish some notation and definitions. Ch. 3 presents pattern models. We define the syntax and semantics in Sect. 3.2 and give an axiomatization of the logic in Sect. 3.3. This axiomatization proves that the logic has the same expressivity as the multi-agent epistemic logic. Sect. 3.4

defines the infinite sequences of pattern models given an arbitrary adversary. Additionally, we show here that the sequences of pattern models proposed actually describe the epistemic change from round of communication to round of communication. In Sect. 3.5 we prove that despite of the fact that action models and pattern models reduce to multi-agent epistemic model, one is able to produce updates that the other is not. Finally, Sect. 4 defines a necessary condition that implies connectivity after an arbitrary number of rounds of communication given an oblivious adversary by analyzing the structure of the initial epistemic model and the pattern model.

Ch. 5 presents parametrized pattern models, that are able to deal with arbitrary protocols. In Sect. 5.1 we define a way to describe an arbitrary deterministic protocol. In Sect.5.2, we define the simplified pattern models which improves the pattern models defined in Ch. 3. In Sect. 5.3, we define the epistemic models for distributed systems which we will use in Sect. 5.4 to interpret the the logic. In Sect. 5.5 we define the sequence of pattern models that describe an adversary and provee that the sequences of simplified pattern models proposed actually describe the epistemic change from round of communication to round of communication.

Ch. 6 closes this work. Comparison with existing work appears in Sect. 6.1. Sect. 6.2 concludes this work, and Sect. 6.3 propose some lines for future research.

Chapter 2

Technical preliminaries

After introducing this thesis in Ch. 1, we give some introductory definitions and fix the notation. We assume some familiarity with distributed systems and modal logic, for which we refer the reader to [4] and [29], for example. From now on, we consider a non-empty finite set of agents $A = \{a_1, a_2, \ldots, a_n\}$ and a non-empty finite set of propositions P.

2.1 The models of interest

We are interested in *dynamic-network models* [18, 30, 33], in which $n \geq 2$ failure-free agents, that are *local-state machines*, proceed in an infinite sequence of *synchronous* rounds of communication. In each round, the communication is specified with a *communication graph*, namely a directed graph whose vertex set is A, with each edge (a_i, a_j) indicating that a message from a_i to a_j is successfully delivered in that round. A communication graph is a reflexive binary relation on A. We will write aGb rather than $(a, b) \in G$. The *in-neighbourhood* of a in G, that we denote Ga, is the set $\{b \in A \mid bGa\}$. Let $\mathbf{G}_A$ denote the set of all communication graphs with vertex set A. Thus, a dynamic-network model Adv is specified with a set of infinite sequences of graphs of $\mathbf{G}_A$, that we call *adversary*. Intuitively, we say that an adversary Adv is *oblivious* if in every round, any communication graph in a given set can happen, regardless of the communication graphs that have happened in previous rounds. This is formalized as follows. An adversary Adv is oblivious if there exists a set of communication graphs X such that $Adv = X^\omega$, where X^ω is the language of all infinite words (sequences) of elements in X.

In consequence, an oblivious adversary can be alternatively specified through the set X of communication graphs; we will say that $Adv = X$.

2.2 Protocols

Each agent locally executes a *protocol* that specifies (1) the messages that an agent sends at the beginning of a round depending on its current local state, and (2) the agent's next local state, given its local state and the messages that it receives in the current round. Each agent starts the computation with a private input, which is usually the state of the agent at the beginning of the first round. First, the *full-information protocol* is assumed. In every round of such a protocol, an agent sends its current local state to the other agents, and then changes its local state accumulating the messages received in the current round. Next, we present a modified formalism designed to deal with arbitrary deterministic protocols.

2.3 Executions and configurations

An *execution* E of an adversary Adv is a pair (I, S^∞). In such a pair, $I = (v_1, v_2, \ldots, v_n)$ is an *input vector* denoting that a_i starts with input v_i, and S^∞ is a sequence of Adv. Each v_i belongs to an *input space* $\mathcal{I}$. An *r-execution* of Adv is a pair (I, S), where I is an input vector and S is a prefix of a sequence of Adv with $|S| = r$. A *configuration* C is an n-tuple whose i-th position is a local state of agent a_i (thus input vectors may be configurations). We say that configurations C and C' are indistinguishable for a_i if and only if $C(i) = C'(i)$. An r-execution (I, S) *ends* at a configuration C if each agent a_i has the local state $C(i)$ after the execution of the sequence of communication rounds described by S with the inputs stated by I; alternatively, we say that C is the configuration at the *end* of (I, S).

In the distributing-computing literature, it is common to regard the local states of the agents as their views. We treat both these concepts as synonyms. We can think of a view of an agent as a single variable whose value changes from round to round. Consider $S_k = [G_1, G_2, \ldots, G_k]$, and $S_{k+1} = S_k \cdot G_{k+1}$ so that (I, S_{k+1}) is a $(k+1)$-execution. For the full-information protocol, the view of an agent a_i in an execution (I, S), may be defined inductively by the

function $view_f$ as follows:

$$view_f(a_i, (I, [\,])) = I(i).$$

$$view_f(a_i, (I, S_{k+1})) = [view[1], view[2], \ldots, view[n]]$$

where

$$view[j] = \begin{cases} view_f(a_j, (I, S_k)) & \text{if } a_j G_{k+1} a_i \\ \perp & \text{otherwise.} \end{cases}$$

2.4 Epistemic models

We use *epistemic models*, that we name M^r, for representing the r-executions of a given adversary Adv. An *epistemic model* for agents A and propositions P is a triple $M = (W, \sim, L)$, where

- W is a finite set of worlds,
- $\sim : A \to \wp(W \times W)$ assigns an equivalence relation to each agent, and
- $L : W \to \wp(P)$ assigns a set of true-valued propositions to each world.

We will write $w \sim_a w'$ rather than $(w, w') \in \sim(a)$. Let us define $\sim_B = \bigcap_{a \in B} \sim_a$, and $\overset{\cup}{\sim}_B = \bigcup_{a \in B} \sim_a$, with $B \subseteq A$. For the reflexive transitive closure of $\sim$, we write $\sim^*$. Each world in M^r represents an r-execution and the equivalence relations represent the indistinguishability relations of the agents over the configurations at the end of the r-executions of Adv. An example is in Fig.2.1. The epistemic model is depicted as a graph whose nodes are the worlds, with their identifier outside the node. The labeled edges are the indistinguishability relations of the agents and the labels inside the nodes are the true valued propositions in each world. In figures of models, we will economize by representing edges by lines, rather than arrows, and omitting loops and some edges recalling that indistinguishability relations are equivalence relations.

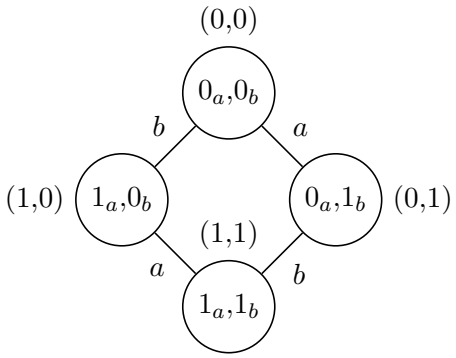


Figure 2.1: Model M^0 for agents a and b with binary inputs.

2.4.1 The initial epistemic model (M^0)

We build the initial epistemic model $M^0 = (W^0, \sim^0, L^0)$ for A, $\mathcal{I}$, and P, with $P = \{v_a \mid a \in A \text{ and } v \in \mathcal{I}\}$ so that $W^0 = \{I \mid I \text{ is an input vector for } A \text{ and } \mathcal{I}\}$, $I \sim^0_{a_i} I'$ if and only if $I(i) = I'(i)$, and $L(I) = \{v_{a_i} \in P \mid I(i) = v\}$. The epistemic model M^0 for agents $A = \{a, b\}$ and inputs $\mathcal{I} = \{0, 1\}$ is depicted in Fig. 2.1.

2.5 Syntax and semantics for $\mathcal{L}_D$

Distributed knowledge is perhaps the modality for analyzing a distributed system. If there is not the required distributed knowlede for finishing a task, the task must be unsolvable. We recall epistemic logic with distributed knowledge, $\mathcal{L}_D$, syntax and semantics interpreted in epistemic models below.

Definition 2.5.1 (Syntax)**.** *The language $\mathcal{L}_D$ is given by the following BNF grammar*

$$\varphi ::= p \mid \neg\varphi \mid \ \varphi \wedge \varphi \mid D_B\varphi$$

where $a \in A$, and $p \in P$, and $B \subseteq A$.

Definition 2.5.2 (Semantics)**.** *Let $M = (W, \sim, L)$ be an epistemic model.*

Let $p \in P$, $w, w' \in W$, $a \in A$, and $\varphi, \psi \in \mathcal{L}_D$ be given.

$$\begin{array}{ll} M, w \models p & \text{iff } p \in L(w) \\ M, w \models \neg\varphi & \text{iff } M, w \not\models \varphi \\ M, w \models \varphi \wedge \psi & \text{iff } M, w \models \varphi \text{ and } M, w \models \psi \\ M, w \models D_B\varphi & \text{iff for all } w' \text{ such that } w \sim_B w' \\ & \quad M, w' \models \varphi \end{array}$$

From now on, $\varphi \vee \psi$ will be a shorthand for $\neg(\neg\varphi \wedge \neg\psi)$, $\top$ will be a shorthand for $p \vee \neg p$, $\varphi \to \psi$ will be a shorthand for $\neg\varphi \vee \psi$, $\varphi \leftrightarrow \psi$ will be a shorthand for $(\varphi \to \psi) \wedge (\psi \to \varphi)$, and $K_a\varphi$ will be a shorthand for $D_{\{a\}}\varphi$.

2.6 Syntax and semantics for $\mathcal{L}_{DC}$

Common knowledge is another modal operator of epistemic logic. We recall the syntax and semantics of $\mathcal{L}_{DC}$, that enrich the logic $\mathcal{L}_D$ with common knowledge.

Definition 2.6.1 (Syntax)**.** *The language $\mathcal{L}_D$ is given by the following BNF grammar*

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid D_B\varphi \mid C_B\varphi$$

where $a \in A$, and $p \in P$, and $B \subseteq A$.

Definition 2.6.2 (Semantics)**.** *Let $M = (W, \sim, L)$ be an epistemic model. Let $p \in P$, $w, w' \in W$, $a \in A$, and $\varphi, \psi \in \mathcal{L}_{CD}$ be given.*

$$\begin{array}{ll} M, w \models p & \text{iff } p \in L(w) \\ M, w \models \neg\varphi & \text{iff } M, w \not\models \varphi \\ M, w \models \varphi \wedge \psi & \text{iff } M, w \models \varphi \text{ and } M, w \models \psi \\ M, w \models D_B\varphi & \text{iff for all } w' \text{ such that } w \sim_B w' M, w' \models \varphi \\ M, w \models C_B\varphi & \text{iff for all } w' \text{ such that } w \overset{\cup}{\sim}{}^*_B w' M, w' \models \varphi \end{array}$$

Let us recall that $\overset{\cup}{\sim}{}^*_B$ is the reflexive transitive closure of $\bigcup_{a \in B} \sim_a$.

2.7 Distinguishing formula

Given a model M with domain W, a formula δ is *distinguishing* for a subset W' of W if $M, w \models \delta$ for all $w \in W'$ and $M, w \not\models \delta$ for all $w \in W \backslash W'$. If W' is a singleton $\{w\}$, we say that δ is distinguishing for w. In order to refer in the logical language to a given world in the model we use *distinguishing formulas* that are true in a particular world but not in any other non-equivalent world of the model. In other words, it *distinguishes* that world from all other worlds in the model. Given that our models are for a finite set of agents A and a finite set of atomic propositions P, such a distinguishing formula always exists, and can be defined recursively and thus constructed.

We use the distinguishing formulas [41] as preconditions for *events*. In particular, we need to determine the worlds in which a communication graph can occur. Such formulas distinguish a world w of a given epistemic model from all the other worlds. The δ_w formulas are defined inductively.

Let $M = (W, \sim, L)$ be an epistemic model, and $w \in W$.

$$\delta^0_w = \tau_w = \bigwedge_{p \in L(w)} p \wedge \bigwedge_{p \notin L(w)} \neg p$$

$$\delta^{n+1}_w = \tau_w \wedge \bigwedge_{a \in A} \bigwedge_{w \sim_a w'} \neg K_a \neg \delta^n_{w'} \wedge \bigwedge_{a \in A} K_a \bigvee_{w \sim_a w'} \delta^n_{w'}$$

Note that in each step of this procedure the set of worlds satisfying δ becomes finer. Initially, at step 0, w is grouped together with all worlds having the same valuation of atoms as in w, and only distinguished from the worlds having a different valuation. But at step one, w is grouped together with all worlds having the same valuation and such that there is a match in the valuations of all worlds that are accessible from w. This set will then be smaller at the end of each further iteration.

As the model is finite, there is $n \in \mathbb{N}$ such that $\delta^{n+1}_w = \delta^n_w$, where it is safe to take $n = |W|$. We therefore take $\delta^{|W|}_w$ to be the distinguishing formula δ_w.[1] Furthermore, for a set $S \subseteq W$ of worlds, the distinguishing formula δ_S is the disjunction of the distinguishing formulas δ_w for the worlds $w \in S$, that is, as $\delta_S := \bigvee_{w \in S} \delta_w$.

[1] It should be noted that we cannot distinguish a world w from so-called bisimilar worlds, because bisimilar worlds satisfy the same formulas. This is a technical detail that one can easily be precise about [41] but that plays no role here.

2.8 Collective bisimulation

Definition 2.8.1 (Collective bisimulation)**.** *A relation Z between the domains of epistemic models $M = (W, \sim, L)$ and $M' = (W', \sim', L')$ is a* (collective) bisimulation*, notation $Z : M \leftrightarroweq M'$, if for all $(w, w') \in Z$:*

- **atoms***: for all $p_a \in P$, $p_a \in L(w)$ iff $p_a \in L'(w')$;*
- **forth***: for all nonempty $B \subseteq A$ and for all $v \in W$, if $w \sim_B v$ then there is $v' \in W'$ such that $(v, v') \in Z$ and $w' \sim_B v'$;*
- **back***: for all nonempty $B \subseteq A$ and for all $v' \in W'$, if $w' \sim_B v'$ then there is $v \in W$ such that $(v, v') \in Z$ and $w \sim_B v$.*

We additionally define a collective bisimulation bounded by n*, as a set of relations $Z^0 \supseteq Z^1 \cdots \supseteq Z^n$ of i-bisimulations for $0 \leq i \leq n$. Relation Z^0 merely satisfies* **atoms***, and for all $(w, w') \in Z^{n+1}$:*

- **atoms***: for all $p_a \in P$, $p_a \in L(w)$ iff $p_a \in L'(w')$;*
- **forth**-$(n+1)$*: for all nonempty $B \subseteq A$ and for all $v \in W$, if $w \sim_B v$ then there is $v' \in W'$ such that $(v, v') \in Z^n$ and $w' \sim_B v'$.*
- **back**-$(n+1)$*: for all nonempty $B \subseteq A$ and for all $v' \in W'$, if $w' \sim_B v'$ then there is $v \in W$ such that $(v, v') \in Z^n$ and $w \sim_B v$.*

If there is a bisimulation Z between M and M' we write $M \leftrightarroweq M'$, and if there is one containing (w, w') we write $(M, w) \leftrightarroweq (M', w')$. We then say that M and M', respectively (M, w) and (M', w'), are bisimilar*. If Z is bounded by n we write $(M, w) \leftrightarroweq^n (M', w')$ and we say that (M, w) and (M', w') are* n-bisimilar*.*

Bounded bisimulations are used to compare models (M, w) and (M', w') up to a depth n from the respective points w and w'. Collective n-bisimilarity implies that both models satisfy the same $\mathcal{L}^-$ formulas of modal depth at most n, as a minor variation of the standard result in [9].

2.9 Update expressivity

To compare dynamic modalities we define *updates* and *update expressivity*.

Definition 2.9.1 (Update, update expressivity)**.** *An* update *(or* update relation*) is a binary relation X on a class of pointed epistemic models. Given updates X and Y, X* is update equivalent to *Y, if for all pointed epistemic models (M, w) the update of (M, w) with X is collectively bisimilar to the update of (M, w) with Y. Update modalities $[X]$ and $[Y]$ are update equivalent, if X and Y are update equivalent. (For more refined notions see [45].)*

A language $\mathcal{L}$ is at least as update expressive as *$\mathcal{L}'$ if for every update modality $[X]$ of $\mathcal{L}'$ there is an update modality $[Y]$ of $\mathcal{L}$ such that X is update equivalent to Y. Language $\mathcal{L}$ is* equally update expressive as *$\mathcal{L}'$ (or 'as update expressive as'), if $\mathcal{L}$ is at least as update expressive as $\mathcal{L}'$ and $\mathcal{L}'$ is at least as update expressive as $\mathcal{L}$. Language $\mathcal{L}$ is* (strictly) more update expressive than *$\mathcal{L}'$, if $\mathcal{L}$ is at least as update expressive as $\mathcal{L}'$ and $\mathcal{L}'$ is not at least as update expressive as $\mathcal{L}$. Languages $\mathcal{L}$ and $\mathcal{L}'$ are incomparable in update expressivity if if $\mathcal{L}$ is not at least as update expressive as $\mathcal{L}'$ and $\mathcal{L}'$ is not at least as update expressive as $\mathcal{L}$.*

2.10 Iterated Immediate Snapshot model

The IIS model [12] is a fundamental model that fully captures what can be solved in asynchronous wait-free shared-memory systems with agent-crash failures. We can view IIS as a failure-free synchronous oblivious dynamic-network adversary. The set of communication graphs describing the adversary is as follows. For every sequence, $S = [C_1, C_2, \ldots, C_k]$, of non-empty subsets of A satisfying (1) $A = \bigcup C_i$ and (2) $C_i \cap C_j = \emptyset$ for $i \neq j$, the adversary has the communication graph with a directed edge (a, b) for every pair of agents $a \in C_i, b \in C_j$ with $1 \leq i \leq j \leq k$. We say that C_i is a *concurrency class*. For example, given the set of agents $\{a, b\}$, the concurrency classes are $\{a\}$, $\{b\}$, and $\{a, b\}$. In Fig. 2.2, we show the communication graphs describing the adversary for two-agent IIS. There is a graph identifier above each one: $G^{a.b}$ for $\{a\}\{b\}$, $G^{b.a}$ for $\{b\}\{a\}$ and G^{ab} for $\{a, b\}$. In Fig. 2.3, we show the epistemic model M^1_{IIS} that represents the configurations at the end of the first round of communication of the full-information protocol for agents a and b with binary input in the IIS model. In the epistemic model, the first element of an identifier corresponds to a shorthand for an identifier in M^0: 0 is a shorthand for $(0, 0)$, 1 is a shorthand for $(0, 1)$, 2 is a shorthand for $(1, 0)$, and 3 is a shorthand for $(1, 1)$. The second element of the identifiers corresponds to a graph identifier.

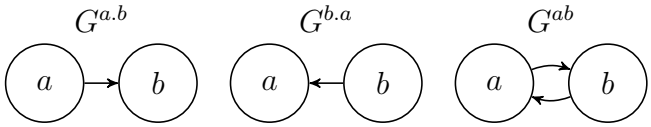


Figure 2.2: Communication graphs for two-agent IIS.

2.11 Action models

Action models were introduced in [5] as a way to model dynamics of knowledge via events. The syntax, and semantics are defined as follows.

Definition 2.11.1 (Action model)**.** *An* action model M *is a triple* $(\mathsf{E}, \sim, \mathsf{Pre})$*, where*

- E *is a non-empty finite set of events,*
- $\sim : \mathsf{A} \to \wp(\mathsf{E} \times \mathsf{E})$ *is a function that associates each agent with an equivalence relation over the set of events, and*
- $\mathsf{Pre} : \mathsf{E} \to \mathcal{L}_D$ *is a function that associates each event with a precondition.*

A pointed action model *is a pair* (M, e) *where* $\mathsf{e} \in \mathsf{E}$*.*

Definition 2.11.2 (Syntax)**.** *The language* $\mathcal{L}_\otimes$ *is given by the following BNF grammar*

$$\varphi ::= p \mid \neg\varphi \mid \ \varphi \wedge \varphi \mid D_B\varphi \mid [\mathsf{M}, \mathsf{e}]\varphi$$

where $a \in A$*, and* $p \in P$*, and* $B \subseteq A$*, and* $[\mathsf{M}, \mathsf{e}]$ *is an update.*

Definition 2.11.3 (Semantics and restricted modal product)**.** *Let* $p \in P$*,* $a \in A$*,* $B \subseteq A$*, a pointed action model* (M, e)*, an epistemic model* $M = (W, \sim, L)$*, and* $w \in W$ *be given. The* satisfaction relation $\models$ *is defined by induction on* $\varphi \in \mathcal{L}_\otimes$*.*

$$\begin{array}{lll}
M, w \models p & \textit{iff} & p \in L(w) \\
M, w \models \neg\varphi & \textit{iff} & M, w \not\models \varphi \\
M, w \models \varphi \wedge \psi & \textit{iff} & M, w \models \varphi \textit{ and } M, w \models \psi \\
M, w \models D_B\varphi & \textit{iff} & M, v \models \varphi \textit{ for all } v \sim_B w \\
M, w \models [\mathsf{M}, \mathsf{e}]\varphi & \textit{iff} & M, w \models \mathsf{pre}(\mathsf{e}) \textit{ implies } M \otimes \mathsf{M}, (w, \mathsf{e}) \models \varphi
\end{array}$$

where $M \otimes \mathsf{M} = (W', \sim', L')$ *is defined so that:*

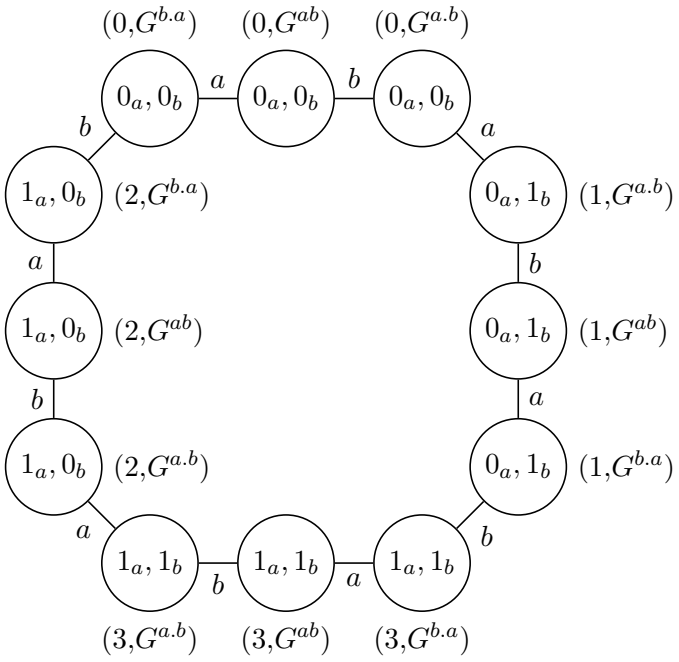


Figure 2.3: Epistemic model M^1_{IIS} that represents the configurations at the end of the first round of the full-information protocol for two agents a and b with binary inputs in the IIS model.

- $W' = \{(w, \mathsf{e}) \in W \times \mathsf{E} \mid M, w \models \mathsf{Pre}(\mathsf{e})\}$
- $\sim'_a = \{((w, \mathsf{e}), (w', \mathsf{e}')) \in W' \times W' \mid w \sim_a w' \text{ and } \mathsf{e} \sim_a \mathsf{e}'\}$
- $L'((w, \mathsf{e})) = L(w)$

Now that we have established the needed technical preliminaries and fixed some notation in this chapter, we move to present pattern models in Ch. 3.

Chapter 3

Pattern models

After establishing technical preliminaries in Ch. 2, we are now in a position to present the pattern models logic which allows the description of arbitrary adversaries and agents communicating using the full-information protocol. In 3.1, we recall an apparently compact family of action models of six events for two-agent IIS that was a motivation for working in this new approach. In Sect. 3.2 we define the pattern model structure and the semantics of the logic. In Sect. 3.3 we give an axiomatization for the logic. In Sect. 3.4 we describe the construction of an infinite sequence of pattern models describing a given adversary and prove that there exists a one-to-one correspondence between the worlds in the updated models and the configurations of the distributed system. Moreover, the relations of the worlds correspond to the indistinguishability relations of the configurations. Finally, in Sect. 3.5 we show that action models are incomparable in update expressivity.

3.1 A family of action models for two-agent IIS

Before presenting pattern models, we present the most compact family of action models that we found for two-agent (a and b) IIS with binary inputs. We exploit the fact that for two-agent IIS, *the epistemic models will always be bipartite graphs*. We can hence partition the set of worlds in M^i into two sets W_1^i and W_2^i so that any pair of distinct worlds in the same set can be distinguished by both agents. For each set W_j^i, we use three events to represent the different sequences of concurrency classes that can happen in

a round: $\{a\}\{b\}$, $\{a,b\}$, and $\{b\}\{a\}$ (see Fig. 2.2). Thus we have six events: three for operating with the worlds in W_1^i, and three for operating with the worlds in W_2^i. The sketch of the action model is shown in Fig. 3.1. The preconditions, φ_1 and φ_2, change from round to round.

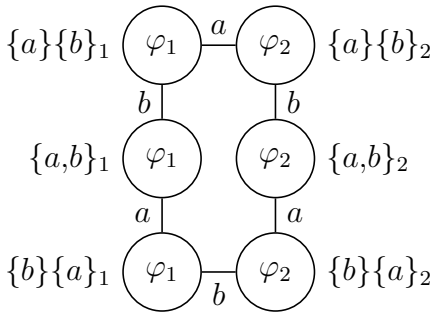


Figure 3.1: Sketch of the action model for two-agent IIS with binary inputs.

φ_j is a disjunction of the δ_w formulas describing the worlds in W_j^i. For the first round, if we consider $W_1^0 = \{(0,0),(1,1)\}$ and $W_2^0 = \{(0,1),(1,0)\}$, possible preconditions are: $\varphi_1 = (0_a \wedge 0_b) \vee (1_a \wedge 1_b)$, and $\varphi_2 = (0_a \wedge 1_b) \vee (1_a \wedge 0_b)$. Note that $\varphi_j = \delta^0_{W_j^0}$ identify the worlds in the initial epistemic model. In general, we propose to compute the distinguishing formulas using $\delta^{|W|}$.

This approach appears to be a succinct representation of the full-information execution dynamics. There are, however, still issues. We would like to represent communication defined by an oblivious model *just once* because the allowed communication patterns are the same regardless of the round. All correct action models we have been able to find have preconditions that change from round to round. Moreover, the size of the *distinguishing formulas* grows exponentially in the number of rounds. This suggests that in certain cases, a straightforward application of action models might not be ideal.

We have not been able to find a similar family of action models for three agents. We would need to analyze if the corresponding epistemic models are always n-partite, and how we could join all the needed events. Finding action models for the case of m-ary inputs for $m \geq 3$ would be even harder. Making things worse, the analysis might be different in distinct models: we would need to study each model to take advantage of its own characteristics. All these facts motivated us to look for a different and more appropriate approach.

3.2 Pattern models logic

Definition 3.2.1 (Pattern model)**.** *A* pattern model*, $\mathcal{P}$, is a pair* $(\mathbf{G}, Pre)$ *where*

- $\mathbf{G}$ *is a set of communication graphs*
- $Pre : \mathbf{G} \to \mathcal{L}_D$ *is a function that assigns a precondition to each communication graph.*

A *pointed pattern model* is a pair $(\mathcal{P}, G)$ where $G \in \mathbf{G}$.

Definition 3.2.2 (Syntax)**.** *The language $\mathcal{L}_\odot$ is given by the following BNF grammar*

$$\varphi ::= p \mid \neg\varphi \mid \ \varphi \wedge \varphi \mid D_B\varphi \mid [\mathcal{P}, G]\varphi$$

where $a \in A$, and $p \in P$, and $B \subseteq A$, and $[\mathcal{P}, G]$ is an update.

Definition 3.2.3 (Semantics and restricted modal product)**.** *Let $p \in P$, $a \in A$, $B \subseteq A$, a pointed pattern model $(\mathcal{P}, G)$, an epistemic model $M = (W, \sim, L)$, and $w \in W$ be given. The* satisfaction relation $\models$ *is defined by induction on $\varphi, \psi \in \mathcal{L}_\odot$.*

$$\begin{array}{lll} M, w \models p & \textit{iff} & p \in L(w) \\ M, w \models \neg\varphi & \textit{iff} & M, w \not\models \varphi \\ M, w \models \varphi \wedge \psi & \textit{iff} & M, w \models \varphi \textit{ and } M, w \models \psi \\ M, w \models D_B\varphi & \textit{iff} & M, v \models \varphi \textit{ for all } v \sim_B w \\ M, w \models [\mathcal{P}, G]\varphi & \textit{iff} & M, w \models Pre(G) \textit{ implies } M \odot \mathsf{M}, (w, G) \models \varphi \end{array}$$

where $M \odot \mathcal{P} = (W', \sim', L')$ is defined as follows:

- $W' = \{(w, G) \in W \times \mathbf{G} \mid M, w \models Pre(G)\}$
- $\sim'_a = \{((w, G), (w', G')) \in W' \times W' \mid \underline{Ga = G'a} \textit{ and } \underline{w \sim_{Ga} w'}\}$
- $L'((w, G)) = L(w)$

Intuitively, the first underlined condition amounts to agent a receiving information from the same set of agents whenever the communication is described by G or G' because the in-neighbourhood of a must be the same in both G and G'. The second one, in turn, is equivalent to all agents in such a set to send the same information in both cases. Since such agents do not distinguish between w and w', the agents are in the same local state in w and w'. Therefore, they send the same messages in either occurrence of G or G'.

3.3 Axiomatization

We proceed with the axiomatization. The axiomatization of the logic of pattern models is that of the logic of distributed knowledge expanded with reduction axioms $\mathbf{C}^1$—$\mathbf{C}^4$ and rule $\mathbf{N}^{\odot}$ involving pattern models, and the auxiliary derivation rule **RE** (replacement of equivalents). [1] The axiomatization is displayed in Table 3.1. A *derivation* is a sequence of formulas such that every formula is an instantiation of an axiom, or the conclusion of an instantiation of a derivation rule where the premisses (or premiss) are prior formulas in the sequence. A formula occurring in a derivation is a *theorem*.

P	all instances of propositional tautologies
$\mathbf{K}^D$	$D_B(\varphi \to \psi) \to D_B\varphi \to D_B\psi$
$\mathbf{T}^D$	$D_B\varphi \to \varphi$
$\mathbf{4}^D$	$D_B\varphi \to D_B D_B\varphi$
$\mathbf{5}^D$	$\neg D_B\varphi \to D_B \neg D_B\varphi$
W	$D_B\varphi \to D_C\varphi$
$\mathbf{C}^1$	$[\mathcal{P}, G]p \leftrightarrow Pre(G) \to p$
$\mathbf{C}^2$	$[\mathcal{P}, G]\neg\varphi \leftrightarrow \neg[\mathcal{P}, G]\varphi$
$\mathbf{C}^3$	$[\mathcal{P}, G](\varphi \wedge \psi) \leftrightarrow ([\mathcal{P}, G]\varphi \wedge [\mathcal{P}, G]\psi)$
$\mathbf{C}^4$	$[\mathcal{P}, G]D_B\varphi \leftrightarrow Pre(G) \to (\bigwedge_{G'B \equiv GB} D_{GB} Pre(G') \to D_{GB}[\mathcal{P}, G']\varphi)$ where $GB \equiv G'B$ is defined as "for all $a \in B$, $Ga = G'a$"
MP	From $\varphi \to \psi$ and φ infer ψ
$\mathbf{N}^D$	From φ infer $D_B\varphi$
$\mathbf{N}^{\odot}$	From φ infer $[\mathcal{P}, G]\varphi$
RE	From $\varphi \leftrightarrow \psi$ infer $\chi[p/\varphi] \leftrightarrow \chi[p/\psi]$

Table 3.1: Axiomatization of pattern models logic, where $\emptyset \neq B \subseteq C \subseteq A$, $a \in A$, and $p \in P$.

In the derivation rule **RE**, $\chi[p/\varphi]$ stands for uniform substitution of the occurrences of atom p in formula χ by φ. The reduction axioms for pattern models resemble those for action models [5], except for the reduction axiom for distributed knowledge after update. There is no reduction axiom for a sequence of two pattern models. This explains the presence of the derivation

[1] $\mathbf{C}^1$—$\mathbf{C}^4$ are called reduction rules because we can use this rules and **RE** for transforming any $\varphi \in \mathcal{L}_{\odot}$ into an equivalent formula $\varphi' \in \mathcal{L}_D$.

rule **RE**, which is not needed in the axiomatization of distributed knowledge. The validity of all axioms and the validity preservation of all rules is obvious, except for that of $\mathbf{C}^4$, $\mathbf{N}^{\odot}$, and **RE** which we prove below.

Proposition 3.3.0.1 ($\mathbf{C}^4$)**.**
$\models [\mathcal{P}, G]D_B\varphi \leftrightarrow Pre(G) \rightarrow (\bigwedge_{G'B \equiv GB} D_{GB}Pre(G') \rightarrow D_{GB}[\mathcal{P}, G']\varphi)$.

Proof. Let $M = (W, \sim, L)$, $w \in W$, $\mathcal{P} = (\mathbf{G}, Pre)$ and $(W', \sim', L') = M' = M \odot \mathcal{P}$. Then:

$M, w \models [\mathcal{P}, G]D_B\varphi$
$\Leftrightarrow$ (semantics of pattern models)
$M, w \models Pre(G)$ implies $M \odot \mathcal{P}, (w, G) \models D_B\varphi$
$\Leftrightarrow$ (semantics of distributed knowledge)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : (w, G) \sim'_B (w', G') \Rightarrow M \odot \mathcal{P}, (w', G') \models \varphi$
$\Leftrightarrow$ (definition of $\odot$)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} :\ M, w' \models Pre(G')$ & $\forall a \in B : Ga = G'a$ & $w \sim_{Ga} w' \Rightarrow M \odot \mathcal{P}, (w', G') \models \varphi$
$\Leftrightarrow$
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} :\ M, w' \models Pre(G')$ & $\forall a \in B : Ga = G'a$ & $\forall a \in B : w \sim_{Ga} w' \Rightarrow M \odot \mathcal{P}, (w', G') \models \varphi$
$\Leftrightarrow$ ($GB \equiv G'B$ is defined as "for all $a \in B$, $Ga = G'a$")
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} :\ M, w' \models Pre(G')$ & $GB \equiv G'B$ & $\forall a \in B : w \sim_{Ga} w' \Rightarrow M \odot \mathcal{P}, (w', G') \models \varphi$
$\Leftrightarrow$ ($\psi \Leftrightarrow \psi \wedge \psi$)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B$ & $\forall a \in B : w \sim_{Ga} w'$ & $M, w' \models Pre(G')$ & $\forall a \in B : w \sim_{Ga} w' \Rightarrow M \odot \mathcal{P}, (w', G') \models \varphi$
$\Leftrightarrow$ (definition of $\sim_B$)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B$ & $w \sim_{GB} w'$ & $M, w' \models Pre(G') \Rightarrow (w \sim_{GB} w' \Rightarrow M \odot \mathcal{P}, (w, G') \models \varphi)$
$\Leftrightarrow$ (definition of distributed knowledge)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B$ & $M, w \models D_{GB}Pre(G') \Rightarrow (w \sim_{GB} w' \Rightarrow M \odot \mathcal{P}, (w, G') \models \varphi)$
$\Leftrightarrow$ (semantics of pattern models)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B$ & $M, w \models D_{GB}Pre(G') \Rightarrow (\forall w' \in W : w \sim_{GB} w' \Rightarrow M, w \models [\mathcal{P}, G']\varphi)$
$\Leftrightarrow$ (semantics of distributed knowledge)
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B$ & $M, w \models$

$D_{GB}Pre(G') \Rightarrow M, w \models D_{GB}[\mathcal{P}, G']\varphi$
$\Leftrightarrow (\varphi_1 \wedge \varphi_2 \Rightarrow \psi \Leftrightarrow \varphi_1 \Rightarrow \varphi_2 \Rightarrow \psi)$
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B \Rightarrow M, w \models D_{GB}Pre(G') \Rightarrow M, w \models D_{GB}[\mathcal{P}, G']\varphi$
$\Leftrightarrow$
$M, w \models Pre(G)$ implies $\forall w' \in W,\ \forall G' \in \mathbf{G} : GB \equiv G'B \Rightarrow M, w \models D_{GB}Pre(G') \rightarrow D_{GB}[\mathcal{P}, G']\varphi$
$\Leftrightarrow$
$M, w \models Pre(G)$ implies $M, w \models \bigwedge_{G'B \equiv GB} D_{GB}Pre(G') \rightarrow D_{GB}[\mathcal{P}, G']\varphi$
$\Leftrightarrow$
$M, w \models Pre(G) \rightarrow (\bigwedge_{G'B \equiv GB} D_{GB}Pre(G') \rightarrow D_{GB}[\mathcal{P}, G']\varphi)$ □

The interaction between pattern models and distributed knowledge of Prop. 3.3.0.1 is reminiscent of [48, Prop. 5]; it is similar to [6, Prop. 4.6], and also somewhat similar to [52, Prop. 2, items 6 & 7].

Proposition 3.3.0.2 ($\mathbf{N}^{\odot}$)**.** *If* $\models \varphi$*, then* $\models [\mathcal{P}, G]\varphi$*.*

Proof. In order to show that $\models [\mathcal{P}, G]\varphi$, let $M = (W, \sim, L)$ and $w \in W$ be given. We wish to show that $M, w \models [\mathcal{P}, G]\varphi$. By definition of the semantics, this is equivalent to $M, w \models Pre(G)$ implies $M \odot \mathcal{P}, (w, G) \models \varphi$. On the one hand, if $M, w \not\models Pre(G)$, $M, w \models [\mathcal{P}, G]\varphi$ holds. On the other hand, if $M, w \models [\mathcal{P}, G]\varphi$, since we assumed that φ is valid, $M \odot \mathcal{P}, (w, G) \models \varphi$ holds. □

Proposition 3.3.0.3 (**RE**)**.** *If* $\models \varphi \leftrightarrow \psi$*, then* $\models \chi[p/\varphi] \leftrightarrow \chi[p/\psi]$*.*

Proof. This is proved by induction on χ.

Base case. $\chi = p$ with $p \in P$.
$M, w \models p[p/\varphi]$
$\Leftrightarrow$ (uniform substitution)
$M, w \models \varphi$
$\Leftrightarrow$ (hypothesis)
$M, w \models \psi$
$\Leftrightarrow$ (uniform substitution)
$M, w \models p[p/\psi]$

Inductive hypothesis.
For any $\varphi, \psi, \chi_1, \chi_2 \in \mathcal{L}_{\odot}$. If $\models \varphi \leftrightarrow \psi$, then $\models \chi_1[p/\varphi] \leftrightarrow \chi_1[p/\psi]$, and $\models \chi_2[p/\varphi] \leftrightarrow \chi_2[p/\psi]$.

Inductive step.
Let us assume $\models \varphi \leftrightarrow \psi$.

Case $\chi = (\neg\chi_1)[p/\varphi]$
$M, w \models (\neg\chi_1)[p/\varphi]$
$\Leftrightarrow$ (semantics of $\neg$)
$M, w \not\models (\chi_1)[p/\varphi]$
$\Leftrightarrow$ (inductive hypothesis)
$M, w \not\models (\chi_1)[p/\psi]$
$\Leftrightarrow$ (semantics of $\neg$)
$M, w \models (\neg\chi_1)[p/\varphi]$

Case $\chi = (\chi_1 \wedge \chi_1)[p/\varphi]$
$M, w \models (\chi_1 \wedge \chi_2)[p/\varphi]$
$\Leftrightarrow$ (semantics of $\wedge$)
$M, w \models (\chi_1)[p/\varphi]$ and $M, w \models (\chi_2)[p/\varphi]$
$\Leftrightarrow$ (inductive hypothesis)
$M, w \models (\chi_1)[p/\psi]$ and $M, w \models (\chi_2)[p/\psi]$
$\Leftrightarrow$ (semantics of $\wedge$)
$M, w \models (\chi_1 \wedge \chi_2)[p/\psi]$

Case $\chi = (D_B\chi_1)[p/\varphi]$
$M, w \models (D_B\chi_1)[p/\varphi]$
$\Leftrightarrow$ (uniform substitution)
$M, w \models D_B\chi_1[p/\varphi]$
$\Leftrightarrow$ (semantics of distributive knowledge)
$M, v \models \chi_1[p/\varphi]$ for all $v \sim_B w$
$\Leftrightarrow$ (inductive hypothesis)
$M, v \models \chi_1[p/\psi]$ for all $v \sim_B w$
$\Leftrightarrow$ (similar to above)
$M, w \models (D_B\chi_1)[p/\psi]$

Case $\chi = ([\boldsymbol{R}, G]\chi_1)[p/\varphi]$
$M, w \models ([\mathcal{P}, G]\chi')[p/\varphi]$
$\Leftrightarrow$ (uniform substitution)
$M, w \models [\mathcal{P}, G]\chi'[p/\varphi]$

$\Leftrightarrow$ (semantics of pattern models)
$M, w \models \chi'$ implies $M \odot \mathcal{P}, (w, G) \models \chi'[p/\varphi]$
$\Leftrightarrow$ (inductive hypothesis)
$M, w \models \chi'$ implies $M \odot \mathcal{P}, (w, G) \models \chi'[p/\psi]$
$\Leftrightarrow$ (similar to above)
$M, w \models ([\mathcal{P}, G]\chi')[p/\psi]$ □

Theorem 3.3.1. *The axiomatization of pattern model logic in Table 3.1 is sound and complete with respect to the semantics in Def. 3.2.3.*

Proof. Soundness follows from the literature on distributed knowledge (see, e.g., [43]) and from the above Props. 3.3.0.1, 3.3.0.2, and 3.3.0.3.

Completeness follows from (*i*) the completeness of the logic of distributed knowledge [43, 36, 48], (*ii*) the admissibility of the derivation rule **RE** in that axiomatization (as in [52], for a similar setting), and (*iii*) termination of an inside-out reduction[2] showing that formulas containing pattern model modalities are provably equivalent to formulas without pattern models (where **RE** is essential, as explained in general in [47]). □

Let us discuss how we are able to determine whether a given formula $\varphi \in \mathcal{L}_{\odot}$ is a theorem: If φ does not contain pattern model modalities, determine whether it is a theorem in the logic of distributed knowledge. Otherwise, apply $\mathbf{C}^1$—$\mathbf{C}^4$ repeatedly to the innermost formula $[\mathcal{P}, G]\psi$, where $\psi \in \mathcal{L}_D$, to one side of $\varphi \leftrightarrow \varphi$ until you get a formula $\varphi' \in \mathcal{L}_D$ provably equivalent to φ and determine whether it is a theorem in the logic of distributed knowledge.

In the axioms $\mathbf{C}^2$—$\mathbf{C}^4$, on the left-hand side the pattern model binds a formula of higher complexity than the formula bound by the pattern model on the right-hand side. In $\mathbf{C}^1$ the pattern model disappears on the right-hand side. By successively applying these axioms the pattern model (update) modality disappears.

The completeness result is a generalization of the completeness of the similar logic of resolving distributed knowledge in [52], and a generalization of the completeness of the similar logic with reading map modalities in [6], and also a special case of the conjectured completeness of the logic with arbitrary reading events in [6]. Rather than presenting our complete axiomatization

[2] An inside-out reduction is a transformation of a formula $\varphi \in \mathcal{L}_{\odot}$ into a formula $\varphi' \in \mathcal{L}_D$ by repeatedly replacing a subformula with just an update modality with one equivalent using $\mathbf{C}^1 - \mathbf{C}^4$ and **RE** until there is no more update modalities.

as an original result, with exhaustive proofs, we credit it to the authors of [52] and of [6], and refer to their proof details.

3.4 Pattern models for arbitrary adversaries

Consider any adversary Adv and the initial model M^0 as defined in Sect. 2. Here, we define an infinite sequence $\mathcal{P}^1, \mathcal{P}^2, \ldots$ of pattern models that describes the evolution of knowledge in the executions of Adv. More precisely, Theorem 3.4.2, in the next subsection, states that the epistemic model $M^r = (W^r, \sim^r, L^r) = M^0 \odot \mathcal{P}^1 \odot \mathcal{P}^2 \odot \cdots \odot \mathcal{P}^r$ captures how knowledge changes after r rounds of communication.

Given $M^{i-1} = M^0 \odot \mathcal{P}^1 \odot \mathcal{P}^2 \odot \cdots \odot \mathcal{P}^{i-1}$, we define the pattern model $\mathcal{P}^i = (\mathbf{G}^i, Pre^i)$ as follows:

- $\mathbf{G}^i = \{G \in \mathbf{G}_A \mid \text{there is an } i\text{-execution } (I, S \cdot G) \text{ of } Adv\}$.
- For every $G \in \mathbf{G}^i$, let $\mathcal{E}_G^{i-1} = \{(I, S) \mid \text{there is an } i\text{-execution } (I, S \cdot G) \text{ of } Adv\}$. Let us recall that each input vector I is a world in M^0. We denote the world $((((I, G_1), G_2), \ldots), G_{i-1}) \in W^{i-1}$ as $w_{(I,S)}$, where $S = G_1, G_2, \ldots, G_{i-1}$. Thus,

$$Pre^i(G) = \bigvee_{\{w_{(I,S)} \mid (I,S) \in \mathcal{E}_G^{i-1}\}} \delta_{w_{(I,S)}},$$

  where $\delta_{w_{(I,S)}}$ is the *distinguishing formula* of $w_{(I,S)}$.

The case of oblivious dynamic-network models. From the definition of $\mathbf{G}^i$, we can see that, for an oblivious adversary Adv, $\mathbf{G}^i = Adv$, for each $i \geq 1$. Thus, all $\mathcal{P}^i$ have the same set of communication graphs. Moreover, for each $G \in \mathbf{G}^i$, $\mathcal{E}_G^{i-1}$ contains *all* $(i-1)$-executions of Adv, and hence $Pre^i(G)$ can be set to $\top$. Therefore, $\mathcal{P}^1 = \mathcal{P}^i$ for all $i \in \mathbb{N}$.

The pattern model , $\mathcal{P}_{\text{two-IIS}} = (\mathbf{G}_{\text{two-IIS}}, Pre_{\text{two-IIS}})$, representing the dynamics for IIS with agents a and b is depicted in Fig. 3.2.

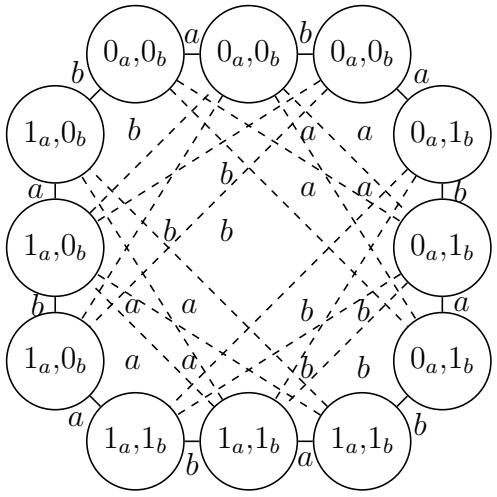


Figure 3.3: $M^1 = M^0 \otimes A_{\text{two-IIS}}$.

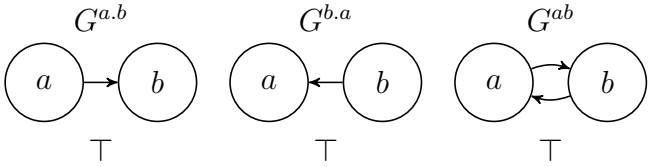


Figure 3.2: Pattern model $\mathcal{P}_{\text{two-IIS}}$ for two-agent IIS.

It is worth observing that the action model $A_{\text{two-IIS}}$, built so that the events are the communication graphs, and $G \sim_a G'$ whenever $Ga = G'a$, and similarly for b, with $\top$ as precondition in all events, and the *usual* restricted modal product $\otimes$ does not model IIS for two agents, not even for the first round. Namely, $M^1 = M^0 \otimes A_{\text{two-IIS}}$, shown in Fig. 3.3 omitting world identifiers, has wrong edges which make M^1 structurally different from a 12-cycle, which is the structure of the epistemic model for two agents with binary inputs after one round of communication in IIS (see Fig. 2.3).

Example 3.4.1. *Let us consider an adversary Adv capable of making at most one directed communication line fail permanently at each round of communication. For simplicity, consider two agents (a and b) with binary input (*0 *or* 1*). The initial epistemic model for this scenario is the one in Fig. 2.1.*

For the first round, the pattern model is $\mathcal{P}^1_{Adv}$, shown in Fig. 3.4. The updated epistemic model $M^1_{Adv} = M^0 \odot \mathcal{P}^1_{Adv}$ is shown in Fig. 3.5. For the

second round, the pattern model is $\mathcal{P}^2_{Adv}$*, shown in Fig. 3.6. In such a pattern model, the sets of worlds for calculating preconditions* $\delta_{S_{\{\}}}$ *for* $\{\}$*,* $\delta_{S_{\{a\}}}$ *for* $\{a\}$*,* $\delta_{S_{\{b\}}}$ *for* $\{b\}$ *and* $\delta_{S_{\{a,b\}}}$ *for* $\{a,b\}$ *are as follows:*

$S_{\{\}} = \{(0,\{a\}),(1,\{a\}),(2,\{a\}),(3,\{a\}),$
$(0,\{b\}),(1,\{b\}),(2,\{b\}),(3,\{b\})\}$
$S_{\{a\}} = \{(0,\{a\}),(1,\{a\}),(2,\{a\}),(3,\{a\}),(0,\{a,b\}),$
$(1,\{a,b\}),(2,\{a,b\}),(3,\{a,b\})\}$
$S_{\{b\}} = \{(0,\{b\}),(1,\{b\}),(2,\{b\}),(3,\{b\}),(0,\{a,b\}),$
$(1,\{a,b\}),(2,\{a,b\}),(3,\{a,b\})\}$
$S_{\{a,b\}} = \{(0,\{a,b\}),(1,\{a,b\}),(2,\{a,b\}),(3,\{a,b\})\}$

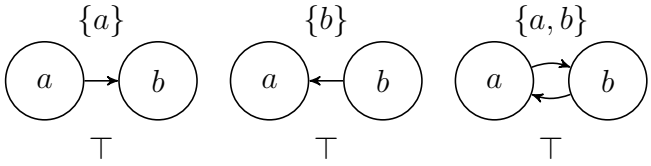


Figure 3.4: Pattern model $\mathcal{P}^1_{Adv}$.

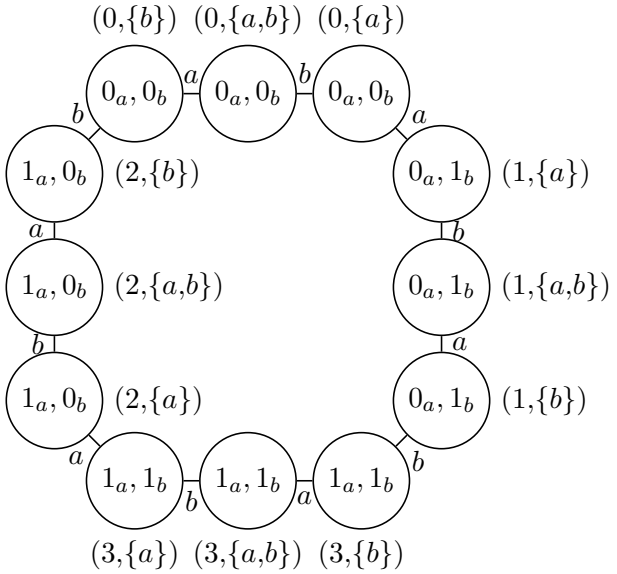


Figure 3.5: Epistemic model $M^1_{Adv} = M^0 \odot \mathcal{P}^1_{Adv}$.

$M^2_{Adv} = M^1_{Adv} \odot \mathcal{P}^2_{Adv}$ *is shown in Fig. 3.7.*

It is worth observing that in the first round, the communication graphs are those of the IIS model for two agents and the precondition is $\delta_W = \top$ in all graphs because all graphs may occur in the first round. The difference appears at the second round of communication where a fourth additional graph is added. The preconditions are distinct and change from round to round. Such preconditions model the property that once a communication line fails, the failure is permanent.

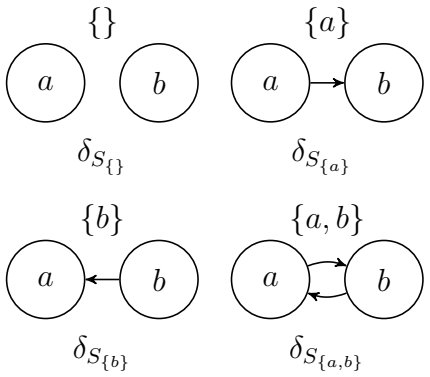


Figure 3.6: Pattern model $\mathcal{P}^2_{Adv}$.

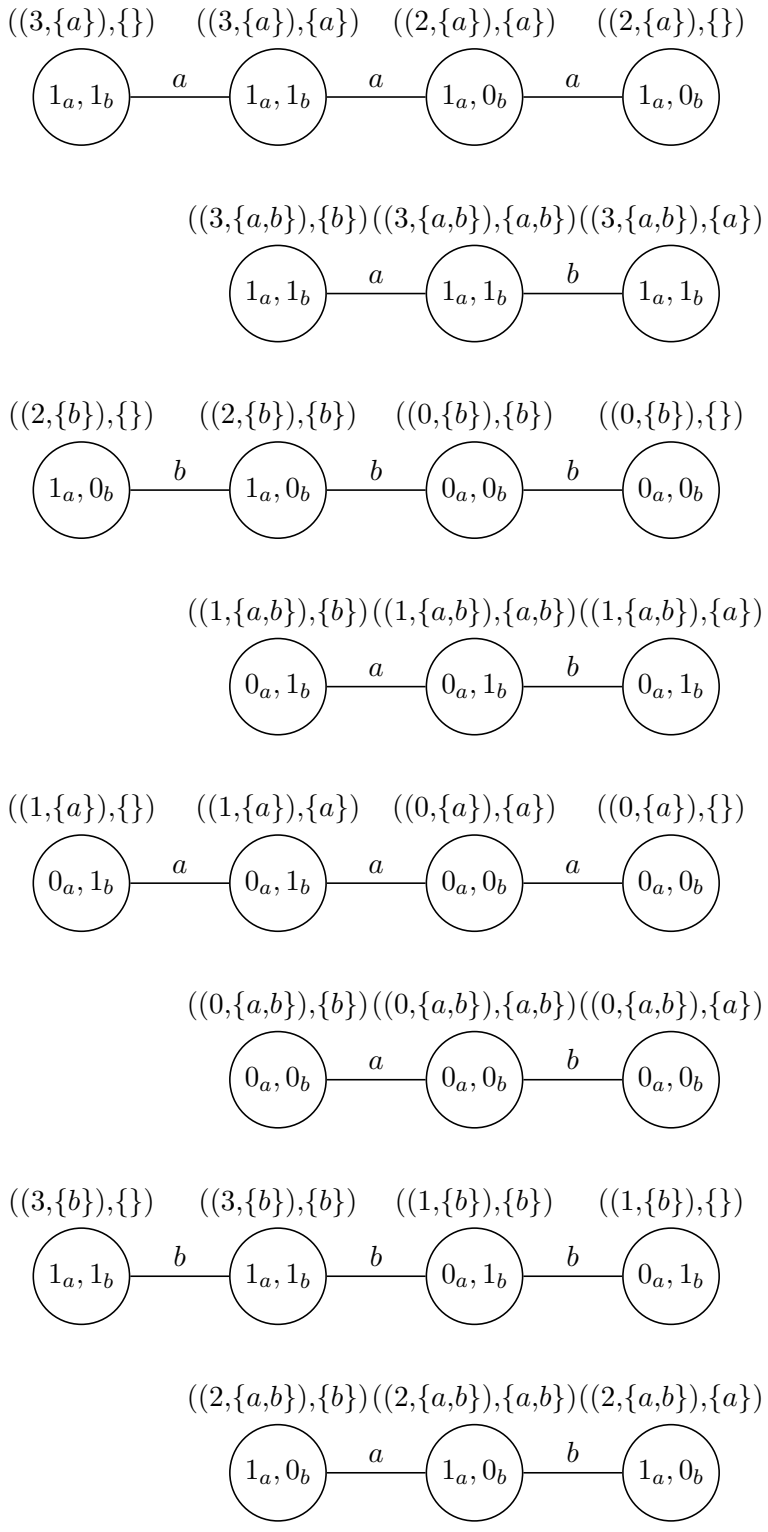


Figure 3.7: Epistemic model $M^2_{Adv} = M^1_{Adv} \odot \mathcal{P}^2_{Adv}$.

3.4.1 The $\odot$ product reflects the change in local states through rounds

The dynamic epistemic logic presented here is focused on reasoning about computations. In particular, we are interested in modeling how *configurations* change through a protocol execution. Until now, the formalism allows us to study systems where agents communicate by executing the full-information protocol. A key point is that when updating an epistemic model with the modified modal product, the resulting epistemic model describes how the local states of agents change. Theorem 3.4.2 below states that the pattern models do model knowledge dynamics. The theorem formalizes this claim using the following notion.

Definition 3.4.1. *Let Adv be an adversary. We define the set* $\mathcal{C}^i_{Adv} = \{C \mid$ *there is an* i*-execution* (I, S) *of Adv that ends in the configuration* $C\}$ *for every* $i \geq 0$*. Let* $\mathcal{P}^1, \mathcal{P}^2, \ldots$ *be an infinite sequence of pattern models. We say that the sequence* $\mathcal{P}^1, \mathcal{P}^2, \ldots$ reflects *the adversary Adv if for each* $r \geq 1$*, there is a bijection* $f^r : W^r \to \mathcal{C}^r_{Adv}$ *such that* $w \sim_{a_i} w'$ *if and only if* $f^r(w)$ *and* $f^r(w')$ *are indistinguishable for* a_i*, where* M^0 *is the initial epistemic model and* $M^r = (W^r, \sim^r, L^r) = M^0 \odot \mathcal{P}^1 \odot \mathcal{P}^2 \odot \cdots \odot \mathcal{P}^r$*. If* $\mathcal{P}^1 = \mathcal{P}^i$ *for all* $i \in \mathbb{N}$*, we simply say that* $\mathcal{P}^1$ *reflects Adv.*

Theorem 3.4.2 (Correspondence between worlds and configurations for pattern models)**.** *Let Adv be an adversary and* $\mathcal{P}^1, \mathcal{P}^2, \ldots$ *be the pattern models built from Adv, as described in Sect. 3.4. Then,* $\mathcal{P}^1, \mathcal{P}^2, \ldots$ *reflects Adv.*

The proof of Theorem 3.4.2 will be as follows. First, we will build f^i for all $i \in \mathbb{N}$ as the composition of two bijective functions. For this, we will present proofs of two auxiliary Lemmas 3.4.3, and 3.4.4. Second, we will prove by induction on the number of rounds that $w_r \sim_{a_i} w'_r$ if and only $f^r(w_r)$ and $f^r(w'_r)$ are indistinguishable for a_i.

Let $\mathcal{E}^r_{Adv}$ be the set of all r-executions of Adv. Let I, I' be two input vectors for A and $\mathcal{I}$. Consider $E_{r+1} = (I, [G_1, G_2, \ldots, G_r, G_{r+1}]) \in \mathcal{E}^{r+1}_{Adv}$. Since $E_{r+1} \in \mathcal{E}^{r+1}_{Adv}$, $E_r = (I, [G_1, G_2, \ldots, G_r]) \in \mathcal{E}^r_{Adv}$ because $[G_1, G_2, \ldots, G_r]$ is a prefix of $[G_1, G_2, \ldots, G_r, G_{r+1}]$. We define $g^r : \mathcal{E}^r_{Adv} \to \mathcal{C}^r_{Adv}$ as follows:

$$g^0((I, [\,])) = I.$$

$$g^{r+1}(E_{r+1}) = C_{r+1} = (C_{r+1}(1), C_{r+1}(2), \ldots, C_{r+1}(n))$$

where

$$C_{r+1}(i)(j) = \begin{cases} g^r(E_r)(j) & \text{if } a_j G_{r+1} a_i \\ \perp & \text{otherwise.} \end{cases}$$

Note that g^r may be defined in terms of $view_f$. Since the proofs of the following lemmas are by induction on the number r of round and the definition of $view_f$ treat the number of round implicitly, we prefer to use the definition above.

The following Lemma formally proves the quite obvious fact that, for full-information protocols, the configuration at the end of round r uniquely represents the entire r-execution.

Lemma 3.4.3. *g^r is a bijection.*

Proof. We will prove that g^r is bijective by induction on the number r of rounds.

Base case. Since the executions in $\mathcal{E}^0_{Adv}$ and the configurations in $\mathcal{C}^0_{Adv}$ can be identified with their respective input vector, g^0 is clearly a bijection.

Inductive hypothesis. We assume g^r is bijective.

Inductive step. First, we prove that g^{r+1} is surjective. Let $C_{E_{r+1}}$ be a configuration in $\mathcal{C}^{r+1}_{Adv}$. By definition of $\mathcal{C}^{r+1}_{Adv}$, there is an $r+1$-execution $E_{r+1} = (I, [G_1, G_2, \ldots, G_{r+1}])$ of Adv that ends in $C_{E_{r+1}}$. Since we are using the full-information protocol, it follows that

$$C_{E_{r+1}} = (C_{E_{r+1}}(1), C_{E_{r+1}}(2), \ldots, C_{E_{r+1}}(n))$$

where

$$C_{E_{r+1}}(i)(j) = \begin{cases} g^r(E_r)(j) & \text{if } a_j G_{r+1} a_i \\ \perp & \text{otherwise.} \end{cases}$$

Thus, $g^{r+1}(E_{r+1}) = C_{E_{r+1}}$. Therefore, g^{r+1} is surjective.

Now, we prove that g^{r+1} is injective by contraposition. Consider $E_{r+1} = (I, [G_1, G_2, \ldots, G_r, G_{r+1}])$ and $E'_{r+1} = (I', [G'_1, G'_2, \ldots, G'_r, G'_{r+1}]) \in \mathcal{E}^{r+1}_{Adv}$, such that $E_{r+1} \neq E'_{r+1}$.

If $I \neq I'$, there is an i such that $I(i) \neq I'(i)$ and trivially $g^{r+1}(E_{r+1}) \neq g^{r+1}(E'_{r+1})$ because the communication graphs are reflexive. Let us focus on the case that $I = I'$.

On the one hand, if $G_{r+1} \neq G'_{r+1}$, there is an agent a_i such that $G_{r+1}a_i \neq G'_{r+1}a_i$. Since $G_{r+1}a_i \neq G'_{r+1}a_i$, there is a j such that $g^{r+1}(E_{r+1})(i)(j) = \perp$ and $g^{r+1}(E'_{r+1})(i)(j) \neq \perp$, or $g^{r+1}(E_{r+1})(i)(j) \neq \perp$ and $g^{r+1}(E'_{r+1})(i)(j) = \perp$. Then, $g^{r+1}(E_{r+1})(i) \neq g^{r+1}(E'_{r+1})(i)$. Thus, $g^{r+1}(E_{r+1}) \neq g^{r+1}(E'_{r+1})$.

On the other hand, if $G_{r+1} = G'_{r+1}$, $g^{r+1}(E_{r+1})(i)(j) = \perp$ if and only if $g^{r+1}(E'_{r+1})(i)(j) = \perp$. Consider $E_r = (I, [G_1, G_2, \ldots, G_r])$, and $E'_r = (I, [G'_1, G'_2, \ldots, G'_r])$. $E_r, E'_r \in \mathcal{E}^r_{Adv}$ because $E_{r+1}, E'_{r+1} \in \mathcal{E}^{r+1}_{Adv}$. Since $G_{r+1} = G'_{r+1}$ and $E_{r+1} \neq E'_{r+1}$, $E_r \neq E'_r$. Since $E_r \neq E'_r$, and g^r is bijective, it follows that

$$\begin{aligned} g^r(E_r) = C_{E_r} = &\ (C_{E_r}(1), C_{E_r}(2), \ldots, C_{E_r}(n)) \\ \neq &\ (C_{E'_r}(1), C_{E'_r}(2), \ldots, C_{E'_r}(n)) \\ = &\ C_{E'_r} = g^r(E'_r). \end{aligned}$$

Then, there is an i such that $C_{E_r}(i) \neq C_{E'_r}(i)$. Then, $g^{r+1}(E_{r+1})(i)(i) \neq g^{r+1}(E'_{r+1})(i)(i)$ because G_{r+1} is reflexive. Then, $g^{r+1}(E_{r+1}) \neq g^{r+1}(E'_{r+1})$. Therefore, g^{r+1} is injective. In both cases, $g^{r+1}(E_{r+1}), g^{r+1}(E'_{r+1}) \in \mathcal{C}^{r+1}_{Adv}$ because $g^r(E_r), g^r(E'_r) \in \mathcal{C}^r_{Adv}$ and the agents communicate using the full-information protocol. Since g^{r+1} is surjective and injective, g^{r+1} is bijective. □

Consider $w_r = (\ldots((I, G_1), G_2)\ldots, G_r) \in W^r$. We define $h^r : W^r \to \mathcal{E}^r_{Adv}$ as follows:

$$h^r(w_r) = (I, [G_1, G_2, \ldots, G_r]).$$

Lemma 3.4.4. *h^r is a bijection.*

Proof. We will prove that h^r is bijective by induction on the number r of rounds.

Base case. Since the worlds in W^0 and the executions in $\mathcal{E}^0_{Adv}$ can be identified with their respective input vector, h^0 is clearly a bijection.

Inductive hypothesis. We assume that h^r is a bijection.

Inductive step. First, we prove by contraposition that h^{r+1} is injective. If $w_{r+1} \neq w'_{r+1}$, $I \neq I'$ or there is $1 \leq i \leq r+1$ so that $G_i \neq G'_i$. Then, $h^{r+1}(w_{r+1}) \neq h^{r+1}(w'_{r+1})$. Thus h^{r+1} is injective.

Second, we prove that h^{r+1} is surjective. Consider $E_{r+1} = (I, [G_1, G_2, \ldots, G_r, G_{r+1}]) \in \mathcal{E}^{r+1}_{Adv}$, and $E_r = (I, [G_1, G_2, \ldots, G_r]) \in \mathcal{E}^{r}_{Adv}$. Since h^r is bijective, there is a world $w_{E_r} = ((\ldots((I, G_1), G_2), \ldots), G_r)$ such that $h^r(w_{E_r}) = E_r$. $W^{r+1} = \{(w_r, G_{r+1}) \in W^r \times \mathbf{G}^{r+1} \mid M, w_r \models Pre(G_{r+1})\}$ by definition of $\odot$. By construction of Pre^{r+1}, $\delta_{w_{E_r}}$ is a disjunct of $Pre(G_{r+1})$ because $h^r(w_{E_r}) \in \mathcal{E}^r_{G_{r+1}}$. Since $\delta_{w_{E_r}}$ is a formula that identifies w_{E_r}, it follows that $w_{E_{r+1}} = (w_{E_r}, G_{r+1}) \in W^{r+1}$. Moreover, $h^{r+1}(w_{E_{r+1}}) = E_{r+1}$. Thus, h^{r+1} is surjective. Therefore h^{r+1} is bijective. □

Now, we start with the proof of Theorem 3.4.2.

Proof. We define

$$f^r : W^r \to \mathcal{C}^r = g^r \circ h^r.$$

Since g^r and h^r are bijective, f^r is bijective.

Now we prove, by induction on the number r of rounds, that the epistemic model M^r reflects indistinguishability between configurations.

Base case.

Consider $I, I' \in W^0$, $C_I = f^0(I) = (I(1), I(2), \ldots, I(n))$, and $C_{I'} = f^0(I') = (I'(1), I'(2), \ldots, I'(n))$. By construction of $\sim^0$, $I \sim^0_{a_i} I'$ if and only if $I(i) = I'(i)$ holds. Since C_I and $C_{I'}$ are indistinguishable for a_i if and only if $I(i) = I'(i)$ holds, $I \sim^0_{a_i} I'$ if and only if C_I and $C_{I'}$ are indistinguishable for a_i.

Inductive hypothesis.

Consider $M^r = (W^r, \sim^r, L^r) = M^0 \odot \mathcal{P}^1 \odot \mathcal{P}^2 \odot \cdots \odot \mathcal{P}^r$. We assume that for all $w_r, w'_r \in W^r$, $f_r : W^r \to \mathcal{C}^r_{Adv}$ satisfies that $w_r \sim^r_{a_i} w'_r$ if and only if $f^r(w_r)$ and $f^r(w'_r)$ are indistinguishable for a_i.

Inductive step.

Consider $w_{r+1} = (w_r, G_{r+1}), w'_{r+1} = (w'_r, G'_{r+1}) \in W^{r+1}$. We need to prove that $w_{r+1} \sim^{r+1}_{a_i} w'_{r+1}$ if and only if $f^{r+1}(w_{r+1})$ and $f_{r+1}(w'_{r+1})$ are indistinguishable for a_i.

By definition of f^{r+1}:

$$f^{r+1}(w_{r+1}) = C_{r+1} = (C_{r+1}(1), C_{r+1}(2), \ldots, C_{r+1}(n))$$

where

$$C_{r+1}(i)(j) = \begin{cases} f^r(w_r)(j) & \text{if } a_j G_{r+1} a_i \\ \bot & \text{otherwise} \end{cases}$$

and

$$f^{r+1}(w'_{r+1}) = C'_{r+1} = (C'_{r+1}(1), C'_{r+1}(2), \ldots, C'_{r+1}(n))$$

where

$$C'_{r+1}(i)(j) = \begin{cases} f^r(w'_r)(j) & \text{if } a_j G'_{r+1} a_i \\ \bot & \text{otherwise} \end{cases}.$$

$w_{r+1} \sim^{r+1}_{a_i} w'_{r+1}$
$\Leftrightarrow$
$G_{r+1}a_i = G'_{r+1}a_i$, and $w_r \sim^r_{G_{r+1}a_i} w'_r$ (by definition of $\odot$)
$\Leftrightarrow$
$G_{r+1}a_i = G'_{r+1}a_i$, and $w_r \sim^r_{a_j} w'_r \; \forall a_j \in G_{r+1}a_i$ (by definition of $\sim_B$)
$\Leftrightarrow$
$G_{r+1}a_i = G'_{r+1}a_i$, and $f^r(w_r)$ and $f^r(w'_r)$ are indistinguishable $\forall a_j \in G_{r+1}a_i$ (by inductive hypothesis)
$\Leftrightarrow$
$C_{r+1}(i)(j) = C'_{r+1}(i)(j) \forall 1 \leq j \leq n$ (if $a_j \notin G_{r+1}a_i$, $C_{r+1}(i)(j) = \bot = C'_{r+1}(i)(j)$), otherwise, $C'_{r+1}(i)(j) = f^r(w_r)(j) = f^r(w'_r)(j) = C'_{r+1}(i)(j)$)
$\Leftrightarrow$
$C_{r+1}(i) = C'_{r+1}(i)$
$\Leftrightarrow$
C_{r+1} and C'_{r+1} are indistinguishable for a_i. □

Corollary 3.4.4.1 (Constant space)**.** *Let Adv be an oblivious adversary. There is an infinite sequence* $\mathcal{P}^1 = (\mathbf{G}^1, Pre^1), \mathcal{P}^2 = (\mathbf{G}^2, Pre^2), \ldots$ *of pattern models that reflects Adv so that for the function* $\sigma : \mathbb{N} \to \mathbb{N}$ *defined as* $\sigma(i) = |\mathbf{G}^i| + \sum_{G \in \mathbf{G}^i} |Pre^i(G)|$ *it holds that* $\sigma \in \Theta(1)$.

Proof. Let $\mathcal{P}$ be the pattern model for *Adv* built as described in Sect. 3.4. In this case, $\mathcal{P} = \mathcal{P}^1 = \mathcal{P}^2 = \ldots$. By Theorem 3.4.2, $\mathcal{P}$ reflects *Adv*. Since $\mathcal{P} = \mathcal{P}^1 = \mathcal{P}^2 = \ldots$, then $\sigma(1) = \sigma(2) = \ldots$ Therefore $\sigma \in \Theta(1)$. □

3.5 Pattern models and action models are incomparable in update expressivity

In this section, we prove that, despite the fact that action model logic and pattern model logic have the same expressivity of the epistemic logic, such logics are incomparable in update expressivity. First, we prove Prop. 3.5.0.1,

which states that there is an epistemic model whose update with an action model cannot be obtained with any pattern model. Then, we prove Prop. 3.5.0.2, which states that there is an epistemic model whose update with a pattern model cannot be obtained with any action model. Finally, we present a workaround that allow us for obtaining the same update effect that of action models using pattern models machinery.

Given an arbitrary action model $\mathsf{M} = (\mathsf{E}, \sim, \mathsf{Pre})$, and an arbitrary pattern model $\mathcal{P} = (\mathbf{G}, Pre)$, we define $[\mathsf{M}]\varphi := \bigwedge_{e \in E} [\mathsf{M}, e]\varphi$ and $[\mathcal{P}]\varphi := \bigwedge_{G \in \mathbf{G}} [\mathcal{P}, G]\varphi$.

Proposition 3.5.0.1. *Communication pattern logic is not at least as update expressive as action model logic.*

Proof. We can show that there are action models that produce updated epistemic models that we cannot obtain using pattern models. Here, we should note that the composition of two action models is again an action model i.e., for all M, M' there is a M'', namely the composition of M and M', such that $[\mathsf{M}][\mathsf{M}']\varphi \leftrightarrow [\mathsf{M}'']\varphi$. Sequentially executing two pattern models is typically not the same as executing a single pattern model, i.e., it is not the case that for all $\mathcal{P}, \mathcal{P}'$ there is a $\mathcal{P}''$ such that $[\mathcal{P}][\mathcal{P}']\varphi \leftrightarrow [\mathcal{P}'']\varphi$. For example, consider the models M^0, in Fig. 2.1, $\mathcal{P}_{\text{two-IIS}}$, in Fig. 3.2, and $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$, that is structurally isomorphic to a 36 node ring. The domain of model M^0 consists of four worlds and that of $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$ consists of 36 worlds; it is nine times larger and it is bisimulation minimal. Now, there are only four different communication graphs for two agents. So the maximum size of a model resulting from updating M^0 with a pattern model is 16. Therefore there is no such pattern model. In other words, there is no $\mathcal{P}$ such that $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$ is bisimilar to $M^0 \odot \mathcal{P}$ which implies that there is no $\mathcal{P}$ that has the same update effect as updating twice with $\mathcal{P}_{\text{two-IIS}}$.

However, there is an action model M such that $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$ is bisimilar to $M^0 \otimes \mathsf{M}$: its domain is the domain of $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$; its relations for a and b are the relations for a and b on the model $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$, and its preconditions are such that the precondition of a world $((I, G), G')$ in the domain of $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$ is τ_w (See Sect. 2.7). It is straightforward to see that $M^0 \odot \mathcal{P}_{\text{two-IIS}} \odot \mathcal{P}_{\text{two-IIS}}$ is even isomorphic to $M^0 \otimes \mathsf{M}$.

We conclude that there is no pattern model that is update equivalent to this action model M. Therefore, pattern model logic is not at least as update

expressive as action model logic. □

We continue by showing that action model logic is not at least as update expressive as pattern model logic. We prove this in a more meaningful way in the following Prop. 3.5.0.2. Its proof assumes towards a contradiction that an action model M exists that is update equivalent to the pattern model $\mathcal{P}_{\text{two-IIS}}$, where we identify M with the multi-pointed action model $(\mathsf{M}, \mathcal{D}(\mathsf{M}))$. We then compare the updates $\mathcal{P}_{\text{two-IIS}}$ and M in epistemic model $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}$ for n exceeding a function of the modal depth of any precondition of M, and derive a contradiction.

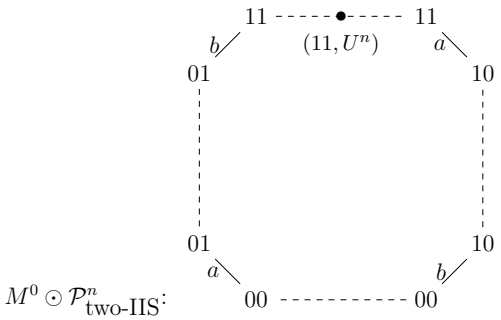
Proposition 3.5.0.2. *Action model logic is not at least as update expressive as pattern model logic.*

Proof. Suppose towards a contradiction that pattern model $\mathcal{P}_{\text{two-IIS}}$ is update equivalent to an action model $\mathsf{M} = (E, \sim, \mathsf{pre})$.

What do we know about M? As $\mathcal{P}_{\text{two-IIS}}$ is always executable, we may assume that the disjunction ψ of all preconditions of actions e in the domain E of M is the triviality. Otherwise, given some model with $M, w \models \neg\psi$, we could update with $\mathcal{P}_{\text{two-IIS}}$ but not with M. Similarly, for any action e in the domain E of M, there must be $f \in E$ such that $e \sim_a f$ and $\mathsf{pre}(e) = \mathsf{pre}(f)$ (and for agent b there must be a $g \in E$ such that $g \sim_b f$ and $\mathsf{pre}(g) = \mathsf{pre}(f)$). Otherwise, consider a model (M, w) that can only be updated with (M, e) (for which $M, w \models \mathsf{pre}(e)$). It can be updated with $(\mathcal{P}_{\text{two-IIS}}, G^{ab})$ and also with $(\mathcal{P}_{\text{two-IIS}}, G^{b.a})$ resulting in states (w, G^{ab}) and $(w, G^{b.a})$ satisfying different properties, as $(w, G^{ab}) \sim_a (w, G^{b.a})$ (because $G^{ab}a = G^{ba} = \{a, b\}$), so that one or the other but not both can be bisimilar to (w, e). Therefore, M must be a refinement of $\mathcal{P}_{\text{two-IIS}}$ seen as a structure $G^{a.b}$—b—G^{ab}—a—$G^{b.a}$. Its actions can therefore be assumed to have shape (G, φ) where G is one of $G^{a.b}, G^{ab}, G^{b.a}$ and where $\varphi \in \mathcal{L}_\otimes$ is the precondition of that action, that is, $\mathsf{pre}(G, \varphi) = \varphi$.[3]

The modality $[\mathsf{M}]$ is an operator in the language $\mathcal{L}_\otimes$ and $|E|$ is finite, so that $md(\mathsf{M}) = \max\{md(\mathsf{pre}(e)) \mid e \in E\}$ is defined. Choose $n \in \mathbb{N}$ with $n > \log_3 2(md(\mathsf{M}) + 1)$. This bound will be justified in the next paragraph. Consider $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}$, schematically depicted as:

[3]By *refinement* we mean that $G^{a.b}$ can be seen as an equivalence class $\{(G^{a.b}, \varphi) \mid (G^{a.b}, \varphi) \in \mathcal{D}(\mathsf{M})\}$, and similarly for G^{ab} and $G^{b.a}$, where two such equivalence classes are indistinguishable for a if there are $(G, \varphi), (G', \varphi')$ such that $(G, \varphi) \sim_a (G', \varphi')$, and similarly for b.



and concretely its three-world fragment:

$$(*):\ (11, G^{ab^{n-1}}G^{b.a}) \overset{a}{—} (11, G^{ab^n}) \overset{b}{—} (11, G^{ab^{n-1}}G^{a.b})$$

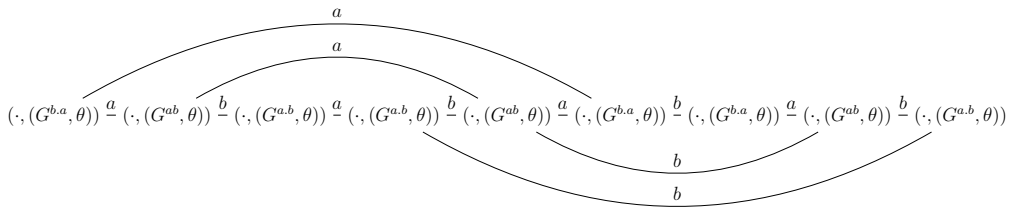
where world $(11, G^{ab^n})$ of $(*)$ is the same as the depicted world $(11, G^{ab^n})$ of $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}$.

We can now justify the bound $n > \log_3 2(md(\mathsf{M}) + 1)$. We need in the proof that the three worlds of $(*)$ satisfy the same actions of M, and we guarantee that because they are bounded collectively bisimilar for an appropriate bound. Given $(11, G^{ab^n})$, the bound should exceed the modal depth of any possible precondition of any action in M, which explains $md(\mathsf{M})$. Plus one, as we need this to hold for the surrounding worlds too, which explains $md(\mathsf{M}) + 1$. Twice that, $2 \cdot (md(\mathsf{M}) + 1)$, is the required length of one side of the squarish model $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}$ with therefore $8 \cdot (md(\mathsf{M}) + 1)$ worlds. Starting with four worlds, every iteration of $\mathcal{P}_{\text{two-IIS}}$ multiplies the number of worlds by 3. So we therefore want to iterate $\mathcal{P}_{\text{two-IIS}}$ by some n such that $4 \cdot 3^n > 8 \cdot (md(\mathsf{M}) + 1)$, that is, $n > \log_3 2(md(\mathsf{M}) + 1)$.

Consider $M^0 \odot \mathcal{P}^n_{\text{two-IIS}} \otimes \mathsf{M}$. Recalling what is known about M, there must be an $e \in E$ such that $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}, (11, G^{ab^n}) \models \mathsf{pre}(e)$. Also, there must be $f, g \in E$ with $e \sim_a f$ and $f \sim_b g$ and $\mathsf{pre}(e) = \mathsf{pre}(f) = \mathsf{pre}(e)$. Let $\mathsf{pre}(e)$ be θ. These actions e, f, g therefore have shape $(G^{a.b}, \theta)$, (G^{ab}, θ), $(G^{b.a}, \theta)$ respectively.

As $n > \log_3 2(md(\mathsf{M})+1)$, the three worlds in $(*)$ are bounded collectively bisimilar: $(M^0 \odot \mathcal{P}^n_{\text{two-IIS}}, (11, G^{ab^{n-1}}, G^{b.a})) \leftrightarrows^{md(\mathsf{M})+1} (M^0 \odot \mathcal{P}^n_{\text{two-IIS}}, (11, G^{ab^n})) \leftrightarrows^{md(\mathsf{M})+1} (M^0 \odot \mathcal{P}^n_{\text{two-IIS}}, (11, G^{ab^{n-1}}, G^{ab}))$. As $md(\theta) \leq md(\mathsf{M})$, all three worlds in $(*)$ satisfy θ, so actions e, f, g can be executed in all these worlds.

The model $M^0 \odot \mathcal{P}^n_{\text{two-IIS}} \otimes \mathsf{M}$ therefore contains the submodel



wherein only some additional pairs for $\sim_a$ and $\sim_b$ are shown, and where from those shown we merely justify one as an example: for the leftmost and the middle worlds, we have that $(11, G^{ab^{n-1}}, G^{b.a}, (G^{b.a}, \theta)) \sim_a (11, G^{ab^n}, (G^{ab}, \theta))$, because by the semantics of action model execution, $(11, G^{ab^{n-1}}, G^{b.a}) \sim_a (11, G^{ab^n})$ in $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}$ and $(G^{b.a}, \theta) \sim_a (G^{ab}, \theta)$ in M. Furthermore, worlds $(\ldots, (G, \theta))$ shown, may be indistinguishable for a or b from worlds $(\ldots, (G, \xi))$ not shown, for actions (G, ξ) with ξ non-equivalent to θ.

Consequently, $M^0 \odot \mathcal{P}^n_{\text{two-IIS}} \otimes \mathsf{M}$ is not a circular ab-chain like $M^0 \odot \mathcal{P}^{n+1}_{\text{two-IIS}}$ that locally looks like:

$(\cdot, G^{b.a})$ $\overset{a}{---}$ $(\cdot, G^{ab})$ $\overset{b}{---}$ $(\cdot, G^{a.b})$ $\overset{a}{---}$ $(\cdot, G^{a.b})$ $\overset{b}{---}$ $(\cdot, G^{ab})$ $\overset{a}{---}$ $(\cdot, G^{b.a})$ $\overset{b}{---}$ $(\cdot, G^{b.a})$ $\overset{a}{---}$ $(\cdot, G^{ab})$ $\overset{b}{---}$ $(\cdot, G^{a.b})$

The assumption of update equivalence implies that $M^0 \odot \mathcal{P}^{n+1}_{\text{two-IIS}}$ is collectively bisimilar to $M^0 \odot \mathcal{P}^n_{\text{two-IIS}} \otimes \mathsf{M}$. The supposed bisimulation relation Z between the domain of $M^0 \odot \mathcal{P}^{n+1}_{\text{two-IIS}}$ and the domain of $M^0 \odot \mathcal{P}^n_{\text{two-IIS}} \otimes \mathsf{M}$ should be such that $((w, \sigma, G), (w, \sigma, (G, \mathsf{pre}(e)))) \in Z$ for all $w \in W$, $\sigma \in \mathbf{G}^n$, and $e \in E$ with $M^0 \odot \mathcal{P}^n_{\text{two-IIS}}, (w, \sigma) \models \mathsf{pre}(e)$, in particular the three worlds in $(*)$ and the e, f, g above with preconditions θ. On the other hand, a pair of worlds in this relation cannot be bisimilar, as the additional a-links and b-links allow shorter paths to a 01-world or a 10-world. In other words, as bounded bisimilarity implies the same truth value for formulas of at most that modal depth, the worlds in such a pair satisfy different formulas.

This contradicts the assumption that M is update equivalent to $\mathcal{P}_{\text{two-IIS}}$ and thus concludes the proof. □

Corollary 3.5.0.1. *Communication pattern logic and action model logic are incomparable in update expressivity.*

3.5.1 Simulation of action model with pattern models

An interesting point consists in answering whether we can simulate updating an epistemic model with an action model using the pattern model machinery.

The only way to avoid the restriction on the number of communication graphs mentioned above is adding agents so that we could design at least the same number of graphs as the number of events. Adding agents, however, is not the complete answer because we will need a procedure to transform an epistemic model M on the original set of agents A into an epistemic model on a set of agents $A' \supseteq A$. We will end with epistemic models that considers a different set of agents and thus different formulas. To get back to the original set of agents we will need an extra procedure to get back to the original set of agents.

In summary, we will be able to simulate the update of an epistemic model M with action model M as follows. First, we will transform M on the set of agents A into M' on a set of agents $A' \supseteq A$. Second, we will build a pattern model $\mathcal{P}_{\mathsf{M}}$, on the set of agents A', so that $M' \odot \mathcal{P}_{\mathsf{M}}$ preserves the indistinguishability relations of the agents in $M \otimes \mathsf{M}$. Third, we will transform $M' \odot \mathcal{P}_{\mathsf{M}}$ into an epistemic model collectively bisimilar to $M \otimes \mathsf{M}$.

Adding agents. Let an epistemic model $M = (W, \sim, L)$ and an action model $\mathsf{M} = (\mathsf{E}, \sim, \mathsf{Pre})$ be given. We will assume without lose of generality that $A \cap \mathsf{E} = \emptyset$. We build $M' = (W, \sim', L)$ over $A \cup \mathsf{E}$, and P such that:

$$\sim'_a = \begin{cases} \sim_a & \text{if } a \in A \\ W \times W & \text{if } a \in \mathsf{E} \end{cases}$$

Building $\mathcal{P}_{\mathsf{M}}$. We will build $\mathcal{P}_{\mathsf{M}} = (\mathbf{G}_{\mathsf{M}}, Pre_{\mathsf{M}})$ such that:

$$\mathbf{G}_{\mathsf{M}} = \{G_{\mathsf{e}} \in \mathbf{G}_{A \cup \mathsf{E}} \mid \mathsf{e} \in \mathsf{E},\ \forall a \in A,\ \forall \mathsf{e}' \in \mathsf{E}\ G_{\mathsf{e}} a = \{a\} \cup \{\mathsf{e}' \in \mathsf{E} \mid \mathsf{e}' \in [\mathsf{e}]_a\}$$
$$\text{and } G_{\mathsf{e}} \mathsf{e}' = \{\mathsf{e}'\}\}$$

where $[\mathsf{e}]_a$ is the equivalence class of a with respect to $\sim_a$

$$Pre_{\mathsf{M}}(G_{\mathsf{e}}) = \mathsf{Pre}(\mathsf{e})$$

Removing added agents. In order to delete the added agents, we define projections on epistemic models on a sets of agents. Let $M = (W, \sim, L)$, an epistemic model over a set of agents A' and a set of propositions P, and a set of agents $A \subseteq A'$ be given. The projection of M on A, $M|_A = (W, \sim|_A, L)$, is defined so that

$$(\sim |_A)_a = \sim_a \forall a \in A$$

.

Claim 3.5.1. *Let us consider $M \otimes \mathsf{M}$ and $M' \odot \mathcal{P}_{\mathsf{M}}$ as described above. For all $a \in A$, $(w, e) \sim_a (w', e')$ iff $(w, G_{\mathsf{e}}) \sim'_a (w', G_{\mathsf{e}'})$*

Proof. Abusing notation, we will denote the indistinguishability relations of agent a in M and $M \otimes \mathsf{M}$ as $\sim_a$ and in M' and $M' \odot \mathcal{P}_{\mathsf{M}}$ as $\sim'_a$
$(w, \mathsf{e}) \sim_a (w', \mathsf{e}')$
$\Leftrightarrow$
$w \sim_a w'$ and $\mathsf{e} \sim_a \mathsf{e}'$
$\Leftrightarrow$
$\mathsf{e} \sim_a \mathsf{e}'$ and $w \sim_a w'$
$\Leftrightarrow$
$[\mathsf{e}]_a = [\mathsf{e}']_a$ and $w \sim_a w'$
$\Leftrightarrow$
$G_{\mathsf{e}}a = \{a\} \cup \{\mathsf{e}' \in \mathsf{E} \mid \mathsf{e}' \in [\mathsf{e}]_a\} = G_{\mathsf{e}'}a$ and $w \sim_a w'$
$\Leftrightarrow$ $\forall \mathsf{e} \in \mathsf{E} \sim_e = W \times W$ by construction of M'
$G_{\mathsf{e}}a = G_{\mathsf{e}'}a$ and $w \sim_{G_{\mathsf{e}}a} w'$
$\Leftrightarrow$
$(w, G_{\mathsf{e}}) \sim'_a (w', G_{\mathsf{e}'})$ □

Proposition 3.5.1.1 (Simulation of action models with pattern models)**.** *Let $M = (W, \sim, L)$ and $\mathsf{M} = (\mathsf{E}, \sim, \mathsf{Pre})$ be given, and M' built as above. $M \otimes \mathsf{M}$ and $(M' \odot \mathcal{P}_{\mathsf{M}})|_A$ are collective bisimilar.*

Proof. First, let us note that for every world (w, e) of $M \otimes \mathsf{M}$ there is a world (w, G_{e}) in $(M' \odot \mathcal{P}_{\mathsf{M}})|_A$ because $\mathsf{Pre}(\mathsf{e}) = Pre(G_{\mathsf{e}})$, the indistinguishability relations $\sim_a = \sim'_a$ for all $a \in A$, and all precondition formulas of the events are defined in M'.

Let us define $Z : W \to W'$ as $Z((w, \mathsf{e})) = (w, G_{\mathsf{e}})$. Let us note that Z is a bijective function.

Since Z is a bijection and by claim 3.5.1, it follows that $M \otimes \mathsf{M}$ and $(M' \odot \mathcal{P}_{\mathsf{M}})|_A$ are collective bisimilar. □

Although this simulation is possible in one way, we may conjecture that something similar cannot happen in the opposite direction because of the size finiteness of the precondition formulas. This is not proved, however.

After presenting pattern models and comparing its update expressivity with action models in this chapter, we present, in Ch. 4, a sufficient condition that preserves connectivity in the epistemic models through all execution. This condition implies impossibility of solving consensus for a given dynamic network model.

Chapter 4

Impossibility of Consensus

Having defined pattern models in Ch. 3, this chapter uses Theorem 3.4.2 to derive an impossibility condition for the fundamental consensus problem introduced by Pease, Shostak, and Lamport [34]. The condition is based on the known connection between agreement and common knowledge by Halpern and Moses [25]. Roughly speaking, they showed that for the agents of a distributed system to decide on the same input v, there must be common knowledge on v. The impossibility condition uses the fact that common knowledge is closely related to connectivity: roughly speaking, there cannot be common knowledge on v in an epistemic model if there is a world where an agent does not know v and the model is connected. The condition that appears below identifies properties of oblivious adversaries that preserve connectivity, hence precluding the solvability of consensus. Since we are interested in computability, it is enough to focus on the case of the full-information protocol (see for example [4, 26]).

4.1 Edges, Paths and Connectivity

Let $M = (W, \sim, L)$ be an epistemic model. We say that M is *connected* if for every pair of worlds $w, w' \in W$, $w \overset{\cup}{\sim}{}^{*}_{A} w'$, i.e., (w, w') is in the reflexive transitive closure of $\overset{\cup}{\sim}_A$. It will be convenient for our discussion below to equivalently define connectivity using the following notion. For every pair of worlds $w, w' \in W$, (w, w') is an *edge* of M if there exists $a \in A$ such that $w \sim_a w'$. A *path* $(w = w_1, w_2, \ldots, w_m = w')$ in M between worlds $w, w' \in W$ is a sequence of worlds such that (w_i, w_{i+1}) is an edge of M, with $m \geq 2$. We say

that the path *passes* through (w_k, w_{k+1}), for each $1 \leq k \leq m-1$. Given an edge set E of M, we say that M is *E-connected* if for every $w, w' \in W$ there is a path between the worlds that passes only through edges of E (thus E must span all worlds of M). Note that if M is E-connected for some E, then it is connected.

Let $\mathcal{P} = (\mathbf{G}, Pre)$ be an oblivious pattern model. Recall that oblivious implies that $Pre(G) = \top$, for every communication graph $G \in \mathbf{G}$. We define the relation $G \sim_a G'$ whenever $Ga = G'a$, with $G, G' \in \mathbf{G}$ and $a \in A$. We say that $\mathcal{P}$ is *connected* if $G \overset{\cup}{\sim}{}^*_A G'$, for every pair or graphs $G, G' \in \mathbf{G}$. Again, we alternatively define connectivity as follows. For every pair of graphs $G, G' \in \mathbf{G}$, (G, G') is an *edge* of $\mathcal{P}$ if there exists $a \in A$ such that $G \sim_a G'$. A *path* $(G = G_1, G_2, \ldots, G_m = G')$ in $\mathcal{P}$ between two graphs $G, G' \in \mathcal{P}$ is a sequence of graphs such that (G_i, G_{i+1}) is an edge of $\mathcal{P}$, with $m \geq 2$. We say that the path *passes* through (G_k, G_{k+1}), for each $1 \leq k \leq m-1$. Given an edge set E of $\mathcal{P}$, we say that $\mathcal{P}$ is *E-connected* if for every $G, G' \in \mathbf{G}$ there is a path between the graphs that passes only through edges of E (hence E must span all graphs of $\mathcal{P}$). Again, if $\mathcal{P}$ is E-connected for some E, then it is connected.

4.2 Consensus and connectivity

Informally, the consensus problem [34] requires each agent of a distributed system to irrevocably decide a value among the inputs of the agents, with the constraint that all decided values must be the same. We consider the following definition of the problem that suits better the failure-free distributed systems studied here.

Recall that $\mathcal{I}$ denotes the finite set of possible inputs of the agents and $\perp$ is a *default* value not in $\mathcal{I}$. In the definition of consensus below, the agents make irrevocable decisions. This is formally captured with a *decision* function d from the set of local agent states of the full-information protocol to the set $\mathcal{I} \cup \{\perp\}$. Intuitively, if an agent is in a local state s and $d(s) \neq \perp$, then $d(s)$ is its decision and that decision never changes. Formally, any decision function d must satisfy the following *irrevocability* property: in every execution, if an agent a is in state s at the end of round $r \geq 0$ and $d(s) \neq \perp$, then at the end of every round $r' \geq r$, $d(s') = d(s)$, where s' is the state of a at the end of round r'.

Recall that each agent a starts every execution with a private input value

from the input set $\mathcal{I}$. We say that a decision function d *solves* the consensus problem *against* adversary Adv, if the following three properties are satisfied in every execution of the adversary Adv:

- **Termination**. For every agent a, there is a finite $r \geq 0$ such that $d(s) \neq \perp$, where s is the state of a at the end of round r.[1]

- **Validity**. For every agent a, if there is a round $r \geq 0$ such that $d(s) \neq \perp$, where s is the state of a at the end of round r, then $d(s)$ is the input of some agent b in that execution.

- **Agreement**. For any pair of agents a, b, if there are rounds $r_a, r_b \geq 0$ such that $d(s_a), d(s_b) \neq \perp$, where s_a and s_b are the states of a and b at the end of rounds r_a and r_b, respectively, then $d(s_a) = d(s_b)$.

The solvability of consensus thus involves determining if there is such a decision function for a given adversary. If there is no such function, we say that consensus is impossible against Adv. As anticipated, the consensus impossibility condition is related to connectivity of the epistemic model of the system after a number of rounds. The relation is the following.

Suppose that there is a decision function d that solves the consensus problem against an oblivious adversary Adv. We start by observing that since $\mathcal{I}$ is finite, there are finitely many input vectors, and hence there is a finite $r \geq 0$ such that every agent makes a decision at the end of round r in every execution, as d satisfies **Termination**. Therefore, for the configuration C at the end of any r-execution of Adv, $d(C(a)) \neq \perp$, for every agent a.

In the rest of the section, we consider only configurations of $\mathcal{C}^r_{Adv}$. We make the following observations about d and any configuration C:

- Since d satisfies **Validity**, for each agent a, $d(C(a))$ is the input of some agent b in the r-execution that ends at C.

- Since d satisfies **Agreement**, for each pair of agents a, b, $d(C(a)) = d(C(b))$. We say that $d(C(a))$ is the *decision* at configuration C, and is denoted $d(C)$.

[1] We will see later that this property implies a stronger property stating that there is a finite $r \geq 0$ such that every agent decides at time r, at the latest, in *every execution*.

- For any configuration C' such that C and C' are indistinguishable for some agent a, i.e., $C(a) = C'(a)$, we have that $d(C(a)) = d(C'(a))$. Moreover, by the previous observation, for any agent b, $d(C(a)) = d(C'(b))$. Namely, the decisions at C and C' are the same.

Combining the three observations we obtain:

Claim 4.2.1. *Suppose that for configurations $C_1, \ldots, C_m$, $m \geq 2$, there are agents $a_1, \ldots, a_{m-1}$ such that $C_k(a_k) = C_{k+1}(a_k)$, $1 \leq k \leq m-1$. Then, $d(C_1)$ is the input of some agent in the r-execution that ends at C_m.*

Consider now the epistemic model $M^r = (W^r, \sim^r, L^r)$ obtained by updating r times M^0 with $\mathcal{P}$, using the restricted modal product $\odot$, where M^0 is the initial epistemic model and $\mathcal{P}$ is the oblivious communication pattern obtained from Adv.

By Theorem 3.4.2, there is a bijection $f : W^r \to \mathcal{C}^r$ such that, for every agent a and worlds w, w', $w \sim_a^r w'$ if and only if $f(w)(a) = f(w')(a)$. Moreover, the proof of the theorem shows that there is such a bijection f additionally satisfying that, for every world w and every input v, $v_a \in L^r(w)$ if and only if v is the input of agent a in the r-execution that ends at configuration $f(w)$. Let us consider such a bijection f.

Consider any world w, and let $v = d(f(w))$. Consider any world w' such that w is connected with w', and let $(w = w_1, w_2, \ldots w_m = w')$, with $m \geq 2$, be any path between w and w'. From the properties of f, we conclude that there are agents $a_1, \ldots, a_{m-1}$ such that the configurations $f(w_1), \ldots, f(w_m)$ and agents $a_1, \ldots, a_{m-1}$ satisfy the properties stated in the hypothesis of Claim 4.2.1, i.e., $f(w_k)(a_k) = f(w_{k+1})(a_k)$, $1 \leq k \leq m-1$. By Claim 4.2.1, v is the input of some agent in the r-execution that ends at $f(w_m)$, and hence, by the properties of f, $v_b \in L^r(w_m)$. Thus, we have that $M^r, w_m \models \psi_v$, where ψ_v is the formula $v_{a_1} \vee v_{a_2} \vee \ldots \vee v_{a_n}$, (intuitively, ψ_v indicates that some agent has input v). Hence, formula ψ_v is true at every world that is reachable from w, which gives the following connection between consensus solvability and common knowledge:

Claim 4.2.2. *For any world w, $M^r, w \models C_A \psi_v$, where $v = d(f(w))$.*

Suppose now that M^r is connected. Then, for every world w and every input value v, there is a world w' such that $v_a \in L^r(w)$ for every agent a (intuitively, in w' all agents have input v), and $w \, (\overset{\cup}{\sim}{}_A^r)^* \, w'$. Thus, $M^r, w \not\models C_A \psi_v$.

We stress that the election of w and d is arbitrary. By Claim 4.2.2, d cannot solve consensus in round r. Therefore, we have the following connection between connectivity and impossibility of consensus:

Lemma 4.2.3 (Impossibility of consensus and connectivity)**.** *Consider any oblivious adversary Adv and let $\mathcal{P} = (\mathbf{G}, Pre)$ be the communication pattern obtained from Adv. If $|\mathcal{I}| > 1$ and for every $r \geq 0$, $M^r = M^{r-1} \odot \mathcal{P}$ is connected, then consensus is impossible against Adv.*

4.3 A Sufficient Condition that Preserves Connectivity

In light of Lemma 4.2.3, we now turn our attention to finding a condition for oblivious communication patterns for preserving connectivity after any number of communication rounds. It is clear that for M^{r+1} to be connected, M^r and $\mathcal{P}$ must be connected; it is not sufficient, however. It is also required that M^r and $\mathcal{P}$ share certain properties. Below we identify one such condition. Intuitively, it says that after any number of rounds r, there is an edge set of M^r that preserves the uncertainty in the initial epistemic model M^0 and the communication pattern $\mathcal{P}$, which in turn will imply that M^{r+1} remains connected.

Definition 4.3.1 (Strong edges)**.** *Consider any epistemic model $M = (W, \sim, L)$ and any oblivious communication pattern $\mathcal{P} = (\mathbf{G}, Pre)$. Let $w, w' \in W$, and $G, G' \in \mathbf{G}$.*

1. *Let $B = \{a \in A \mid w \sim_a w'\}$. An edge (w, w') of M (possibly with $w = w'$) is* strong with respect to $\mathcal{P}$*, or just* strong *when $\mathcal{P}$ is clear from the context, if there is a graph $G \in \mathbf{G}$ and a set $\emptyset \neq C \subseteq B$ such that $GC = \bigcup_{a \in C} Ga = C$. An edge set E of M is* strong with respect to $\mathcal{P}$*, or just* strong*, if each edge of E is strong and M is E-connected. The model M is* strong with respect to $\mathcal{P}$*, or just* strong*, if it has a strong edge set.*

2. *Let $B = \{a \in A \mid G \sim_a G'\}$. An edge (G, G') of $\mathcal{P}$ (possibly with $G = G'$) is* strong *if there is a graph $H \in \mathbf{G}$ and a set $\emptyset \neq C \subseteq B$ such that $HC = \bigcup_{a \in C} Ha = C$. An edge set E of $\mathcal{P}$ is* strong *if each edge of E is strong and $\mathcal{P}$ is E-connected. The pattern model $\mathcal{P}$ is* strong *if it has a strong edge set.*

Claim 4.3.1 (Preserving strong edges)**.** *Consider any epistemic model $M = (W, \sim, L)$ and any oblivious communication pattern $\mathcal{P} = (\mathbf{G}, Pre)$. The restricted modal product $\odot$ preserves strong edges:*

1. *For any strong edge (w, w') of M, there is a graph $G \in \mathbf{G}$ such that $((w, G), (w', G))$ is an edge of the product $M \odot \mathcal{P}$ and is strong.*

2. *For any strong edge (G, G') of $\mathcal{P}$, for* every *world $w \in W$, $((w, G), (w, G'))$ is an edge of the product $M \odot \mathcal{P}$ and is strong.*

Proof. Let $M \odot \mathcal{P} = (W', \sim', L')$.

For the first case, consider any graph G and set C as stated in Definition 4.3.1 (1). Since $\mathcal{P}$ is oblivious, we have that $Pre(G) = \top$, hence $(w, G), (w', G) \in W'$. Observe that, for every $a \in C$, $Ga \subseteq GC = C \subseteq B$, then, $w \sim_{Ga} w'$. Thus, $(w, G) \sim'_a (w', G)$ because $w \sim_{Ga} w'$ and obviously $Ga = Ga$. Consequently, $(w, G) \sim'_C (w', G)$ and hence $((w, G), (w', G))$ is an edge of $M \odot \mathcal{P}$. Observe that the edge $((w, G), (w', G))$ is also strong, since $(w, G) \sim'_C (w', G)$ and $GC = C$.

For the second case, consider any graph H and set C as stated in Definition 4.3.1 (2). As $w \sim_A w$, clearly $w \sim_C w$ and $w \sim_{Ga} w$, for every $a \in C$. Since $\mathcal{P}$ is oblivious, we have $Pre(G) = Pre(G') = \top$, hence $(w, G), (w, G') \in W'$. It follows from the definition of C that $Ga = G'a$ for every $a \in C$ ($G \sim_a G'$ if and only if $Ga = G'a$). Thus, for every $a \in C$, $(w, G) \sim'_a (w, G')$, and consequently $(w, G) \sim'_C (w, G')$ and hence $((w, G), (w, G'))$ is an edge of $M \odot \mathcal{P}$. Moreover, we have that $((w, G), (w, G'))$ is strong because $(w, G) \sim'_C (w, G')$ and $HC = C$. □

The next lemma shows that whenever it is the case that M and $\mathcal{P}$ have strong edge sets, the product $M \odot \mathcal{P}$ has a strong edge set too. A repeated application of the lemma thus implies that if we are able to find strong edge sets of the initial epistemic model M^0 and $\mathcal{P}$, then consensus will be impossible.

Lemma 4.3.2 (Strong edge sets invariant)**.** *Consider any epistemic model $M = (W, \sim, L)$ and any oblivious pattern model $\mathcal{P} = (\mathbf{G}, Pre)$. Suppose that M and $\mathcal{P}$ are strong. Then, $M \odot \mathcal{P} = (W', \sim', L')$ is strong.*

Figure 4.1 graphically exemplifies the invariant in Lemma 4.3.2. A model M appears at the top-left of the figure, with a strong edge set having all the edges drawn with bold lines. Since the edge set is strong, for every pair of

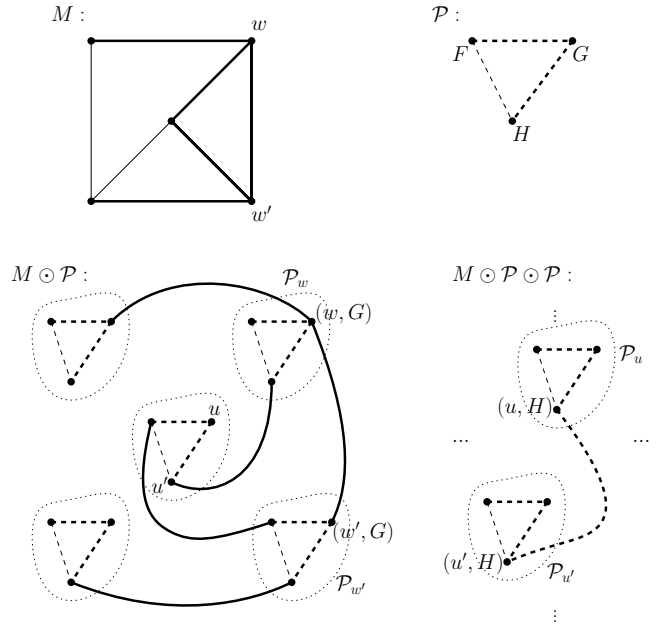


Figure 4.1: A graphic description of the invariant for strong edge sets in Lemma 4.3.2.

worlds there is a path between them that passes only through edges of the set. A communication pattern $\mathcal{P}$ appears at the top-right, with a strong edge set having all edges drawn with bold-dashed lines. In the product $M \odot \mathcal{P}$, shown below M and $\mathcal{P}$, every world w of M is replaced with a copy of $\mathcal{P}$. In the example, $\mathcal{P}_w$ denotes the copy of $\mathcal{P}$ replacing w in $M \odot \mathcal{P}$. Each bold edge is strong, and hence there exists a graph of $\mathcal{P}$ that preserves the edge in $M \odot \mathcal{P}$ (in the sense of Claim 4.3.1 (1)). In the figure, for the edge (w, w') of M, such a graph of $\mathcal{P}$ is G. Hence $M \odot \mathcal{P}$ has the edge $((w, G), (w', G))$. That edge connects the copies $\mathcal{P}_w$ and $\mathcal{P}_{w'}$. The same is true for every bold edge in Fig. 4.1. Hence we conclude that if $\mathcal{P}$ is connected and each bold edge is strong, $M \odot \mathcal{P}$ is connected. Let E' denote the edge set with bold and bold-dashed edges of $M \odot \mathcal{P}$. We thus have that $M \odot \mathcal{P}$ is E'-connected. To prove the impossibility of consensus, we would like to repeatedly apply Lemma 4.3.2. If we want to argue that $M \odot \mathcal{P} \odot \mathcal{P}$ is connected using the same idea, we will need the edge set E' of $M \odot \mathcal{P}$ to be strong. Making sure that E' is strong is the purpose of assuming that $\mathcal{P}$ has a strong set. In the

figure, the edge (G, H) of $\mathcal{P}$ is strong, then (u, u') of $M \odot \mathcal{P}$ will be strong (by Claim 4.3.1 (2)), and hence the copies $\mathcal{P}_u$ and $\mathcal{P}_{u'}$ will be connected in $M \odot \mathcal{P} \odot \mathcal{P}$ (partially depicted at the bottom).

Before delving into the proof of Lemma 4.3.2, we discuss two examples showing that, in some cases, the existence of strong edge sets of epistemic models and communication patterns are necessary for preserving connectivity after an arbitrary number of applications of the restricted modal product. In both examples the set of agents is $A = \{a, b, c\}$.

For the first example, consider a connected epistemic model M with only two worlds, w and w', such that $w \sim_a w'$ and $w \sim_b w'$, but $w \nsim_c w'$. Also consider an oblivious communication pattern $\mathcal{P}$ with a single graph U, that is, the universal relation, i.e., the in-neighbourhood in U of each agent is A. Clearly $\mathcal{P}$ is also connected. We observe that M is *not* strong (with respect to $\mathcal{P}$). If that were the case, then its only edge, (w, w') would be strong, which implies that for some non-empty subset $C \subseteq \{a, b\}$, $UC = C$. However, note that for every non-empty $C \subseteq \{a, b\}$, $UC = A \neq C$. Thus, (w, w') is not strong. Finally, it is easy to check that $M \odot \mathcal{P}$ is disconnected.

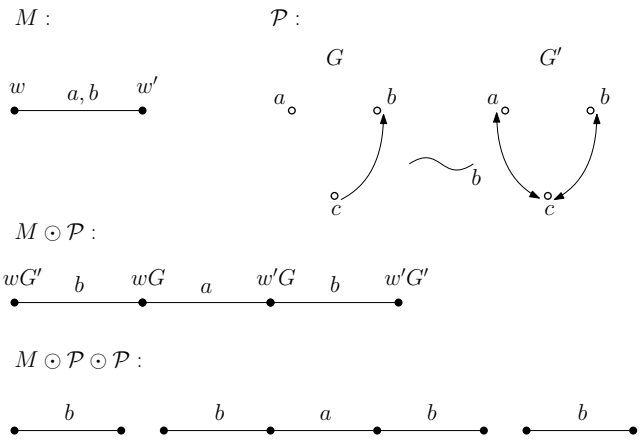


Figure 4.2: An example of M and $\mathcal{P}$ such that (1) both are connected, (2) M is strong, (3) $\mathcal{P}$ is not strong and (4) $M \odot \mathcal{P} \odot \mathcal{P}$ is disconnected.

Figure 4.2 depicts the second example. The connected epistemic model M has two worlds w and w' with distinct valuations (not shown in the picture). The oblivious communication pattern $\mathcal{P}$ consists of two graphs, G and G' with $\top$ precondition. Note that $\mathcal{P}$ is connected as well because $Gb = G'b =$

$\{c\}$. We can verify that the edge (w, w') is strong (with respect to $\mathcal{P}$), since the graph G and the set $C = \{a\}$ satisfy the requirements in Def. 4.3.1 and hence M is strong. We now show that $\mathcal{P}$ is not strong. For $\mathcal{P}$ to be strong, its edge (G, G') should be strong. If the edge is strong, then there will be a graph $H \in \{G, G'\}$ and a non-empty set $C \subseteq \{b\}$ such that $HC = C$. It must be the case that $C = \{b\}$, but $Gb = G'b = \{b, c\}$ so $\mathcal{P}$ is not strong. Finally, as Figure 4.2 shows, the epistemic model obtained after two applications of the restricted modal product is not connected. For clarity, a world (w, G) of $M \odot \mathcal{P}$ is denoted wG, and the world identifiers of $M \odot \mathcal{P} \odot \mathcal{P}$ are not depicted.

Proof of Lemma 4.3.2. Let E_M an $E_{\mathcal{P}}$ be strong edge sets of M and $\mathcal{P}$, respectively, whose existence is guaranteed by assumption. For any pair of worlds $w, w' \in W$, fix a path $P_{w,w'} = (w_1, w_2, \ldots, w_m)$ between w and w' (i.e., $w = w_1$ and $w' = w_m$) such that each $(w_k, w_{k+1}) \in E_M$ (i.e., the edge is strong); the existence of $P_{w,w'}$ directly follows from the definition of E_M. Similarly, for every pair of graphs $G, G' \in \mathbf{G}$, fix a path $P_{G,G'} = (G = G_1, G_2, \ldots, G_{m'} = G')$ between G and G' such that each $(G_k, G_{k+1}) \in E_{\mathcal{P}}$ (i.e., the edge is strong); the existence of $P_{G,G'}$ follows from the definition of $E_{\mathcal{P}}$. Below we use all these paths $P_{w,w'}$ and $P_{G,G'}$ to define a strong edge set E' of $M \odot \mathcal{P}$.

For each (w_k, w_{k+1}) of $P_{w,w'}$, $((w_k, G_k), (w_{k+1}, G_k))$ is a strong edge of $M \odot \mathcal{P}$ by Claim 4.3.1 (1) for some graph $G_k \in \mathbf{G}$. Let E' contain the strong edges $((w_k, G_k), (w_{k+1}, G_k))$.

For each (G_k, G_{k+1}) of $P_{G,G'}$ and for any world $w \in W$, $((w, G_k), (w, G_{k+1}))$ is a strong edge of $M \odot \mathcal{P}$ by Claim 4.3.1 (2). Observe that $((w, G_1), (w, G_2), \ldots, (w, G_{m'}))$ is a path in $M \odot \mathcal{P}$ between (w, G) and (w, G'). Let $P_{w,G,G'}$ denote that path. Let E' contain the strong edges $((w, G_k), (w, G_{k+1}))$.

To conclude the proof of the lemma, we show that E' is a strong edge set. By construction, all edges of E' are strong. Thus it remains to argue that, for every pair of worlds $(w, G), (w', G') \in W'$ there is a path between (w, G) and (w', G') that passes only through edges of E'. Consider the path $P_{w,w'}$ defined above. For all $1 \leq k \leq m - 1$, we have $((w_k, G_k), (w_{k+1}, G_k)) \in E'$, for some $G_k \in \mathbf{G}$. Observe that it can be the case that $G \neq G_1$, $G' \neq G_{m-1}$ or $G_k \neq G_{k+1}$, for some $1 \leq k \leq m - 2$. Hence $((w_1, G_1), (w_2, G_1), (w_3, G_2), \ldots, (w_{m-1}, G_{m-1}), (w_m, G_{m-1}))$ might not be a path between (w, G) and (w', G'). To overcome the issue, we use the paths $P_{w,G,G'}$ defined above: the paths P_{w,G,G_1}, $P_{w',G_{m-1},G'}$

and $P_{w_k,G_k,G_{k+1}}$, for each $1 \leq k \leq m-2$ together with the edges above give the desired path between (w, G) and (w', G'):

$$P_{(w=w_1),G,G_1} \cdot ((w_1, G_1), (w_2, G_1)) \cdot P_{w_2,G_1,G_2}$$
$$\cdot((w_2, G_2), (w_3, G_2)), \ldots, ((w_{m-1} \cdot G_{m-1}), (w_m, G_{m-1}))$$
$$\cdot P_{(w_m=w'),G_{m-1},G'}.$$

This concludes the proof. □

We can now show that strong edge sets preclude solving consensus:

Corollary 4.3.2.1 (Strong edge sets and impossibility of consensus)**.** *Consider any oblivious adversary Adv and let $\mathcal{P}$ be the oblivious communication pattern obtained from Adv. If the initial epistemic model M^0 and $\mathcal{P}$ are strong and $|\mathcal{I}| > 1$, then consensus is impossible against the adversary Adv.*

Proof. By induction on $r \geq 0$, we show that M^r has a strong edge set, which implies it is connected. The base case, $r = 0$, holds by assumption. For the inductive step, assume that M^r has a strong edge set. By assumption, $\mathcal{P}$ also has a strong edge set. Lemma 4.3.2 directly implies that $M^{r+1} = M^r \odot \mathcal{P}$ has a strong edge set. Thus, consensus is impossible against *Adv*, by Lemma 4.2.3. □

4.4 Applying the Condition

We now use the previous corollary to argue that consensus is impossible against three oblivious adversaries: iterated snapshot [1], iterated immediate snapshot [11], and an adversary that models the test-and-set primitive in multicore architectures (see for example [28]). The impossibility results are well known but here we derive them using our machinery. The three oblivious adversaries were originally defined for shared-memory distributed systems. Here we describe them as oblivious adversaries.

Definition 4.4.1.

1. *The* Test-And-Set *adversary, denoted TAS, contains every communication graph G satisfying that there is $a \in A$ such that $Ga = \{a\}$, and for any $b \in A$ different from a, $Gb = A$.*

2. *The* Iterated Immediate Snapshot *adversary, denoted IIS, contains every communication graph $G \in IS$ such that if $b \in Ga$, then $Gb \subseteq Ga$, for every $a, b \in A$.*

3. *The* Iterated Snapshot *adversary, denoted IS, contains every communication graph G satisfying that $Ga \subseteq Gb$ or $Gb \subseteq Ga$, for every $a, b \in A$.*

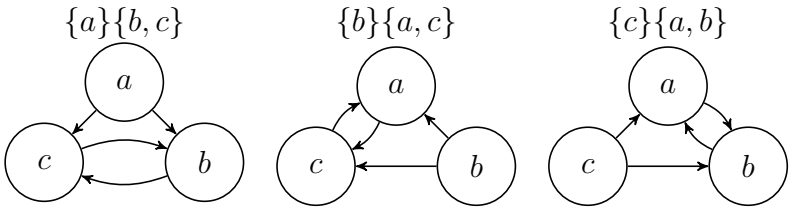


Figure 4.3: Communication graphs for three-agent TAS.

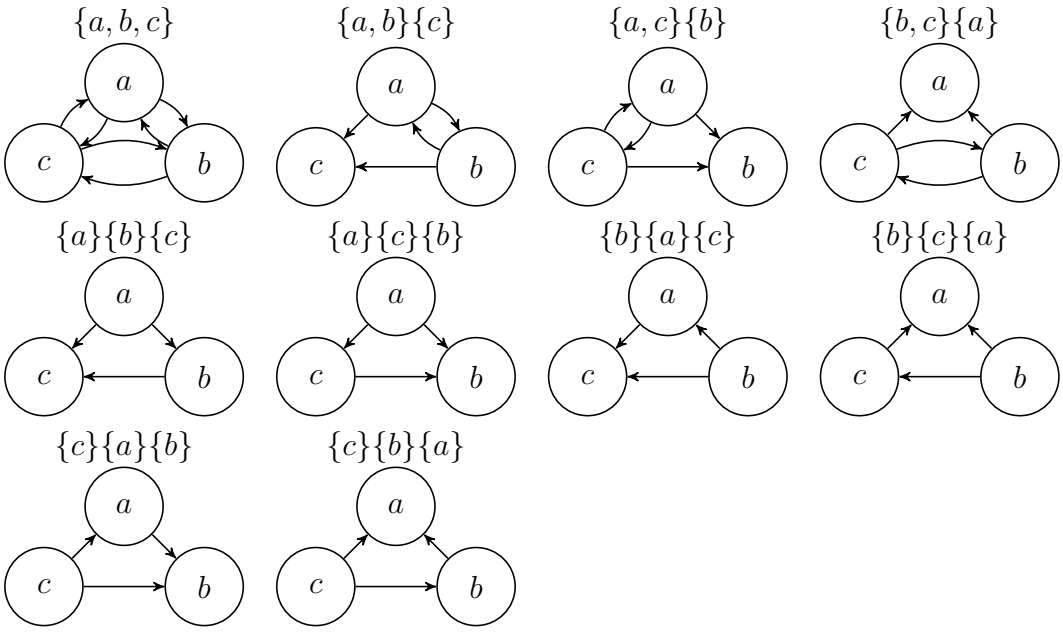


Figure 4.4: Communication graphs for three-agent IIS except for the graphs in Fig. 4.3.

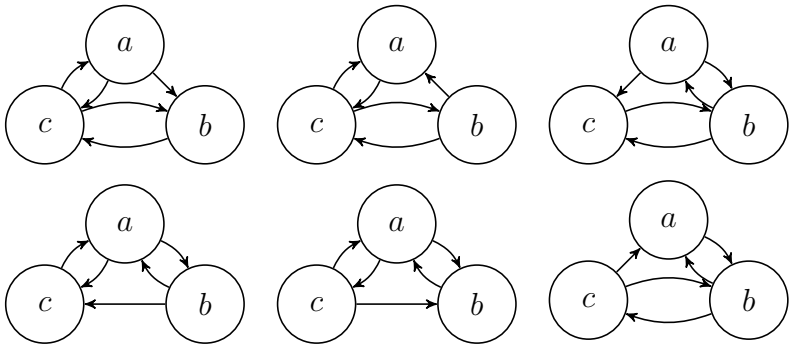


Figure 4.5: Communication graphs for three-agent IS except the graphs in Fig. 4.3 and Fig. 4.4.

It is not difficult to check that the adversaries are related as follows:

Claim 4.4.1. *If $n \geq 3$, $TAS \subset IIS \subset IS$.*

For instance, the graphs for three-agent TAS are shown in Fig. 4.3. The graphs for three-agent IIS are the graphs in Fig. 4.3 and Fig. 4.4. Finally, the graphs for three-agent IS are the graphs in Fig. 4.3, Fig. 4.4 and Fig. 4.5.

Before applying the condition, we show that M^0, the initial epistemic model, is connected. This result will be used later: we will argue that the set with all edges of M^0 is a strong edge set with respect to TAS, IIS and IS.

Claim 4.4.2 (Connectivity of M^0)**.** *The initial epistemic model M^0 is connected.*

Proof. By definition, the worlds of $M^0 = (W^0, \sim^0, L^0)$ are all input vectors over the input set $\mathcal{I}$. Let I and I' be any pair of distinct worlds (input vectors) in W^0. We have $I \sim_a^0 I'$ if and only if $I(a) = I'(a)$. Let a be any agent such that $I(a) \neq I'(a)$. Consider the world $I'' \in W^0$ such that $I''(a) = I'(a)$ and $I''(b) = I(b)$ for any agent $b \neq a$. Thus, $I \sim_a^0 I''$. Note that I'' and I' differ in one entry fewer than do I and I'. Repeating the argument at most n times we obtain that $I \, (\overset{\cup}{\sim}{}_a^0)^* \, I'$. Therefore, M^0 is connected. □

We now use Corollary 4.3.2.1 to show that the impossibility of consensus against *TAS*, as long as $n \geq 3$.[2]

[2]For the case $n = 2$, it is actually the case that consensus is possible against *TAS*. The adversary has only two graphs. In a consensus algorithm, the agents decide after one round of communication; the agent that receives no message from the other agent decides its input, and the agent receiving the input of the other decides that input.

Theorem 4.4.3 (Impossibility for *TAS*)**.** *If $n \geq 3$, consensus is impossible against TAS.*

Proof. Let $\mathcal{P}_{TAS}$ be the communication pattern obtained from *TAS*. It is not difficult to see that $\mathcal{P}_{TAS}$ is connected: since $n \geq 3$, for every two of its graphs, G and G', there is at least one $a \in A$ such that $Ga = G'a = A$, hence $G \sim_a G'$. We now argue that each edge of M^0 is strong and each edge of $\mathcal{P}_{TAS}$ is strong too. Thus, the edge sets with all edges of M^0 and $\mathcal{P}_{TAS}$, respectively, are strong.

Consider any edge (w, w') of M^0. Pick any $a \in A$ such that $w \sim_a w'$. Let $C = \{a\}$. Let G be the graph of $\mathcal{P}_{TAS}$ such that $Ga = \{a\}$. Definition 4.3.1 (1) is satisfied since $Ga = C$. Thus the edge is strong.

Consider any edge (G, G') of $\mathcal{P}_{TAS}$. We have two cases. If $G = G'$, then for $C = A$, $H = G$, Definition 4.3.1 (2) is clearly satisfied. In the second case, $G \neq G'$, the definition of *TAS* and the fact that $n \geq 3$, by assumption, guarantee that there are distinct $a, b \in A$ such that $Ga = \{a\}$, $G'b = \{b\}$ and $G'a = Gb = A$, and for every $c \in A \setminus \{a, b\}$, $Gc = G'c = A$. Pick any $c \in A \setminus \{a, b\}$. Let $C = \{c\}$ and H be the graph of $\mathcal{P}_{TAS}$ such that $Hc = \{c\}$. We have $G \sim_c G'$ and $Hc = C$, which clearly satisfies Definition 4.3.1 (2). Thus, (G, G') is also strong in this case.

The conditions of Corollary 4.3.2.1 hold, and hence the theorem follows. □

The previous theorem and the containments in Claim 4.4.1 directly give the consensus impossibility for the other two adversaries, *IS* and *IIS*. The reason is that if there is a consensus algorithm against $Adv \supset TAS$, that algorithm solves consensus against *TAS*, as all executions of *TAS* are executions of *Adv*. But we already know that consensus is impossible against *TAS*, by the previous theorem, therefore consensus is impossible against *Adv*. However, for completeness, we present impossibility proofs for *IS* and *IIS* based on strong edge sets.

Theorem 4.4.4 (Impossibility for *IS*)**.** *Consensus is impossible against IS.*

Proof. Let $\mathcal{P}_{IS}$ be the communication pattern obtained from *IS*. We first check that $\mathcal{P}_{IS}$ is connected. Consider any two graphs G, G' of $\mathcal{P}_{IS}$. By definition of *IS*, $Ga \subseteq Gb$ or $Gb \subseteq Ga$, for every $a, b \in A$. As for every $a \in A$, $a \in Ga$, there must exist a $c \in A$ such that $Gc = A$. Let c be any such agent. It similarly happens to G'. Pick any $d \in A$ with $G'd = A$.

Let H be the graph of $\mathcal{P}_{IS}$ such that $Ga = A$, for every agent $a \in A$. Hence we have $G \sim_c H \sim_d G'$. Therefore we conclude that $\mathcal{P}_{IS}$ is connected.

We now argue that each edge of both M^0 and $\mathcal{P}_{IS}$ is strong. Consider any edge (w, w') of M^0. Pick any $a \in A$ such that $w \sim_a w'$. Let $C = \{a\}$ and G be any graph of $\mathcal{P}_{IS}$ such that $Ga = \{a\}$ (an example of such a graph is the one in which $Gb = A$, for every agent $b \neq a$). Clearly $Ga = C$, which satisfies Definition 4.3.1 (1). Thus the edge is strong.

Consider any edge (G, G') of $\mathcal{P}_{IS}$. If $G = G'$, then we set $C = A$ and $H = G$, which clearly satisfy Definition 4.3.1 (2). If $G \neq G'$, consider any agent $a \in A$ such that $G \sim_a G'$. The definition of *IS* implies that there is a graph H of $\mathcal{P}_{IS}$ such that $Ha = \{a\}$. Let $C = \{a\}$. Clearly, $G \sim_a G'$ and $Ha = C$, which satisfies Definition 4.3.1 (2). Thus, (G, G') is strong.

The theorem directly follows from Corollary 4.3.2.1. □

Theorem 4.4.5 (Impossibility for *IIS*)**.** *Consensus is impossible against IIS.*

Proof. Let $\mathcal{P}_{IIS}$ be the pattern model built from IIS. We will identify the graph that corresponds to the sequence of concurrency classes $[C_1, C_2, \ldots, C_k]$ as $G_{[C_1,C_2,\ldots,C_k]}$.

First, M^0 is connected from Thm. 4.4.2. We will prove that the edges of M^0 are strong. Let (w, w') be an edge of M^0 and $B = \{a \in A \mid w \sim_a w'\}$. If $B = A$, then (w, w') is strong with $C = A$ and the graph $G_{[A]}$ Otherwise, $B \subset A$ and (w, w') is strong with $C = B$ and the communication graph $G_{[B,A-B]}$.

Second, consider the set of edges $E_{\mathcal{P}_{IIS}} = \{(G_{[A]}, G_{[C_1,C_2,\ldots,C_k]})$ of $\mathcal{P}_{IIS}$. $E_{\mathcal{P}_{IIS}}$ clearly connects $\mathcal{P}_{IIS}$. The edge $(G_{[A]}, G_{[C_1,C_2,\ldots,C_k]})$ is strong with $C = C_1$, and the communication graph $G_{[C_1,A-C_1]}$. Thus, $E_{\mathcal{P}_{IIS}}$ is strong.

The theorem directly follows from Corollary 4.3.2.1.

□

After presenting our sufficient condition for preserving connectivity in this chapter and showing some examples on how to test the condition, we present, in Ch. 5, a parametrization of pattern models with an arbitrary protocol. This parametrization is needed for protocol verification. Additionally, the structures are modified to simplify the formalism.

Chapter 5

Parametrized pattern models

We can arguably say that the pattern model formalism, defined in Ch. 3 and used in Ch. 4, has two inconveniences: (1) the the size of the preconditions of the communication graphs with respect to the number of rounds may grow exponentially, and (2) the fact that only the full-information protocol is describable using the formalism. (1) is expected, as the number of worlds grows exponentially in r and the precondition of a communication graph selects the worlds where the graph may describe a possible next round of communication between the agents. (2) is an inconvenience because formal verification may require to analyze communication running an arbitrary protocol. In this chapter, we present a modification of pattern models that overcomes both inconveniences.

Now, we modify the formalism for dealing with arbitrary protocols avoiding the computation of graph preconditions.

First, we will provide a way to define a protocol. Second, we will redefine a pattern model deleting the preconditions of the graphs. Third, we will define *epistemic models for distributed systems* by replacing $\sim$, in the epistemic models, with a function, $\mathcal{S}$, that we will use for tracking local states of agents and adding other function for tracking the sequence of communication graphs. Also, since we do not have graph preconditions anymore, we will aggregate other function S for tracking the sequence of communication graphs occurred in a given r-execution. Note that $\mathcal{S}$ and S have different typography. Finally, we will redefine the semantics for interpreting a formula in $\mathcal{L}_{DC}$ on an epistemic model for distributed systems, and the restricted modal product taking into account a given protocol. In this case, the semantics will be defined for the logic $\mathcal{L}_{DC}$, i.e., without an update modality. We

will treat an update as an external procedure because we have not found an axiomatization for a logic with update an modality yet.

5.1 Protocol definition

Let us define an arbitrary deterministic protocol. Let *Loc* be a set of local states of the agents that run a protocol. Let *Msg* be a set of possible message contents (i.e., the information that an agent can send to the other agents). A protocol $\mathbf{P}$ is a pair of functions (μ, λ), where:

- $\mu : Loc \to (Msg \cup \{\bot\})^n$ is a function that specifies the messages that an agent may send to the other agents
- $\lambda : Loc \times (Msg \cup \{\bot\})^n \to Loc$ is a function that specifies the next local state of an agent given the messages that it receives from the other agents

The i-th element in the tuples in the image of μ is the message to be sent to agent a_i. The domain of λ are pairs. The second element of each pair is an n-tuple. The i-th element in such an n-tuple is the message from a_i or $\bot$ if no message arrived from a_i. The $\bot$ symbol represents that a message was not sent or received and it is assumed that $\bot \notin Msg \cup \mathcal{I}$.

5.2 Simplified pattern models

Definition 5.2.1 (Simplified pattern model)**.** *A* simplified pattern model*,* $\mathbf{G}$*, is a set of communication graphs.*

Now, a pattern model only consists of a set of communication graphs.

5.3 Epistemic models for distributed systems

Definition 5.3.1. *An* epistemic model for distributed system D *is a four-tuple* $(W, \mathcal{S}, L, S)$ *where* W *and* L *are defined as in Sect. 2.4, and*

- $\mathcal{S} : W \times A \to Loc$ *is a function that associates a (world, agent)-pair with a local state*

- $S : W \to \mathbf{G}_A^*$ *is a function that associates a world with a sequence of communication graphs where* $\mathbf{G}_A^*$ *is the set of finite sequences of communication graphs of* $\mathbf{G}_A$.

Note that, in some sense, $\mathcal{S}$ associates a world w with the corresponding configuration of the system: $(\mathcal{S}(w, a_1), \mathcal{S}(w, a_2), \ldots, \mathcal{S}(w, a_n))$.

5.3.1 The initial epistemic model for distributed systems.

Since the protocols are now arbitrary, we need a way to state the initial local state of an agent depending on a protocol $\mathbf{P}$. We use a function $s_{\mathbf{P}} : A \times I \to Loc$, which depends on the protocol, that associates an (agent, input)-pair, (a_i, v), with the initial local state of a_i on protocol $\mathbf{P}$ starting with input v.

We build $D^0_{\mathbf{P},Adv} = (W^0, \mathcal{S}^0, L^0, S^0)$, the initial epistemic model for distributed systems for $\mathbf{P}$, any adversary Adv and $\mathcal{I}$ with $P = \{v_{a_i} \mid a_i \in A \text{ and } v \in \mathcal{I}\}$ as follows. W^0, $\mathcal{S}^0$ and L^0 are defined as in Subsect. 2.4.1, and

- $\mathcal{S}^0(I, a_i) = s_{\mathbf{P}}(a_i, I(i))$
- $S(w) = \epsilon \; \forall w \in W^0$

An example is shown in Fig. 5.1. In such a model, $s_{\mathbf{P}}(a_i, I(i)) = I(i)$. W is depicted as nodes. L is depicted in the top of the nodes. S is added in the middle of the nodes. And, $\mathcal{S}$ is depicted in the bottom of the nodes as an array where its i-th position is $\mathcal{S}(w, a_i)$. An edge (w, w') labelled with agent a is shown if $\mathcal{S}(w, a_i) = \mathcal{S}(w', a_i)$.

5.4 Parametrized pattern models logic

From now on, consider an arbitrary protocol $\mathbf{P}$ and an arbitrary input set $\mathcal{I}$, the pattern models $\mathbf{G}^1, \mathbf{G}^2, \ldots$ for Adv as defined above, and $D^0_{\mathbf{P}}$, the initial epistemic model for distributed systems for A, $\mathcal{I}$, and $\mathbf{P}$.

Let us define $m^w_{a_i} \mid_G = (m_1, m_2, \ldots, m_n) \in (Msg \cup \{\perp\})^n$ where

$$m_j = \begin{cases} \mu(\mathcal{S}(w, a_j))(i) & \text{if } a_j G a_i \\ \perp & \text{otherwise.} \end{cases}$$

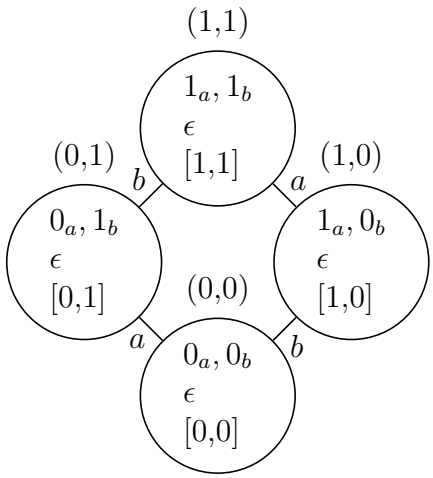


Figure 5.1: Epistemic model for distributed systems.

The j-th element, m_j, of $m^w_{a_i} \mid_G$ is the message that agent a_i receives from agent a_j in world w when G occurs or $\perp$ if the message does not arrive.

Since the syntax of $\mathcal{L}_{DC}$, as well as the pattern model structure, are already defined, we only define here the epistemic models for distributed systems, the parametrized restricted modal product given a protocol $\mathbf{P}$, and the semantics on epistemic models for distributed systems.

Definition 5.4.1 (Epistemic models for distributed systems and parametrized restricted modal product)**.** *Let $\mathbf{G}^1, \mathbf{G}^2, \ldots$ be the pattern models describing Adv. We define the epistemic models for distributed systems for $\mathbf{P}$ and Adv and the parametrized restricted modal product simultaneously as follows.*

The initial epistemic model for distributed systems, $D^0_{\mathbf{P},Adv}$, is an epistemic model for distributed systems for $\mathbf{P}$ and Adv.

Given $D^i_{\mathbf{P},Adv} = (W^i, \mathcal{S}^i, L^i, S^i)$, and $\mathbf{G}^{i+1}$, $D^{i+1}_{\mathbf{P},Adv} = D^i_{\mathbf{P},Adv} \odot_{\mathbf{P}} \mathbf{G}^{i+1} = (W^{i+1}, \mathcal{S}^{i+1}, L^{i+1}, S^{i+1})$ is an epistemic model for distributed systems for $\mathbf{P}$ and Adv, where $\odot_{\mathbf{P}}$ is defined so that:

- $W^{i+1} = \{(w, G) \in W^i \times \mathbf{G}^{i+1} \mid S^i(w) \cdot G$ *is a prefix of Adv*$\}$
- $\mathcal{S}^{i+1}((w, G), a) = \lambda(\mathcal{S}^i(w, a), m^w_a \mid_G)$
- $L^{i+1}((w, G)) = L^i(w)$
- $S^{i+1}((w, G)) = S^i(w) \cdot G$

Definition 5.4.2 (Semantics)**.** *Let $D = (W, \mathcal{S}, L)$ be an epistemic model for distributed systems. Let $p \in P$, $w, w' \in W$, $a \in A$, and $\varphi, \psi \in \mathcal{L}_{CD}$ be given. Let us define $\sim_a = \{(w, w') \in W \times W \mid \mathcal{S}(w, a) = \mathcal{S}(w', a)\}$.*

$D, w \models p$	*iff* $p \in L(w)$
$D, w \models \neg\varphi$	*iff* $D, w \not\models \varphi$
$D, w \models \varphi \wedge \psi$	*iff* $D, w \models \varphi$ *and* $D, w \models \psi$
$D, w \models D_B\varphi$	*iff for all* w' *such that* $w \sim_B w'$ $D, w' \models \varphi$
$D, w \models C_B\varphi$	*iff for all* w' *such that* $w \overset{\cup}{\sim}{}^*_B w' D, w' \models \varphi$

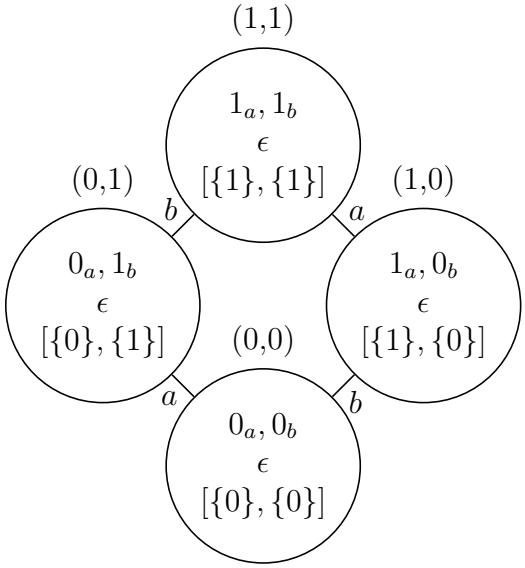


Figure 5.2: Epistemic model for distributed computing $D^0_{\mathbf{P}_\cup}$.

Example 5.4.1. *As an example, we demonstrate a protocol for two processes, $\mathbf{P}_\cup = (\mu_\cup, \lambda_\cup)$, that changes the structure after one update on IIS. In this protocol, an agent sends its local state to the other process and updates its local state with the set of inputs that it receives at the end of the round. Thus, we will have three local states: $\{0\}$, $\{1\}$, and $\{0, 1\}$. In this case, we can define $Loc_\cup = Msg_\cup = \{\{0\}, \{1\}, \{0, 1\}\}$. Let $s, s_1, s_2 \in Loc_\cup = Msg_\cup$.*

The message function, $\mu_\cup$, is defined as follows:

$$\mu_\cup(s) = (s, s).$$

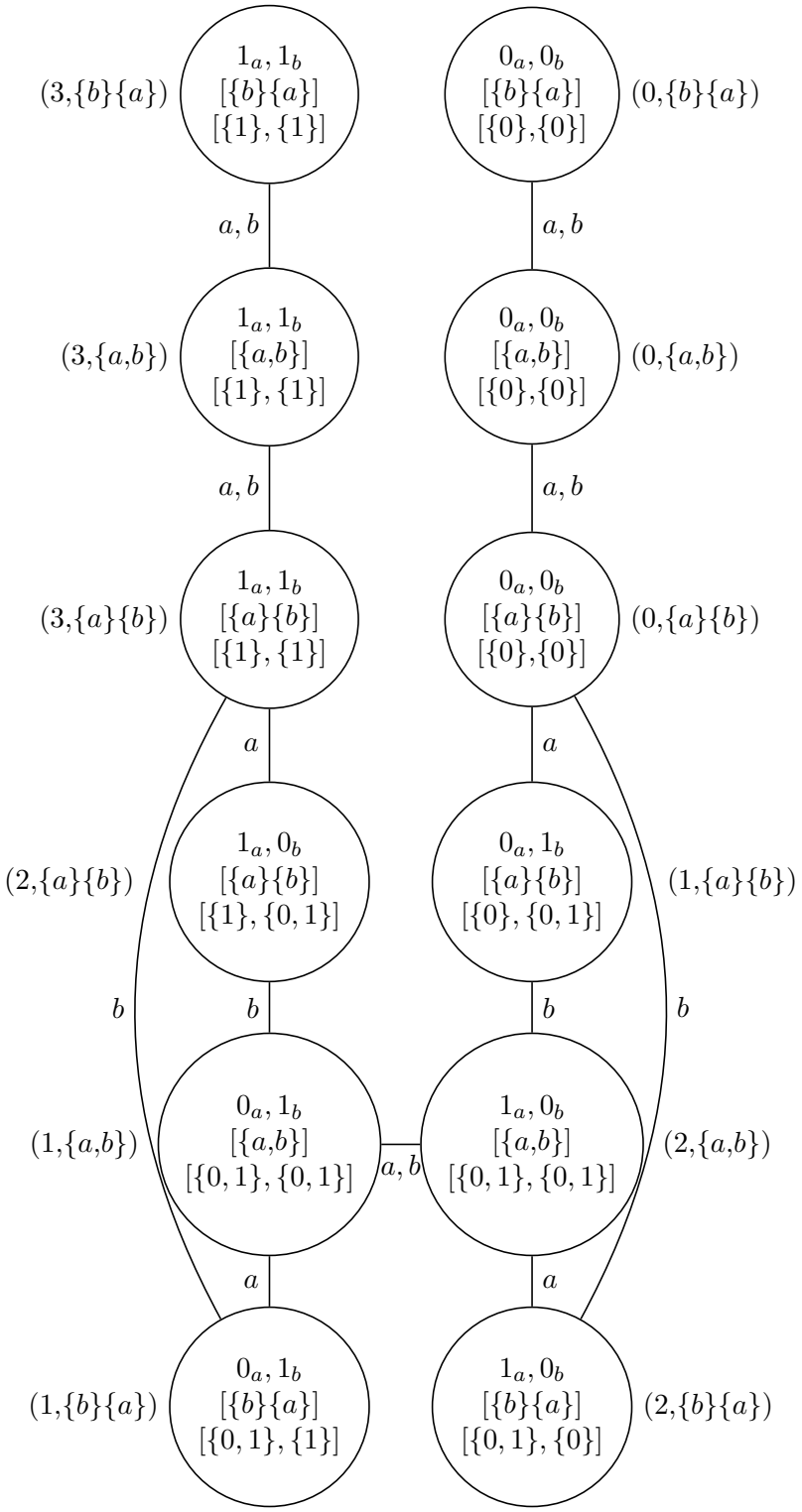


Figure 5.3: $D^0_{\mathbf{P}_\cup} \odot_{\mathbf{P}_\cup} \mathcal{P}_{\textit{two-IIS}}$.

The local state transition function, $\lambda_\cup$, is defined as follows:

$$\lambda_\cup(s,(s_1,s_2)) = s_1 \cup s_2.$$

Note that for any value ($\{0\}$, $\{1\}$, or $\{0,1\}$) of s_1 and s_2, $s = s_1$ or $s = s_2$.

The initial epistemic model for distributed systems for this protocol, $D^0_{\mathbf{P}_\cup}$, is shown in Fig. 5.2. In such a model, $s_{\mathbf{P}_\cup}(a_i, I(i)) = \{I(i)\}$. After one round of communication in IIS executing $\mathbf{P}_\cup$, the updated epistemic model for distributed systems is $D^0_{\mathbf{P}_\cup} \odot_{\mathbf{P}_\cup} \mathcal{P}_{two\text{-}IIS}$, shown in Fig. 5.3.

5.5 Parametrized pattern models for arbitrary adversaries

The sequence of pattern models $\mathbf{G}^1, \mathbf{G}^2, \ldots$ that describes an adversary *Adv* is defined as follows.

- $\mathbf{G}^i = \{G \in \mathbf{G}_A \mid$ there is an i-execution $(I, S \cdot G)$ of $Adv\}$.

5.5.1 The product $\odot_{\mathbf{P}}$ reflects the local state change through rounds

In Subsect. 3.4.1 we defined the conditions that a sequence of pattern models must satisfy to reflect an adversary. In such a definition, the full-information protocol was assumed. Now, we slightly modify the definition of reflect, for arbitrary protocols.

Let *Adv* be an adversary. For every $i \geq 0$, we define the set $\mathcal{C}^i_{Adv,\mathbf{P}} = \{C \mid$ there is an i-execution (I, S) of *Adv* that ends in the configuration C executing the protocol $\mathbf{P}\}$, where the configurations are elements in Loc^n.

Let $\mathbf{G}^1, \mathbf{G}^2, \ldots$ be an infinite sequence of pattern models. We say that the sequence $\mathbf{G}^1, \mathbf{G}^2, \ldots$ *reflects* the adversary *Adv* on protocol $\mathbf{P}$ if for each $r \geq 1$, there is a surjection $f^r : W^r \to \mathcal{C}^r_{Adv,\mathbf{P}}$, where $D^0_{\mathbf{P},Adv}$ is the initial epistemic model for distributed systems built as described above and $D^r_{\mathbf{P}} = (W^r, \mathcal{S}^r, L^r, S^r) = D^0_{\mathbf{P}} \odot \mathbf{G}^1 \odot \mathbf{G}^2 \odot \cdots \odot \mathbf{G}^r$. If $\mathbf{G}^1 = \mathbf{G}^2 = \ldots$, we simply say that $\mathbf{G}^1$ reflects *Adv* on protocol $\mathbf{P}$.

The "reflect" definition for arbitrary protocols takes into account the protocol in the definition of the $\mathcal{C}^i_{Adv,\mathbf{P}}$ sets, and changes the requirement of f^r to be a surjection instead of a bijection. The property of f^r that

ensures $w \sim_a w'$ (in an epistemic model) if and only if $f^r(w)$ and $f^r(w')$ are indistinguishable for a is actually still there, but it relies on the new semantics defined because knowledge is defined in terms of $\mathcal{S}$.

Notice that in this case, depending on the protocol, there may be no bijection between the worlds of $D^0_{\mathbf{P},Adv} \odot_{\mathbf{P}} \mathbf{G}^1 \odot_{\mathbf{P}} \mathbf{G}^2 \odot_{\mathbf{P}} \cdots \odot_{\mathbf{P}} \mathbf{G}^i$ and the configurations because *two different i-executions may end in the same configuration*. To see this, consider the following scenario:

For simplicity, we will think of only one agent that has a binary input, and that the agent is able to distinguish between both values. The protocol $\mathbf{P}_f = (\mu_f, \lambda_f)$ is described as follows: agent a sends its own local state to itself and changes its local state to the local state $-$. Formally, $Loc_{\mathbf{P}_f} = Msg_{\mathbf{P}_f} = \{0, 1, -\}$, $\mu_f(v) = (v)$, and $\lambda_f(v, (v)) = -$, with $v \in Loc_{\mathbf{P}_f} = Msg_{\mathbf{P}_f}$. The initial epistemic model for distributed systems $D^0_{\mathbf{P}_f}$ modeling such a situation is shown in Fig. 5.4. Here, $s_{\mathbf{P}_f}(a_i, I(i)) = I(i)$. The pattern model $\mathbf{G}_f$, shown in Fig. 5.5, consists of the unique reflexive communication graph on the set $\{a\}$. The epistemic model for distributed systems $D^0_{\mathbf{P}_f} \odot_{\mathbf{P}} \mathcal{P}$ is shown in Fig. 5.6.

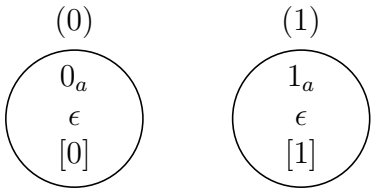


Figure 5.4: Epistemic model for distributed systems $D^0_{\mathbf{P}_f}$.

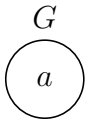


Figure 5.5: Pattern model $\mathbf{G}_f$.

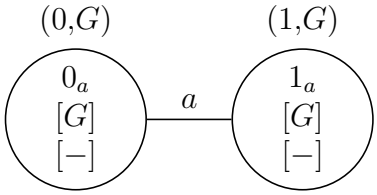


Figure 5.6: Epistemic model for distributed systems $D \odot_{\mathbf{P}} \mathbf{G}$.

Theorem 5.5.1 (Correspondence between worlds and configurations for parametrized pattern models)**.** *Let Adv be an adversary,* $\mathbf{P}$ *be an arbitrary protocol, and* $\mathbf{G}^1, \mathbf{G}^2, \ldots$ *be the pattern models built from Adv, as described above. Then,* $\mathbf{G}^1, \mathbf{G}^2, \ldots$ *reflects Adv on protocol* $\mathbf{P}$*.*

To prove Theorem 5.5.1, we will define f^r for all $r \in \mathbb{N}$ and prove it is a surjective function.

Proof. For each $r \in \mathbb{N}$ we define the function

$$f^r(w) = (\mathcal{S}^r(w, a_1), \mathcal{S}^r(w, a_2), \ldots, \mathcal{S}^r(w, a_n))$$

We will prove that f^r is surjective by induction on r.

Base case. Let us observe two facts. First, $D^0_{\mathbf{P}}$ is built so that we have a world I for each input vector I. Second, the 0-execution $(I, [\,])$ ends in configuration $C = (s_{\mathbf{P}}(a_1, I(1)), s_{\mathbf{P}}(a_2, I(2)), \ldots, s_{\mathbf{P}}(a_n, I(n)))$.

Let $C \in \mathcal{C}^0_{Adv,\mathbf{P}}$. By definition of $\mathcal{C}^0_{Adv,\mathbf{P}}$, there is a 0-execution, $(I, [\,])$, that ends in C. Moreover, $C = (s_{\mathbf{P}}(a_1, I(1)), s_{\mathbf{P}}(a_2, I(2)), \ldots, s_{\mathbf{P}}(a_n, I(n)))$ because $s_{\mathbf{P}}$ computes the initial local state of a process given its input. In particular, $f^0(I) = (s_{\mathbf{P}}(a_1, I(1)), s_{\mathbf{P}}(a_2, I(2)), \ldots, s_{\mathbf{P}}(a_n, I(n)))$, and $I \in W^0$ by construction. Then, $f^0(I) = C$. Thus, f^0 is surjective.

Inductive hypothesis. Let $D^k_{\mathbf{P},Adv} = (W^k, \mathcal{S}^k, L^k, S^k) = D^0_{\mathbf{P},Adv} \odot_{\mathbf{P}} \mathbf{G}^1 \odot_{\mathbf{P}} \mathbf{G}^2 \odot_{\mathbf{P}} \cdots \odot_{\mathbf{P}} \mathbf{G}^k$, and $w, w' \in W^k$. We assume that f^k is surjective.

Inductive step. Let $D^{k+1}_{\mathbf{P}} = (W^{k+1}, \mathcal{S}^{k+1}, L^{k+1}, S^{k+1}) = D^k_{\mathbf{P}} \odot_{\mathbf{P}} \mathcal{P}^{k+1}$. We need to show that f^{k+1} is surjective.

Let $C^{k+1} = (l^{k+1}_{a_1}, l^{k+1}_{a_2}, \ldots, l^{k+1}_{a_n}) \in \mathcal{C}^{k+1}_{Adv,\mathbf{P}}$. By definition of $\mathcal{C}^{k+1}_{Adv,\mathbf{P}}$, there is a $k+1$-execution, $(I, [G_1, G_2, \ldots, G_k, G_{k+1}])$, that ends in C^{k+1}. Since

$C^{k+1} \in \mathcal{C}^{k+1}_{Adv,\mathbf{P}}$, there is a configuration $C^k \in \mathcal{C}^k_{Adv,\mathbf{P}}$ such that the k-execution $(I, [G_1, G_2, \ldots, G_k)$ ends in C^k. Since f^k is surjective, and $C^k \in \mathcal{C}^k_{Adv,\mathbf{P}}$, there is $w \in W^k$ such that (1) $f^k(w) = C^k$, because f^k is surjective, (2) for all $1 \leq i \leq n$ $\lambda(S^k(w, a_i), m^w_{a_i} \mid G^{k+1}) = l^{k+1}_{a_i}$, otherwise C^{k+1} would not be an element of $\mathcal{C}^{k+1}_{Adv,\mathbf{P}}$ because $\mathbf{P}$ is defined by μ and λ, and (3) $S^k(w) \cdot G^{k+1}$ is a prefix of Adv. $(w, G) \in W^{k+1}$ because $(I, [G_1, G_2, \ldots, G_k, G_{k+1}])$ is a $k+1$-execution. Since $f^{k+1}(w, G) = C^{k+1}$, f^{k+1} is surjective.

□

After presenting the parametrized pattern models in this chapter, we conclude this thesis, in Ch. 6, with a final discussion.

Chapter 6

Final discussion

After presenting our results in Ch. 3, Ch. 4 and Ch. 5, we close this thesis by mentioning related work in Sect. 6.1, concluding in Sect. 6.2, and mentioning some avenues for future research in Sect. 6.3.

6.1 Related work

The formal treatment of knowledge in distributed computing was pioneered by Halpern and Moses in [24]. Perhaps their most important result is having proved that common knowledge amounts to simultaneity. The book by Fagin, Halpern, Moses, and Vardi [20] was pivotal, as it summarized numerous results and compared different approaches to studying many aspects of knowledge in a system of agents.

Action models first appeared in [5]. Such a formalism, however, was only first considered for modeling evolution of knowledge in distributed systems in [22], by Goubault, Ledent, and Rajsbaum and in [35], by Pfleger and Schmid, as far as we know.

Closer to our work is [22], where the authors exhibit a tight connection between the topological approach [26] to distributed computing and Kripke models. A second contribution of [22] is employing the *restricted modal product* operator of action models to model knowledge change between agents after a round. A third important result is employing action models to represent "tasks". A task is the equivalent of a function in distributed computability. The task defines the possible inputs to the agents, and for each set of inputs, it specifies the set of outputs that the agents may produce. By representing

the task itself, the possibility of solving a task amounts to the existence of a certain simplicial map.

The objective of [35], which uses action models as well, is that of obtaining lower limits on the number of bits necessary for implementing a protocol that is specified with an initial epistemic model and an infinite sequence of action models that describe how the epistemic model is updated through an infinite sequence of communication rounds. Like us, [35] uses dynamic-network models. Unlike us, [35] assumes that the action models are given. As a result, [35] does not build an action model and is not concerned with the size of the action models.

The work in [8] observes drawbacks similar to the ones we found when using the action model framework in other contexts. The authors propose an extension of epistemic models adding a function and an update mechanism. Adding such a function decreases the number of events needed to represent certain problems. Our proposal, however, can be directly applied to the context of distributed systems by the communication between agents.

In [46], the pattern models are presented as an extension of action models where each event is associated with a communication graph. In this contribution, we correct the construction of the infinite sequences of pattern models given an adversary whose preconditions were wrong in [46]. Additionally, the formalism is modified so that the set of communication graphs are the set of events, and the event relations are removed. The first modification limits the number of events because the number of communication graphs depends on the number of agents. The second modification causes losing a direct transformation from action models to communication pattern models. In contrast to [46], that describes agents exclusively communicating using the full-information protocol, we propose a parametrization of the logic for analyzing arbitrary protocols. Hence, despite the consequences of the changes above, pattern models are more adequate for analyzing distributed systems.

Baltag and Smets [6] proposed a similar formalism to that in [46]. The motivation in [6] was that of analyzing epistemic superiority between groups of agents in scenarios where an agent reads the whole databases from other agents. Our motivation in [46], in contrast, was that of finding succinct epistemic change representations in distributed models in a DEL approach. The main differences between these two works are the use of preconditions in [46] to describe communication allowed for non-oblivious models and the modalities for epistemic superiority in [6]. Both approaches, however, were designed to model agents communicating all they know to the others.

In [15], the authors present a derived version of pattern models without preconditions similar to the logical semantics in [6] with an axiomatization that is a modification of the one presented in [6]. Additionally, they present a simplicial complex interpretation of the language, similar to the one used in [22, 42]. In [16], the authors prove that the pattern models of [15] and action models are incomparable in update expressivity, i.e., there are some updates that can be obtained using pattern models and cannot be obtained using action models and vice versa.

The solvability of consensus against oblivious adversaries [19] is well understood. First, Santoro and Widmayer showed that consensus is impossible if up to $n-1$ messages may be lost by the same agent in each round [37]. This result was later complemented by Schmid, Weiss and Keidar [38], who showed that consensus is possible if a quadratic number of messages is lost per round, as long as these losses do not isolate the processes.

Coulouma, Godard, and Peters [19] were the first to provide a full consensus solvability characterization, based on their *beta* equivalence relation over the communication graphs of the adversary. The characterization states that consensus is possible if and only if each equivalence class satisfies the property of being *broadcastable*. Roughly speaking, a beta equivalence class defines a connected component of the Kripke model of the full-information protocol, and broadcastability means that if only graphs in a beta class occur in an execution, there is a round such that, at the end of it, there is at least one agent that is able to communicate (i.e., broadcast) its input to all agents. Intuitively, this means that at that moment there is common knowledge on the input of such agents, in the corresponding connected component. The condition of Coulouma et al. shows that, for consensus to be solvable, it is not enough to require Kripke models to be disconnected. Our impossibility characterization only guarantees Kripke models remain connected at all times, implying that there is a single beta equivalence class, which is necessarily non-broadcastable. Arguably, our impossibility characterization is simpler to state and test, although it is not a full solvability characterization.

Recently, Winkler, Paz, Rincon Galeana, Schmid, and Schmid [49] identified a simpler characterization, based on the notion of *root components*: a root component of a communication graph is a connected component with no incoming edges from the outside of the component. They observed that the non-trivial case is when each graph G of the adversary has a single root component, which is denoted $root(G)$. Winkler et al. provided a simple and elegant sequential (i.e., centralized) algorithm that decides whether consen-

sus is solvable or not. Intuitively, their algorithm iteratively constructs the beta classes of Coulouma et al. (through the root component notion) up to a moment when it is possible to determine whether consensus is solvable or not, against the adversary the algorithm starts with. This constructive approach allowed them to provide time bounds for the solvability of consensus, in the cases where consensus is solvable. Roughly, the algorithm starts with the graph $\mathbf{G}$ of the communication pattern $\mathcal{P} = (\mathbf{G}, Pre)$, and each iteration produces a new graph $\mathbf{G}'$ with $V(\mathbf{G}') = V(\mathbf{G})$ and $E(\mathbf{G}') \subseteq E(\mathbf{G})$. An edge $(G, G') \in E(G)$ remains in $\mathbf{G}'$ if there exists a graph H in the connected component of G (and G') in $\mathbf{G}$ such that $root(H) \subseteq B = \{a \in A \mid G \sim_a G'\}$. At the end of the execution of the algorithm, consensus is solvable if and only if all connected components of the final graph $\mathbf{G}'$ are *root compatible*, which basically means that they are broadcastable. Our notion of strong edges (Definition 4.3.1) is related to the requirement $root(H) \subseteq B = \{a \in A \mid G \sim_a G'\}$. Both guarantee that the edge (G, G') somehow remains in the next round (recall Claim 4.3.1 and see [49, Claim 1]).

Being based on DEL machinery, our consensus impossibility characterization for oblivious adversaries is obtained using a different approach from those in the aforementioned papers.

The case of non-oblivious adversaries is more complicated. One of the main difficulties is that an adversary of this kind might not be limit-closed, namely models do not need to be *compact* (e.g. [7, 31, 50]). For the case $n = 2$, Fevat and Godard [21] showed a full solvability characterization for non-oblivious adversaries. Intuitively, their characterization states that particular communication graph sequences should not be in the adversary to make consensus solvable. Recently, Nowak, Schmid and Winkler [33] provided a full solvability characterization for consensus against general message adversaries for any number of processes using point-set topology techniques in the style of [3]. Roughly speaking, their characterization states that consensus is solvable if and only if a topological space obtained from the communication graph sequences of the adversary can be partitioned into at least two non-empty sets that are both closed and open.

6.2 Concluding remarks

The formalization of knowledge in the distributed-computing literature has still to have a more significant impact. The evidence is that many papers in

distributed computing refer to knowledge informally.

At the same time, in the epistemic-logic literature, the formalism of action models has emerged as an important mechanism for modeling the evolution of knowledge. Hence, the works by Goubault, Ledent, and Rajsbaum [22], establishing a connection between action models and a topological approach to distributed systems, and by Pfleger and Schmid [35], determining communication complexity lower bounds for solving distributed computing problems, are relevant.

The use of action models for describing knowledge dynamics on distributed systems has certain inconveniences, as already discussed in [46], for example. The main disadvantage is that of describing the communication between agents by the event preconditions. To overcome these disadvantages, we proposed the *pattern models* formalism. Our models are applicable to a large variety of distributed-computing models, called dynamic-network models. We define pattern models systematically for every round of execution given an adversary. In the case of oblivious models, the pattern model remains the same all through the computation so that we can describe an oblivious dynamic-network model in constant space.

A simple modification in the definition of the restricted modal product $\odot$, parametrizing such a product with a given protocol $\mathbf{P}$, generates a new dynamic epistemic logic that works specifically for such a protocol. Hence, our approach can be applied in automated distributed-system verification. Given a set of inputs, a protocol and a number r of rounds, an automated process can create an initial epistemic model, update the model r times, and verify system properties.

A non-standard point in the parametrized logics is that formulas are interpreted on epistemic models for distributed systems, instead of on usual epistemic models. The definitions of such logics, however, can be presented in a standard way. The local states can be represented with sets of propositions and the local state change can be modelled with assignments or postconditions as in [39, 44]. A straightforward way to make this change is as follows. First, use a (possibly countably infinite) set of local propositions P_{a_i} for each agent $a_i \in A$ so that $P_{a_1}, P_{a_1}, \ldots, P_{a_n}$ partitions P. A local state of a_i will be described by a subset of P_{a_i}. Second, change the description of the protocols so that μ, and λ change their signs as follows: $\mu : P \to (Msg \cup \{\bot\})^n$, and $\lambda : P \times (Msg \cup \{\bot\})^n \to P$. $\mu(L(w) \cap P_{a_i})$ calculates a tuple with the messages that a_i sends to the other agents in world w. $\lambda(L(w) \cap P_{a_i}, m^w_{a_i})$ calculates the finite set of true valued propositions that describes the new

local state of a_i, where $m^w_{a_i}$ is a tuple such that $m^w_{a_i}(j)$ is the message that a_i receives from a_j in world w. $m^w_{a_i}$ can be defined in terms of μ. Finally, the product must be modified so that $L'(w,G) = \bigcup_{a\in A} \lambda(L(w) \cap P_a, m^w_a)$ and $(w,G) \sim'_a (w',G')$ if and only if $L(w,G) \cap P_a = L(w',G') \cap P_a$.

The dynamic epistemic logics proposed in this thesis provide a different way of modelling distributed computing scenarios separating the description of the distributed computing models, that relies on the pattern models, and the protocols. The protocol is arbitrary in the parametrized logics or is the full-information protocol in the non-parametrized formalism. Such a separation makes the logics more practical compared to action models, that make the analysis more complex for finding compact representations.

An alternative approach to modeling distributed systems epistemically is via interpreted systems, as in [13], [24], or [32]. In these works, protocols are modeled explicitly, and indistinguishability is generated directly from the local states; consequently there is no need for a communication pattern model (or an action model) that models the dynamics of the system. Since we use epistemic models (or epistemic models for distributed systems) and pattern models (or parametrized pattern models), we need to show that the indistinguishability relation that they generate coincides with the one based on local states in the corresponding model, which is shown in Theorems 3.4.2 and 5.5.1.

We may note now that interpreted systems and parametrized pattern models are similar formalisms: both work with configurations. There are differences, however. In the formalism of interpreted systems, we first select the executions to be analyzed and then we label the configurations with propositional variables. In the case of parametrized pattern models, by contrast, we label the worlds at the beginning describing the input of the processes and then we update the epistemic model for distributed systems r times. Thus, we may end with worlds w and w' such that $\mathcal{S}^r(w,a) = \mathcal{S}^r(w',a)$ for all $a \in A$ and $L^r(w) \neq L^r(w')$. This first difference may be removed by adding ontic change to the formalism as in [39, 44]. Another difference is that once we choose the executions in interpreted systems, the analysis considers configurations after an arbitrary number of rounds of communication, while in the case of parametrized pattern models we analyze configurations in the r-th round of communication with the possibility of further updates. Finally, it is important to note that an interpreted system may be restricted to studying a subset of all executions while parametrized pattern models are designed to study all executions that start given a set of input vectors.

6.3 Future research

Now, let us present some lines for future research. First, in contrast to pattern models logic of Ch. 3, in the parametrized pattern models logic in Ch. 5, an update modality is not part of the logic. We treat the updates as an external procedure because we do not have an axiomatization for the parametrized logic with the update modality. For the case of parametrized pattern models, it is not clear that the logic with update modality has an axiomatization that reduces the logic to epistemic logic. This is an interesting problem to study.

Second, logics where agents may die as in [23, 40] are static, namely they do not have update modalities. Proposing new logics, similar to these ones, adding update modalities is another problem that we may want to study.

Third, in dynamic-network models, the communication is synchronous. In principle, we could only describe synchronous distributed-computing models. However, IIS, that is an asynchronous model of computation, can be described using dynamic-network models. Then, another avenue of research would be to find the properties that an asynchronous model must have to be describable using dynamic-network models.

Bibliography

[1] Yehuda Afek, Hagit Attiya, Danny Dolev, Eli Gafni, Michael Merritt, and Nir Shavit. Atomic snapshots of shared memory. *J. ACM*, 40(4):873–890, 1993.

[2] Yehuda Afek and Eli Gafni. Asynchrony from synchrony. In Davide Frey, Michel Raynal, Saswati Sarkar, Rudrapatna K. Shyamasundar, and Prasun Sinha, editors, *International Conference on Distributed Computing and Networking (ICDCN 2013)*, pages 225–239, 2013.

[3] Bowen Alpern and Fred B. Schneider. Defining liveness. *Inf. Process. Lett.*, 21(4):181–185, 1985.

[4] Hagit Attiya and Jennifer Welch. *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*. Wiley Series on Parallel and Distributed Computing. Wiley, 2004.

[5] Alexandru Baltag, Lawrence S. Moss, and Slawomir Solecki. The logic of public announcements, common knowledge, and private suspicions. In *Proceedings of the 7th Conference on Theoretical Aspects of Rationality and Knowledge (TARK 1998)*, pages 43–56, 1998.

[6] Alexandru Baltag and Sonja Smets. Learning what others know. In Elvira Albert and Laura Kovacs, editors, *LPAR23. LPAR-23: 23rd International Conference on Logic for Programming, Artificial Intelligence and Reasoning*, volume 73 of *EPiC Series in Computing*, pages 90–119. EasyChair, 2020.

[7] Martin Biely, Peter Robinson, Ulrich Schmid, Manfred Schwarz, and Kyrill Winkler. Gracefully degrading consensus and k-set agreement in directed dynamic networks. *Theor. Comput. Sci.*, 726:41–77, 2018.

[8] Adam Bjorndahl and Will Nalls. Endogenizing epistemic actions. *Studia Logica*, Mar 2021.

[9] Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2001.

[10] Elizabeth Borowsky and Eli Gafni. Generalized FLP impossibility result for t-resilient asynchronous computations. In *Proceedings of the Twenty-Fifth ACM Symposium on Theory of Computing (STOC 1993)*, pages 91–100, 1993.

[11] Elizabeth Borowsky and Eli Gafni. Immediate atomic snapshots and fast renaming (extended abstract). In *Proceedings of the Twelth Annual ACM Symposium on Principles of Distributed Computing (PODC 1993)*, pages 41–51, 1993.

[12] Elizabeth Borowsky and Eli Gafni. A simple algorithmically reasoned characterization of wait-free computation (extended abstract). In *Proceedings of the Sixteenth Annual ACM Symposium on Principles of Distributed Computing (PODC 1997)*, pages 189–198. ACM, 1997.

[13] Armando Castañeda, Yannai A. Gonczarowski, and Yoram Moses. Unbeatable consensus. *Distributed Comput.*, 35(2):123–143, 2022.

[14] Armando Castañeda, Hans van Ditmarsch, David A. Rosenblueth, and Diego A. Velázquez. Communication pattern logic: Epistemic and topological views, 2022. https://arxiv.org/abs/2207.00823.

[15] Armando Castañeda, Hans van Ditmarsch, David A. Rosenblueth, and Diego A. Velázquez. Communication pattern logic: Epistemic and topological views. *Journal of Philosophical Logic*, Jul 2023.

[16] Armando Castañeda, Hans van Ditmarsch, David A. Rosenblueth, and Diego A. Velázquez. Comparing the update expressivity of communication patterns and action models. In *Proceedings Nineteenth conference on Theoretical Aspects of Rationality and Knowledge, Oxford, United Kingdom, June 28-30*, pages 157–172. Open Publishing Association, Waterloo, Australia, 2023.

[17] Armando Castañeda, Hans van Ditmarsch, David A Rosenblueth, and Diego A Velázquez. Pattern Models: A Dynamic Epistemic Logic For Distributed Systems. *The Computer Journal*, page bxae016, 02 2024.

[18] Bernadette Charron-Bost and André Schiper. The heard-of model: Computing in distributed systems with benign faults. *Distributed Comput.*, 22(1):49–71, 2009.

[19] Étienne Coulouma, Emmanuel Godard, and Joseph G. Peters. A characterization of oblivious message adversaries for which consensus is solvable. *Theor. Comput. Sci.*, 584:80–90, 2015.

[20] Ronald Fagin, Joseph Y. Halpern, Yoram Moses, and Moshe Y. Vardi. *Reasoning about Knowledge*. MIT Press, 1995.

[21] Tristan Fevat and Emmanuel Godard. Minimal obstructions for the coordinated attack problem and beyond. In *25th IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2011, Anchorage, Alaska, USA, 16-20 May, 2011 - Conference Proceedings*, pages 1001–1011. IEEE, 2011.

[22] Eric Goubault, Jérémy Ledent, and Sergio Rajsbaum. A simplicial complex model for dynamic epistemic logic to study distributed task computability. In *Proceedings of the Ninth International Symposium on Games, Automata, Logics, and Formal Verification (GandALF 2018)*, pages 73–87, 2018.

[23] Éric Goubault, Jérémy Ledent, and Sergio Rajsbaum. A Simplicial Model for KB4_n: Epistemic Logic with Agents That May Die. In *39th International Symposium on Theoretical Aspects of Computer Science (STACS 2022), Marseille, France, March 15-18*, pages 33:1–33:20. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2022.

[24] Joseph Y. Halpern and Yoram Moses. Knowledge and common knowledge in a distributed environment. In *Proceedings of the Third ACM Symposium on Principles of Distributed Computing (PODC 1984)*, pages 50–61, 1984.

[25] Joseph Y. Halpern and Yoram Moses. Knowledge and common knowledge in a distributed environment. *J. ACM*, 37(3):549–587, 1990.

[26] Maurice Herlihy, Dmitry Kozlov, and Sergio Rajsbaum. *Distributed Computing Through Combinatorial Topology*. Morgan-Kaufmann, 2014.

[27] Maurice Herlihy and Nir Shavit. The asynchronous computability theorem for t-resilient tasks. In *Proceedings of the Twenty-Fifth ACM Symposium on Theory of Computing (STOC 1993)*, pages 111–120, 1993.

[28] Maurice Herlihy and Nir Shavit. *The Art of Multiprocessor Programming, Revised Reprint*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1st edition, 2012.

[29] Michael Huth and Mark Ryan. *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press, USA, 2004.

[30] Fabian Kuhn and Rotem Oshman. Dynamic networks: models and algorithms. *SIGACT News*, 42(1):82–96, 2011.

[31] Petr Kuznetsov, Thibault Rieutord, and Yuan He. An asynchronous computability theorem for fair adversaries. In Calvin Newport and Idit Keidar, editors, *Proceedings of the 2018 ACM Symposium on Principles of Distributed Computing, PODC 2018, Egham, United Kingdom, July 23-27, 2018*, pages 387–396. ACM, 2018.

[32] Yoram Moses. Relating knowledge and coordinated action: The knowledge of preconditions principle. *Electronic Proceedings in Theoretical Computer Science*, 215:231–245, Jun 2016.

[33] Thomas Nowak, Ulrich Schmid, and Kyrill Winkler. Topological characterization of consensus under general message adversaries. In Peter Robinson and Faith Ellen, editors, *Proceedings of the 2019 ACM Symposium on Principles of Distributed Computing (PODC 2019)*, pages 218–227. ACM, 2019.

[34] Marshall C. Pease, Robert E. Shostak, and Leslie Lamport. Reaching agreement in the presence of faults. *J. ACM*, 27(2):228–234, 1980.

[35] Daniel Pfleger and Ulrich Schmid. On knowledge and communication complexity in distributed systems. In *International Colloquium on Structural Information and Communication Complexity (SIROCCO 2018)*, pages 312–330. Springer, 2018.

[36] Floris Roelofsen. Distributed knowledge. *Journal of Applied Non-Classical Logics*, 17(2):255–273, 2007.

[37] Nicola Santoro and Peter Widmayer. Time is not a healer. In Burkhard Monien and Robert Cori, editors, *STACS 89, 6th Annual Symposium on Theoretical Aspects of Computer Science, Paderborn, FRG, February 16-18, 1989, Proceedings*, volume 349 of *Lecture Notes in Computer Science*, pages 304–313. Springer, 1989.

[38] Ulrich Schmid, Bettina Weiss, and Idit Keidar. Impossibility results and lower bounds for consensus under link failures. *SIAM J. Comput.*, 38(5):1912–1951, 2009.

[39] Johan van Benthem, Jan van Eijck, and Barteld Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.

[40] Hans van Ditmarsch. Wanted dead or alive: Epistemic logic for impure simplicial complexes. In *27th International Workshop on Logic, Language, Information, and Computation, Virtual Event, October 5-8*, pages 31–46. Springer International Publishing, Cham, Switzerland, 2021.

[41] Hans van Ditmarsch, David Fernández-Duque, and Wiebe Hoek. On the definability of simulation and bisimulation in epistemic logic. *Journal of Logic and Computation*, 24:1209–1227, 11 2012.

[42] Hans van Ditmarsch, Éric Goubault, Jérémy Ledent, and Sergio Rajsbaum. Knowledge and simplicial complexes. In Björn Lundgren and Nancy Abigail Nuñez Hernández, editors, *Philosophy of Computing*, pages 1–50, Cham, 2022. Springer International Publishing.

[43] Hans van Ditmarsch, Joseph .Y. Halpern, Wiebe van der Hoek, and Barteld Kooi. An introduction to logics of knowledge and belief. In Hans van Ditmarsch, Joseph .Y. Halpern, Wiebe van der Hoek, and Barteld Kooi, editors, *Handbook of epistemic logic*, pages 1–51, 2015.

[44] Hans van Ditmarsch, W. van der Hoek, and B. P. Kooi. Dynamic epistemic logic with assignment. In *Proceedings of the Fourth International*

Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS '05, page 141–148, New York, NY, USA, 2005. Association for Computing Machinery.

[45] Hans van Ditmarsch, Wiebe van der Hoek, Barteld Kooi, and Louwe B. Kuijer. Arrow update synthesis. *Information and Computation*, page 104544, 2020.

[46] Diego A. Velázquez, Armando Castañeda, and David A. Rosenblueth. Communication pattern models: An extension of action models for dynamic-network distributed systems. In Joseph Y. Halpern and Andrés Perea, editors, *Proceedings Eighteenth Conference on Theoretical Aspects of Rationality and Knowledge, TARK 2021, Beijing, China, June 25-27, 2021*, volume 335 of *EPTCS*, pages 307–321, 2021.

[47] Yanjing Wang and Qinxiang Cao. On axiomatizations of public announcement logic. *Synthese*, 190(1 supp.):103–134, 2013.

[48] Yì N. Wáng and Thomas Ågotnes. Public announcement logic with distributed knowledge: expressivity, completeness and complexity. *Synthese*, 190(Suppl.-1):135–162, 2013.

[49] Kyrill Winkler, Ami Paz, Hugo Rincon Galeana, Stefan Schmid, and Ulrich Schmid. The Time Complexity of Consensus Under Oblivious Message Adversaries. In *14th Innovations in Theoretical Computer Science Conference (ITCS 2023), Cambridge, Massachusetts, January 10-13*, pages 100:1–100:28. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2023.

[50] Kyrill Winkler, Manfred Schwarz, and Ulrich Schmid. Consensus in rooted dynamic networks with short-lived stability. *Distributed Comput.*, 32(5):443–458, 2019.

[51] Michael Zacks and Fotios Zaharoglou. Wait-free k-set agreement is impossible: the topology of public knowledge. In *Proceedings of the Twenty-Fifth ACM Symposium on Theory of Computing (STOC 1993)*, pages 101–110, 1993.

[52] Thomas Ågotnes and Yì N. Wáng. Resolving distributed knowledge. *Artificial Intelligence*, 252:1–21, 2017.