Chapter 9
Temporal Aspects of the Dynamics of Knowledge

Valentin Goranko and Eric Pacuit

Abstract Knowledge and time are fundamental aspects of agency and their interaction is in the focus of a wide spectrum of philosophical and logical studies. This interaction is two-fold: on the one hand, knowledge evolves over time; on the other hand, in the subjective view of the agent, time only passes when her knowledge about the world changes. In this chapter we discuss models and logics reflecting the temporal aspects of the dynamics of knowledge and offer some speculations and ideas on how the interaction of temporality and knowledge can be systematically treated.

9.1 Introduction

Knowledge and time are fundamental aspects of agency and their interaction is in the focus of a wide spectrum of philosophical and logical studies. This interaction is a two-way street. On the one hand, knowledge evolves over time as the agent may learn new information, but also forgets previously known facts or their truth value may change over time. On the other hand, one can argue that, in the subjective view of the agent, time only passes when her knowledge about the world changes; and this is certainly the case when the agent has a watch and continuously keeps herself aware of the current time, but also when the agent has no other concept or measure of time except as a succession of events.

V. Goranko (✉)
Technical University of Denmark, 2800 Kongens Lyngby, Denmark
e-mail: vfgo@dtu.dk

V. Goranko
University of Johannesburg, Auckland Park, Johannesburg 2006, South Africa

E. Pacuit
Department of Philosophy, University of Maryland, College Park, MD, USA

A. Baltag and S. Smets (eds.), *Johan van Benthem on Logic and Information Dynamics*, Outstanding Contributions to Logic 5,
DOI: 10.1007/978-3-319-06025-5_9,

Each of these two concepts—knowledge and time—has been extensively formalized and studied in various logical frameworks, respectively forming the families of epistemic and temporal logics, since the 1960s, starting with the seminal works of Hintikka [45] and Prior [54]. Adding a multi-agent perspective makes the interaction between knowledge and time much more complex and versatile not only because of the intrinsic complexity of multi-agent epistemics, but also because of the partial information that individual agents have about the actual succession of events, and the problems arising with the synchronization of their communication.

Studies in formal logic gradually started reflecting on that interaction and a variety of logics combining knowledge and temporality started appearing in the 1980s. First, these were temporal-epistemic (aka, epistemic-temporal) logics [29, 30, 40, 41, 43, 53], coming mainly from the field of distributed computing and looking at the purely observable, explicit effect of the change of knowledge over time, but not at the reasons for such changes. Later, dynamic-epistemic logics emerged with the idea to focus on the causal aspects of the dynamics of knowledge, while leaving the temporality implicit, simply as succession of epistemic updates [3, 12, 26]. A strong impetus for new developments in logic reflecting the interaction between time, agents' knowledge and agents' abilities came from multi-agent systems where the so called "alternating-time temporal logic" [2] and numerous variations and extensions emerged in the early 2000s [36, 46, 60]. Another line of discussions related to issues that arise when developing logics that combine knowledge and temporality is in chapters on modal logics for reasoning about *strategies* and strategic analysis of multi-agent protocols in game situations [15, 49, 51]. A few approaches for combining the temporal, epistemic, and dynamic aspects have been recently proposed, too (see Sect. 9.6), but no commonly embraced Unification Theory has emerged yet. In this chapter we do not propose such theory, either, but rather offer some speculations on how it might look, and make some steps towards it.

This chapter is devoted to Johan van Benthem's influential contributions in this area. They go back at least to the 1990s and were first presented as a systematic research program in [9], later followed, inter alia, by [6, 10–13, 17] and culminated, so far, in [14] and [16]. While we do not purport here to survey Johan's work in this area, we certainly derive inspiration from it to chart some follow-up developments.

9.2 Preliminaries

We provide here only some basics of epistemic and temporal models and logics. For more details see the chapter [19] and the references at the end.

9.2.1 Models and Logics of (Static) Knowledge

9.2.1.1 Relational Models

The models that we discuss in this chapter are all instances of a *relational model*. Let At be a (finite) set of atomic sentences. A *relational model* (based on At) is a tuple $\langle W, \mathscr{R}, V\rangle$ where W is a nonempty set whose elements are called *possible worlds* or *states*; $\mathscr{R}$ is a set of relations on W, i.e., for each $R \in \mathscr{R}$, $R \subseteq W \times W$; and $V : \mathsf{At} \to \wp(W)$ is a valuation function mapping atomic propositions to sets of states. Elements $p \in \mathsf{At}$ are intended to describe ground facts about the situation being modeled, such as "the red card is on the table". The set W is intended to represent the different possible "scenarios" (elements of W are called possible worlds or states). The valuation function V associates with every ground fact the set of situations where that fact holds.

9.2.1.2 Epistemic Models

A basic epistemic model is a relational model with a single relation R, hereafter denoted as $\sim$, which represents the agent's knowledge, in terms of its epistemic uncertainties, as traditional in epistemic logic. The relation $\sim$ is the agent's *indistinguishability relation*, i.e., $q \sim q'$ means that the agent(s) is (are) not able to discern between the possible worlds q and q'; thus, both worlds appear identical from the agent's perspective. The indistinguishability relations representing the epistemic uncertainties are traditionally assumed to be *equivalence relations*. The knowledge of the agent is then determined as follows: the agent *knows* a property O in the world q if O is the case in all states indistinguishable from q for that agent.[1]

A multi-agent epistemic model involves an indistinguishability relation $\sim_i$ for every agent i. Formally, a *multi-agent epistemic structure* is a tuple $\mathscr{S} = \langle \mathbb{A}, \mathsf{St}, \{\sim_i \mid i \in \mathbb{A}\}, V\rangle$ where $\mathbb{A}$ is the set of agents, St is the set of states (possible worlds) and $\sim_i$ is the indistinguishability relation over St associated with the agent i, for each $i \in \mathbb{A}$. Then, a *multi-agent epistemic model* (MAEM) is defined by adding a valuation to a multi-agent epistemic structure.

9.2.1.3 Epistemic Logics

Basic epistemic logic. A simple propositional modal language is often used to describe epistemic models. Let $\mathscr{L}_{EL}$ be the (smallest) set of sentences generated by the following grammar:

$$\varphi := p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathrm{K}\varphi$$

[1] More generally, by varying the properties of the relation R, these models can also represent other informational attitudes of the agent, such as beliefs.

where $p \in \mathsf{At}$ (the set of atomic propositions). The additional propositional connectives ($\rightarrow, \leftrightarrow, \vee$) are defined as usual and the dual of K, often denoted L, is defined as follows: $\mathrm{L}\varphi := \neg\mathrm{K}\neg\varphi$. The intended interpretation of $\mathrm{K}\varphi$ is "according to the agent's current (hard) information, φ is true" (more standardly "the agent knows that φ is true"). Truth of the above language is defined as follows: Let $\mathcal{M} = \langle W, \sim, V\rangle$ be an epistemic model. For each $w \in W$, φ *is true at state* w, denoted $\mathcal{M}, w \models \varphi$, is defined by induction on the structure of φ:

- $\mathcal{M}, w \models p$ iff $w \in V(p)$
- $\mathcal{M}, w \models \neg\varphi$ iff $\mathcal{M}, w \not\models \varphi$
- $\mathcal{M}, w \models \varphi \wedge \psi$ iff $\mathcal{M}, w \models \varphi$ and $\mathcal{M}, w \models \psi$
- $\mathcal{M}, w \models \mathrm{K}\varphi$ iff for all $v \in W$, if $w \sim v$ then $\mathcal{M}, v \models \varphi$ □

Multi-agent epistemic logics. Besides the individual knowledge for each agent, the multi-agent epistemic framework involves several very natural and important notions of *multi-agent knowledge* and respective knowledge operators, for every non-empty set of agents A. These operators with their intended interpretations are:

- $\mathrm{K}_A\varphi$, saying *'Every agent in the group A knows that* φ'. When $A = \{i\}$ we write K_i instead of $K_{\{i\}}$.
- $\mathrm{D}_A\varphi$, saying *'It is a distributed knowledge amongst the agents in the group A that* φ', intuitively meaning that the collective knowledge of all agents in the group A implies φ. For instance, if $\mathrm{K}_a\varphi$ and $\mathrm{K}_b(\varphi \rightarrow \psi)$ hold for some agents a and b, then $\mathrm{K}_{a,b}(\varphi \wedge (\varphi \rightarrow \psi))$ holds, and therefore $\mathrm{K}_{a,b}\psi$ holds, too, by the closure of knowledge under logical consequence.
- $\mathrm{C}_A\varphi$, saying *'It is a common knowledge amongst the agents in the group A that* φ', intuitively meaning that not only every agent in A knows φ, but also that every agent in A knows that every agent in A knows φ, and every agent in A knows that, etc., ad infinitum.

The language of the multi-agent epistemic logic builds on the basic epistemic logic by adding some or all of these operators and the formulae are defined by the following recursive definition:

$$\varphi := p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathrm{D}_A\varphi \mid \mathrm{C}_A\varphi,$$

where p ranges over At and A ranges over the set $\mathcal{P}^+(\mathbb{A})$ of non-empty subsets of $\mathbb{A}$. The individual knowledge $\mathrm{K}_i\,\varphi$ is definable as $\mathrm{D}_{\{i\}}\varphi$, and then the group knowledge $\mathrm{K}_A\varphi$ is definable as $\bigwedge_{i\in\mathbb{A}} \mathrm{K}_i\,\varphi$.

The formal semantics of the multi-agent epistemic operators at a state in a multi-agent epistemic model $\mathcal{M} = (\mathbb{A}, \mathsf{St}, \{\sim_i \mid i \in \mathbb{A}\}, V)$ is given by the clauses:

($\mathrm{K}_{\mathbf{A}}$) $\mathcal{M}, q \models \mathrm{K}_{\mathsf{A}}\varphi$ iff $\mathcal{M}, q' \models \varphi$ for all q' such that $q \sim^E_{\mathsf{A}} q'$, where $\sim^E_{\mathsf{A}} = \bigcup_{\mathsf{i}\in\mathsf{A}} \sim_{\mathsf{i}}$.

($\mathrm{C}_{\mathbf{A}}$) $\mathcal{M}, q \models \mathrm{C}_{\mathsf{A}}\varphi$ iff $\mathcal{M}, q' \models \varphi$ for all q' such that $q \sim^C_{\mathsf{A}} q'$, where $\sim^C_{\mathsf{A}}$ is the transitive closure of $\sim^E_A$.

($\mathbf{D_A}$) $\mathscr{M}, q \models \mathsf{D}_\mathsf{A}\varphi$ iff $\mathscr{M}, q' \models \varphi$ for all q' such that $q \sim^D_\mathsf{A} q'$, where $\sim^D_\mathsf{A} = \bigcap_{\mathsf{i}\in\mathsf{A}} \sim_\mathsf{i}$.

For more a more detailed discussion on epistemic models and the relevant modal logics see the chapter [19].

9.2.2 Temporal Models and Logics

Temporal reasoning stems from philosophical analysis of time and temporality, initiated in the Antiquity by Diodorus Chronos and Aristotle, but only formalized in precise logical terms first by Arthur Prior in his historical work culminating with his seminal book "Past, Present and Future" [54].

9.2.2.1 Temporal Models

There are various ontological assumptions for the nature of time, reflecting on the types of time flows and models used to formalize temporal reasoning: instant-based or interval-based, discrete or dense, continuous or not, endless or not, linear or branching, etc. The simplest formal model of time, aka *temporal frame*, is $\langle T, \preceq\rangle$ where T is a nonempty set of *time instants* or *moments* and $\preceq$ is a time precedence relation on T, which is generally a partial order, often assumed linear or tree-like, i.e., every time instant having a linearly ordered by $\preceq$ set of predecessors. More abstractly, one can adopt time intervals (periods) or entire time histories as primitive temporal entities and build models based on these, as demonstrated in [7].

In our context here, time is not an abstract flow of moments but rather a metaphor for the discrete succession of events—explicit or implicit—that determine the time instants and represents the passing of time. Such time flow can be linear, corresponding to a single time line (trace, history, etc.) or branching, corresponding to a non-deterministically evolving future of possible succession of events. Thus, the only observable effect of time passing is a discrete transition of one 'snapshot' of the world to another.

Depending on whether one is more interested in the sequence of events causing the passing of time or in the actual sequence of time states ('snapshots') of the world, this concept of time formalizes in either event-based temporal structures (sometimes called 'protocols') or in (temporal) transition systems. The former are more prominent in theories of events and agency as well as in distributed computing, whereas the latter are fundamental for the applications of temporal logics in computer science for model verification. Event-based protocols will be introduced later, and here we briefly present the basics of transition systems.

9.2.2.2 Transition Systems and Computations in Them

Formally, these are simply relational frames, consisting of a set of states and transition relations between them, possibly labelled by different types of actions. The states in a transition system can be thought of as program states, control states, configuration states, memory registers, etc. The actions can represent agents' actions, autonomous processes, or simply program instructions. Formally, a *labeled transition system* is a structure $\mathscr{T} = \langle \mathsf{St}, \{\overset{a}{\longmapsto}\}_{a \in \mathsf{Act}} \rangle$ consisting of a non-empty set St of *states*; a non-empty set Act of *actions* or *transitions*, and a binary *transition relation* $\overset{a}{\longmapsto} \subseteq \mathsf{St} \times \mathsf{St}$ associated with every action $a \in \mathsf{Act}$. The intuition is that each $a \in \mathsf{Act}$ acts, possibly non-deterministically, on states and produces *successor states*. We write $s \overset{a}{\longmapsto} t$ to indicate that the action a can transform the state s into the state t and say that s is an a*-predecessor* of t, while t is an a*-successor* of s. The successor relation between states generates a branching time discrete temporal structure. A labelled transition system that involves only one type of action is called a *simple transition system*, or just a *transition system* simpliciter. Then we omit the label and typically denote it by (St, R),

A state may have various properties. For instance, a state of a transition system modeling a computing process can be initial, accepting, safe or unsafe, critical or terminal for a given process, etc. Such properties of states can be indicated by special *atomic propositions*. The set of such propositions that are declared true at a given state is the *description* of that state. A transition system where every state is assigned such description is an *interpreted transition system*. Formally, this is a pair $\mathscr{M} = \langle \mathscr{T}, \mathsf{L} \rangle$ where $\mathscr{T}$ is a transition system and $\mathsf{L} : \mathsf{St} \to \mathbf{2}^{\mathsf{At}}$ is a *state description mapping* that assigns to every state s the set of atomic propositions from a fixed set At, that are true at s. Abstractly, interpreted transition systems are simply relational models, with valuation uniquely derived from the state description.

A *path (run, execution)* in a transition system $\mathscr{T}$ is a (finite or infinite) sequence of states and actions transforming every state into its successor: $s_0 \overset{a_0}{\longmapsto} s_1 \overset{a_1}{\longmapsto} s_2 \ldots$. Thus, a path is a linear time flow representing a possible time history.

A *computation*, or *trace*, in an interpreted transition system $(\mathscr{T}, \mathsf{L})$ is a (finite or infinite) sequence of state descriptions and respective actions along a path: $\mathsf{L}(s_0) \overset{a_0}{\longmapsto} \mathsf{L}(s_1) \overset{a_1}{\longmapsto} \mathsf{L}(s_2) \ldots$. Thus, a computation, intuitively, is the *observable effect* (the 'trace') of a path in a transition system. It can be regarded as a record of all successive intermediate results of the computing process. The idea is that the information encoded by the state descriptions includes all that is essential in the computation, including the values of all important variables. However, agents typically can only observe part of the state description, which represents their current information about the world. Usually, unless otherwise specified, we assume that the transition relation R is *serial*, or *total*, i.e., every state has at least one R-successor. When the actions are not important, one can represent paths and computations simply as sequences $s_0, s_1, s_2, \ldots$ and respectively $\mathsf{L}(s_0), \mathsf{L}(s_1), \mathsf{L}(s_2), \ldots$, or, more abstractly, as mappings $\sigma : \mathbb{N} \to \mathsf{St}$, respectively $\sigma : \mathbb{N} \to 2^{\mathsf{At}}$.

9.2.2.3 Basic Temporal Logic

The basic temporal language $\mathfrak{L}_t$, essentially due to Prior, is a propositional bimodal language, containing, besides a fixed set of atomic propositions and boolean connectives, the temporal operators H, G respectively referring to "always in the past" and "always in the future". The set of formulae is recursively defined by:

$$\varphi = p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid G\varphi \mid H\varphi$$

The dual temporal connectives, referring to "sometime in the past" and "sometime in the future", are defined as usual:

$$F\varphi := \neg G\neg\varphi, \; P\varphi := \neg H\neg\varphi.$$

Temporal model is a tuple $\langle T, \preceq, V\rangle$ where $\langle T, \preceq\rangle$ is a temporal frame and V is a valuation. Truth of a temporal formula at an instant t in a temporal model $M = \langle T, \preceq, V\rangle$ is defined in a traditional modal logic style, assuming that $\preceq$ is the accessibility relation associated with G and its converse $\succeq$ is associated with H:

- $M, t \models G\varphi$ if $M, s \models \varphi$ for every $s \in T$ such that $t \preceq s$.
- $M, t \models H\varphi$ if $M, s \models \varphi$ for every $s \in T$ such that $t \succeq s$.

Several additional temporal operators can be added, especially in a discrete setting, such as "Nexttime" N, "Since" S, "Until" U, etc. For further general references on temporality and temporal logics see [4, 7, 8].

9.3 From Static to Dynamic Reasoning about Knowledge: Temporal-Epistemic Frameworks

The traditional temporal models implicitly assume complete and fixed, unchangeable knowledge of all agents at all times. Furthermore, the epistemic models described above are *static*, or rather *timeless*. They describe what the agents know and believe in a fixed 'snapshot of the world'. Thus, none of these reflects the deficiencies, nor the dynamics, of knowledge over time.

There are various ways to relate models of time with knowledge and provide a framework for logical reasoning about their interaction. The syntactic merger seems easy: one can simply put together the desired repertoires of temporal and epistemic operators in a common logical language. However, the conceptual modeling of their interaction and the formal semantics capturing that interaction are the main challenges. In this chapter, we discuss further some known and some new ideas of how that can be done. The relevant literature is rich and diverse, and we only mention and briefly discuss some selected sources further in the text. A discussion on some of these approaches can also be found in the chapter [19].

- One of the general formal construction is *fusion* [33] of temporal and epistemic models into temporal-epistemic models, where possible worlds are regarded both as time instants and as epistemic alternatives. This construction is technically simple and elegant but conceptually deficient because it neither reflects nor explains the *temporal dynamics* of knowledge.
- Another generic formal approach is *temporalization* of a logical system [32], in this case of the epistemic logic. Semantically, it is based on temporal-epistemic models obtained by taking a temporal model and associating every time point in it with an epistemic model.
- *Protocol-based epistemic temporal models* are refinements of the temporalization construction, obtained as collections of epistemic models, related over time by protocols assigning a model to every time instant. The protocols are directed trees representing the possible sequences of events (or, actions) effecting the evolution of knowledge.
- A more refined approach alternates adding temporal and epistemic layers. It starts e.g., with a temporal model and then adds a 'cloud' of epistemic alternatives representing the uncertainties for each agent at every moment of time. Further, all epistemic alternatives are "temporalized" by adding time stamps to each of them and then extended with full time lines, thus creating a bundle of interleaved temporal models over epistemic states. Then the resulting models are endowed with clouds of epistemic alternatives for each time moment, etc. The alternation of adding epistemic alternatives and time lines until saturation or forever. The limit of that construction is the intended temporal-epistemic model.
- A similar, yet somewhat technically different idea is implemented in Halpern and Vardi's *interpreted systems* (generalizing and extending "interpreted transition systems" as defined earlier) [30, 31] built on sets of 'runs', each representing a possible evolution in time of a system consisting of several processors running in parallel, each having their own local state and all these local states composing into a 'global state'. Every agent can only observe its local component of the state and this partial information creates the agent's epistemic indistinguishability relation.
- Alternatively, one can start with an epistemic model and then add a temporal structure to each of the possible worlds: Following this approach, a "possible world" is no longer a primitive object in the model. The "possible worlds" of the above model are constructed from more basic objects, such as *events*, *local states* and *moments*. Thus, one can describe the different model transformations that are intended to represent different "epistemic actions". This is a "change-based" view of knowledge dynamics.
- A global epistemic approach takes a class of pointed temporal models and adds epistemic uncertainties between them (for each agent), producing an epistemic 'super-model'.
- An epistemic analogue of STIT models, where the choice relation of an agent is interpreted as his epistemic relation over possible alternative futures, which changes dynamically over time as the future is gradually revealed. See [18] for an initial discussion of this idea.

In the rest of this section we present in more details and discuss some of these approaches.

9.3.1 From Adding Epistemic Clouds to Fusion of Temporal and Epistemic Models

There is a whole spectrum of possible extensions of a temporal model with an epistemic dimension, reflecting the agents' awareness and knowledge of time and the degree of synchrony between agents. The conceptually simplest approach is to start with a temporal model—be it for linear or branching time—and to expand every time instant in it with a 'cloud' of epistemic alternatives for each agent. The resulting formal models can be defined as $\langle T, \preceq, \{W_t\}_{t \in T}\rangle$ where $\langle T, \preceq\rangle$ is a temporal model and for each $t \in T$ the set W_t is an epistemic model consisting of the alternative worlds that some of the agents consider possible at moment t.

In order to give semantics of a temporal epistemic language in such models we have to restrict it to formulae where temporal operators cannot be nested in epistemic operators. The semantics is then a straightforward combination of the temporal and purely epistemic semantic clauses. With such language one can reason about what one will know tomorrow, sometime, or always, but not what one knows now about what will be true tomorrow, sometime, or always. In order to interpret such statements, and further nesting of temporal and epistemic operators, the models defined above must be enriched with time stamps for every epistemically alternative world, or with alternative time lines passing through them, and then adding epistemic clouds for each instant on these alternative time lines, then arranging these in timelines, etc. Eventually, in the limit of that construction we obtain a full fusion of temporal and epistemic models: $\langle \mathbb{A}, \mathsf{St}, \preceq, \{\sim_i \mid i \in \mathbb{A}\}, V\rangle$, where the possible worlds incorporate the time instants and $\preceq$ represents time precedence over a possible timeline in the model. Depending on how the epistemic alternatives relate to the temporal knowledge of the agents, a variety of models can emerge here.

One extremity is a *fully synchronous system* where there is a global clock observable by all agents at all times. In this case, all epistemic alternatives of a possible world in the temporal model share the same time stamp. These alternatives can, however, appear or disappear in time, as the agent learns or forgets. However, these epistemic alternatives may, but need not, evolve over time which therefore renders it possibly meaningless to reason about what the agent may know in the future or has known in the past.

The other extremity is a *fully asynchronous system* where the agents have no knowledge, or possibly not even concept, of time. In this case, their epistemic alternatives may have different time stamps, or no time stamps at all. One can imagine that in this case agents have instantaneous knowledge at every time instant, but no memory at all, and the evolution of their knowledge is exogenous for them.

Finally, we note one particular construction of a temporal model with 'epistemic clouds' based on *partial observability*: every agent can only observe the truth value of some atomic propositions, which naturally creates the cloud of alternative possibilities for the actual world, of which it has a partial view. Note that this partial observability can vary over time for each agent, thus creating more elaborate scenarios about the dynamics of their knowledge.

9.3.2 Interpreted Systems as Temporal-Epistemic Models

Various proposals in the distributed systems literature of the 1980s [41, 43, 53] gradually crystalized in Halpern and Vardi's *interpreted systems* [40, 42], further developed in [30]; see further references in the latter. Interpreted systems model the evolution of the knowledge of one or several agents (processors) over time and are technically very similar to the fusion of temporal and epistemic models discussed above.

9.3.2.1 Interpreted Systems

Informally, an *interpreted system* is defined for a fixed set of agents $\mathbb{A}$ and builds on a *state space* St consisting of *(global) states*, where a global state can be viewed as a tuple of *local states*, one for each agent. More generally, a global state can be regarded as an abstract entity, of which every agent only has a partial *local view*. This allows for some parts of the state to be visible by several agents, and others—possibly by none.

A basic concept in an interpreted system is a *run*: an infinite sequence of global states from St; formally, a mapping $r{:}\mathbb{N} \to \mathsf{St}$. Generally, an interpreted system $\mathscr{I}$ may comprise any non-empty set $\mathscr{R}$ of runs on its state space. Then, a pair (r, n), where $r \in \mathscr{R}$ and $n \in \mathbb{N}$ is a *(time) point* on the run r. Thus, the set of points in $\mathscr{I}$ is $P(\mathscr{I}) = \mathscr{R} \times \mathbb{N}$. The point (r, n) corresponds to a unique state $r(n)$; however, different points may correspond to the same state.

The knowledge of every agent in an interpreted system is determined, as in pure epistemic logic, in terms of its uncertainty. The agent's uncertainty here is between different *time points*, however, Halpern and Vardi reduce it to uncertainty between states: two points t_1 and t_2 are *indistinguishable* for an agent i iff their corresponding states have the same local component for i, i.e., iff i has the same local view on them. This is denoted $t_1 \sim_i t_2$, where $\sim_i$ is the *indistinguishability relation* on $\mathscr{R} \times \mathbb{N}$ for i.

Finally, an interpreted system involves labeling of the points with sets of atomic propositions from a fixed set PROP. The label of a point is supposed to describe all essential features of that point. Thus, formally, an interpreted system is a tuple:

$$\mathscr{I} = \langle \mathbb{A}, \mathsf{St}, \mathscr{R}, \{\sim_i\}_{i \in \mathbb{A}}, \mathsf{L} \rangle$$

where $\mathbb{A}$ is the set of agents, St is the global state space, $\mathscr{R}$ is the set of runs, for each $i \in \mathbb{A}$ the relation $\sim_i$ is an equivalence relation of indistinguishability on $P(\mathscr{I}) = \mathscr{R} \times \mathbb{N}$ for the agent i, and $\mathsf{L} : P(\mathscr{I}) \to \mathrm{PROP}$ is the labeling function.

Note that for any given agent i, a run is a sequence of *local views*, that is, of point clusters with respect to $\sim_i$.

Given an interpreted system $\mathscr{I}$, we will refer to the relation $\{((r, n), (r, n+1)) \in P(\mathscr{I}) \times P(\mathscr{I}) \mid r \in \mathscr{R}, n \in \mathbb{N}\}$ as the *temporal relation* induced in $\mathscr{I}$.

A *computation* in the interpreted system $\mathscr{I}$ corresponding to the run $r : \mathsf{St} \to \mathbb{N}$ is the observable—by an external observer who has full view of all states— effect of that run: $\mathsf{L}(r(0)), \mathsf{L}(r(1)), \ldots$.

Now, the dynamics of the knowledge of an agent can be modeled in terms of the evolution of its local views in the course of a run or a computation. One can argue that the local view and, respectively, the knowledge of an agent should be based on the *label* of the current point, rather than on the point itself. This approach can be implemented by assigning to every agent i a subset PROP_i of *observable for i* atomic propositions [30, 43].

9.3.2.2 Some Important Properties of Interpreted Systems

Following [42] one can identify some key properties of interpreted systems that turn out to be crucial for the computational complexity of the problem of deciding satisfiability in them. An interpreted system $\mathscr{I} = \langle \mathbb{A}, \mathsf{St}, \mathscr{R}, \{\sim_i\}_{i \in \mathbb{A}}, \mathsf{L} \rangle$ has the property of:

- **Unique initial state** if $r(0) = r'(0)$ for all runs $r, r' \in \mathscr{R}$.
- **No forgetting** if for every $i \in \mathbb{A}$, if $((r, n) \sim_i (r', n'))$ then for all $k \leq n$ there exists a $k' \leq n'$ such that $((r, k) \sim_i (r', k'))$.
- **No learning** if for every $i \in \mathbb{A}$, if $((r, n) \sim_i (r', n'))$ then for all $k \geq n$ there exists a $k' \geq n'$ such that $((r, k) \sim_i (r', k'))$.
- **Synchrony** if for every $i \in \mathbb{A}$, if $((r, n) \sim_i (r', n'))$ then $n = n'$.

The property of **Synchrony** expresses the idea that the agents are able to perceive time and have a common clock. **No learning** expresses that agents do not learn over time, in the sense that if a coalition A of agents cannot distinguish two runs at a given time, it will not be able to do so later on. Likewise, **No forgetting** means that if A at a given time point can tell two different runs apart, it must have been able to do so at any previous point in time.[2]

9.3.2.3 Temporal-Epistemic Logics Over Interpreted Systems

Based on the choice of language: single-agent or multi-agent, linear time or branching time, including or not operators for common knowledge, as well as on the combi-

[2] This is somewhat more complicated in the asynchronous case, see [42, 44] for discussion and explanation.

nations of the semantic properties listed above, Halpern and Vardi identify in [40] and [42] a total of 96 temporal-epistemic logics and analyze the complexities of the satisfiability problems in them. The languages of these logics and their semantics are a fairly straightforward combination of the temporal and epistemic logics presented in Sect. 9.2. For instance, the formulae of the multi-agent linear time temporal epistemic logic with the temporal operators X ("next") and U ("until") of the logic LTL and individual and common knowledge (of all agents) operators are built as follows:

$$\varphi := p \mid \neg\varphi \mid (\varphi \wedge \varphi) \mid \mathrm{X}\varphi \mid (\varphi\, \mathrm{U}\varphi) \mid \mathrm{K}_i\, \varphi \mid \mathrm{C}\varphi$$

where $i \in \mathbb{A}$. The essential semantic clauses are:

$\mathscr{M}, (r, n) \models \mathrm{X}\varphi$ iff $\mathscr{M}, (r, n+1) \models \varphi$;
$\mathscr{M}, (r, n) \models \varphi\, \mathrm{U}\psi$ iff $\mathscr{M}, (r, i) \models \psi$ for some $i \geq n$ such that $\mathscr{M}, (r, j) \models \varphi$ for every $n \leq j < i$;
$\mathscr{M}, (r, n) \models \mathrm{K}_i\, \varphi$ iff $\mathscr{M}, (r', n') \models \varphi$ for every (r', n') such that $((r, n) \sim_i (r', n'))$.

For branching time logics, quantifiers over runs are added to the language, with semantics:
$\mathscr{M}, (r, n) \models \exists\varphi$ iff $\mathscr{M}, (r', n') \models \varphi$ for some r' such that $r(n) = r'(n')$.

Some of the properties listed above can be expressed by suitable axioms. For instance, **No learning** in linear-time interpreted systems corresponds to the axiom $\mathrm{XK}_i\, \varphi \to \mathrm{K}_i\, \mathrm{X}\varphi$, whereas **No forgetting** corresponds to $\mathrm{K}_i\, \mathrm{X}\varphi \to \mathrm{XK}_i\, \varphi$. For more, see [44] for the linear time logics and [48] for the branching time logics.

It turns out that when time and knowledge do not interact, or only weak forms of interaction are imposed (e.g., only synchrony) then these temporal-epistemic logics are computationally reasonably behaved, with EXPTIME complexity of their satisfiability/validity problems [40, 42] and allow relatively simple tableau-based decision procedures, even when the full epistemic repertoire, with common and distributed knowledge for each group of agents, is added to the language [38, 39]. However, most of these logics that involve more than one agent *whose knowledge interacts with time* (e.g., who do not learn or do not forget)—turn out undecidable (with common knowledge), or decidable but with non-elementary time lower bound (without common knowledge) [40, 42]. See [17] for a discussion of these issues. Even in the single-agent case, the interaction between knowledge and time proved to be quite costly (pushing the complexities of deciding satisfiability up to EXPSPACE and 2EXPTIME) (ibid.).

9.3.3 Protocol Based Epistemic-Temporal Models: Modeling Uncertainty About What Has Happened

This approach goes back to [53]; see also [52]. We fix a finite set of agents $\mathbb{A}$ and a (possibly infinite) set of events Σ. There is a large literature addressing the many subtleties surrounding the very notion of an *event* and when one event *causes* another event. However, for this chapter we take the notion of event as primitive. What is needed is that if an event takes place at some time t, then the fact that the event took place can be observed by a relevant set of agents at t. Compare this with the notion of an event from probability theory. If we assume that at each clock tick a coin is flipped exactly once, then "the coin landed heads" is a possible event. However, "the coin landed head more than tails" would not be an event, since it cannot be observed at any one moment. As we will see, the second statement will be considered a *property* of histories, or sequences of events. A Σ*-history* is a finite sequence of events from Σ. We write Σ^* for the set of Σ-histories. From now on we consider Σ fixed and call the elements of Σ^* just histories. For any history h, we denote by $\mathsf{len}(h)$ the length of h and we write he for the history h followed by the event e. Given $h, h' \in \Sigma^*$, we write $h \preceq h'$ if h is a prefix of h', and $h \prec_e h'$ if $h' = he$ for some event e.

There are several simplifying assumptions that we adopt. Since histories are sequences of (discrete) events, we assume the existence of a global discrete clock. The length of the history then represents the amount of time that has passed. Thus, this implies that we are assuming a finite past with a possibly infinite future. Furthermore, we assume that at each clock tick, or moment, *some* event—which need not be directly observable by any agent—takes place. Thus, we may include a special event e_t representing a "clock tick".

Definition 9.1 (*ETL Models*) Let Σ be a set of events and At a set of atomic propositions. An *epistemic temporal model (ETL model)* is a tuple $\langle \mathbb{A}, \Sigma, \mathsf{H}, \{\sim_i\}_{i\in\mathbb{A}}, V\rangle$ where $\mathbb{A}$ is a set of agents, H is a set of histories closed under prefixes, for each $i \in \mathbb{A}$, $\sim_i$ is an equivalence relation on H and V a valuation function ($V : \mathsf{At} \rightarrow \wp(\mathsf{H})$).

An ETL model describes how the agents' *knowledge* evolves over time. Formally, ETL models are very similar to interpreted systems, introduced in the previous section (consult [50] for an extended discussion). The domain of an ETL model (set of histories closed under non-empty prefixes) is called a *protocol*. Histories in a protocol are the analogues of the global states in an interpreted system. In addition, the protocol describes the temporal structure, with h' such that $h \prec_e h'$ representing the point in time after e has happened in h. The relations $\sim_i$ represent the uncertainty of the agents about how the current history has evolved. Thus, $h \sim_i h'$ means that from agent i's point of view, the history h' *looks the same as* the history h.

Assumptions about the domain of an ETL model corresponds to "fixing the playground" where the agents will interact. In other words, the protocol not only describes the temporal structure of the situation being modeled, but also any *causal* relationships between events (eg., sending a message must always preceed receiving that message) plus the motivations and dispositions of the participants (eg., liars send messages that they *know*—or believe—to be false). Thus the "knowledge" of agent i at a history h in an ETL model is derived from both i's observational powers (via the $\sim_i$ relation) and i's information about the "protocol" generating the histories in the model.

Analogously to properties of interpreted systems, we identify the following key properties of an ETL model: Let $\mathscr{M} = \langle \Sigma, \mathsf{H}, \{\sim_i\}_{i \in \mathbb{A}}, V \rangle$ be an ETL model. $\mathscr{M}$ satisfies:

- **Synchronicity** iff for all $h, h' \in \mathsf{H}$, if $h \sim_i h'$ then $\mathsf{len}(h) = \mathsf{len}(h')$
- **Perfect Recall** iff for all $h, h' \in \mathsf{H}$, $e, e' \in \Sigma$ with $he, h'e' \in \mathsf{H}$, if $he \sim_i h'e'$, then $h \sim_i h'$
- **Uniform No Miracles** iff for all $h, h' \in \mathsf{H}$, $e, e' \in \Sigma$ with $he, h'e' \in \mathsf{H}$, if there are $h'', h''' \in \mathsf{H}$ with $h''e, h'''e' \in \mathsf{H}$ such that $h''e \sim_i h'''e'$ and $h \sim_i h'$, then $he \sim_i h'e'$.

Note that the properties defined above only refer to the underlying *frames* of the ETL models.

Remark 9.1 (*Alternative Definition of Perfect Recall*) Johan van Benthem gives an alternative definition of Perfect Recall in [12]:

$$\text{if } he \sim_i h' \text{ then there is an event } f \text{ with } h' = h''f \text{ and } h \sim_i h''.$$

This property is equivalent over the class of ETL models to the above definition of Perfect Recall and synchronicity. The formulation of Perfect Recall given in the former definition above is closer to the one found in the computer science literature on verifying multiagent systems (cf. [30]) and the game theory literature (cf. [21]).

ETL models describe how the agents' knowledge changes during a given sequence of events. The example in Fig. 9.1 illustrates the type of knowledge flow that ETL models describe. Suppose that there is a deck of red and black cards. An agent is observing the cards being placed on a table. Suppose that a red card is placed face down on the table (the agent can see that there is a card on the table, but not the color of the card). The next two cards that will be chosen by the dealer are a black card followed by a red card. Furthermore, both cards will be placed faced up on the table. An ETL model describing this situation runs as follows: There are four events $\Sigma = \{\mathsf{R}_d, \mathsf{R}_u, \mathsf{B}_d, \mathsf{B}_u\}$ where R_d is the event[3] "a red card is placed on the table face down", R_u is the event "a red card is placed on the table face up" (similarly for B_d and B_u). We are interested in describing how the agent's knowledge changes during the history $h = \mathsf{R}_d\mathsf{B}_u\mathsf{R}_u$. The set H is the set of Σ-histories of length ≤ 3 depicted in Fig. 9.1. The dotted lines represent agent i's information cells. For a history h, we write $[h]_i$ for equivalence class of h under $\sim_i$. We make the following observations about this model, to be followed by some analysis in later sections. First

[3] To be more precise, R_d is an *event type* (similarly for the other events in Σ).

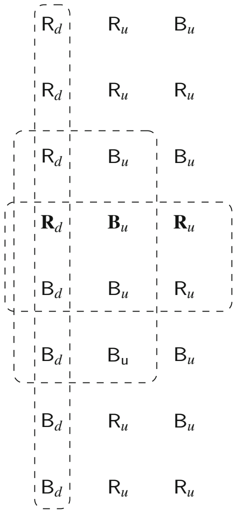


Fig. 9.1 Epistemic temporal model of the card dealing

some notation, if h is a history and $t \in \mathbb{N}$, we write h_t for the initial segment of h of length t and $h_{(t)}$ for the t^{th} event in h. The actual history is $h = \mathsf{R}_d \mathsf{B}_u \mathsf{R}_u$.

- Restricting the set of "admissible" histories H allows the agent to incorporate knowledge of the "rules of the game" into his information. For example, the agent "knows" that the game proceeds by putting one card face down on the table followed by two cards placed face up. Furthermore, we assume that the agent cannot distinguish the two event R_d and B_d; however she can distinguish between the two events R_u and B_u (as well as between R_u and R_d, for example).
- After the first card is placed on the table face down, the agent does not know whether the card is black or red. This follows since the equivalence class of h is $[h]_i = \{h'_1 \mid h' \in \mathsf{H}\}$.
- On history h, after the first card is placed on the table (at moment 1), it is settled that the next card will be black, but the agent does not know this. This follows since $h_{(2)} = \mathsf{B}_u$ (the next event will be that a black card is placed face up on the table) and there is a $h' \in \mathsf{H}$ such that $h_1 \sim_i h'_1$ and $h'_{(2)} = \mathsf{R}_u$.
- After the second card is placed on the table, the agent learns that it is a black card. This follows since $[h_2]_i = \{h' \in \mathsf{H} \mid h'_{(2)} = \mathsf{B}_u\}$.

9.3.4 Adding Epistemics to Temporal Models: Modeling Uncertainty About What Will Happen

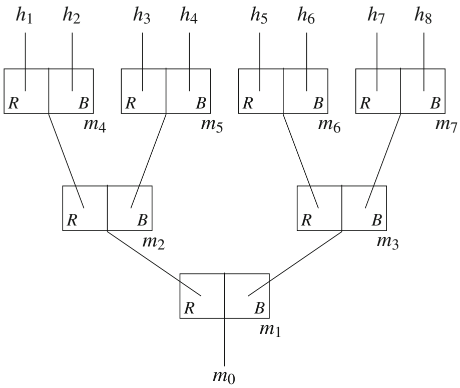
Let $\langle T, \preceq \rangle$ be a temporal model, e.g., T is a nonempty set of moments and $\preceq$ is the predecessor relation on T. A *full history h* is a maximal linearly ordered set of moments. Let $\mathscr{H}$ be the set of all full histories. For $h \in \mathscr{H}$, we write h/m for the pair (h, m) where $m \in h$. Each pair h/m is associated with a set of atomic propositions: the non-epistemic facts that are true at moment m given full history h. Let $\mathscr{H}_m = \{h \mid h \text{ is a full history with } m \in h\}$ be the set of full histories passing through moment m.

At each moment m there is a relation $\sim_i^m$ representing i's information uncertainties at moment m. This is intended to be a "forward-looking" notion of knowledge representing the information that agent i has about how the situation will evolve from moment m onwards. For simplicity, we assume that each $\sim_i^m$ is an equivalence relation, though this is not crucial for what follows. We further suppose that there is a distinguished moment $0 \in T$ representing the initial state of affairs. Then, $\sim_i^0$ represents i's initial uncertainty about all possible histories.

We have left open exactly which set of histories $\sim_i^m$ ranges over. There are two natural choices. The first choice is to let $\sim_i^m \subseteq \mathscr{H}_m \times \mathscr{H}_m$. So, if $h \sim_i^m h'$ then both h and h' are histories containing moment m. This builds in the assumption that the agent correctly observes all actions leading up to this moment. In addition, we may impose a stronger perfect recall condition:

- For all m and m', if $m \preceq m'$, then $\sim_i^{m'}$ is a *refinement* of $\sim_i^m$. I.e., if $h, h' \in \mathscr{H}_{m'}$ and $h \sim_i^{m'} h'$, then $h \sim_i^m h'$.

Recall the example discussed in the previous section. There is a deck of red and black cards, cards are being chosen one at a time and placed on a table in front of an agent i. A branching time model of this situation looks as follows:



There are eight moments $T = \{m_0, \ldots, m_7\}$. If we assume that for each $m \in T$, $\sim_i^m = \mathscr{H}_m \times \mathscr{H}_m$, then we have the following observations:

- At moment m_1, the agent does not know whether the card chosen by the dealer is red or black.
- At the pair h_1/m_1, the card chosen by the dealer is red, but the agent does not know this. Indeed, the agent thinks that it may be black.
- At moment m_3, the agent knows that the card chosen at moment m_1 was black, but does not know the card the dealer is currently holding.

- At m_1, it is not settled yet which card the dealer will choose during the next round and the agent knows this.

Note that we assume that $\sim_i^m \subseteq \mathscr{H}_m \times \mathscr{H}_m$, and so, the agent's uncertainty at moment m ranges only over the set of histories running through moment m. Alternatively, we can assume that the agents uncertainty ranges over all histories: for each $m \in T$, $\sim_i^m \subseteq \mathscr{H} \times \mathscr{H}$. A natural constraint here is:

- For all h, h' if $h \sim_i^m h'$ then there is some $m' \preceq m$ such that $h, h' \in \mathscr{H}_{m'}$ and $h \sim_i^{m'} h'$

This means that if the agent cannot distinguish between h and h' at moment m, there must be some earlier moment m' such that both h and h' run through m and the agent could not distinguish h and h' at that moment. The flow of knowledge can be described as follows:

- At m_1, we have $\sim_i^{m_1} = \mathscr{H} \times \mathscr{H}$.
- The equivalence classes of $\sim_i^{m_j}$ with $j = 2, 3$ are $\big\{\{h_1, h_2, h_5, h_6\}, \{h_3, h_4, h_7, h_8\}\big\}$
- The equivalence classes of $\sim_i^{m_j}$ with $j = 4, 5, 6, 7$ are $\big\{\{h_1, h_5\}, \{h_2, h_6\}, \{h_3, h_7\}, \{h_4, h_8\}\big\}$

Given these definitions, we make the following observations:

- At moment m_1 on history h_1, the card chosen by the dealer is red, but the agent does not know this (this follows since $h_1 \sim_i^{m_1} h_5$).
- at moment m_2 on history h_1, the card chosen by the dealer is red and the agent knows this (this follows since it is true on all the histories that are $\sim_i^{m_2}$-equivalent to h_1 the red card is chosen at moment m_2). Furthermore, the agent still does not know that the card chosen at m_1 was red (this follows since $h_1 \sim_i^{m_2} h_5$).

9.3.5 Comparing Modeling Formalisms

We conclude this section with some brief comparisons between the various epistemic temporal constructions and models.

- Interpreted systems are a special kind of fusion of temporal and epistemic models, and protocol-based models can be regarded as a special kind of interpreted systems, where the runs are chains of histories along branches of the protocol-tree.
- The temporal models extended with uncertainties between histories are technically closely related to STIT models, see [18] for an initial discussion. Indeed, one can simply take a STIT model and treat the partition of all histories passing through a given point determined by the possible choices of an agent as arising from the epistemic indistinguishability relation between these histories for the agent. Note that the standard additional requirement in STIT models, that every selection of choices by all agents intersects in a single history, can now be interpreted as saying

that the agents have a complete distributed knowledge about the entire actual future (the 'thin red line').

We also note that there is an important distinction in the STIT literature between "moments" and "instants". The general idea is that instants represents the general flow of time while moments are specific "realizations" of the instances. Formally, an instant i is a partition of the moments such that every history intersects each instant at exactly one moment (i.e., each $i \in \mathrm{i}$, for all $h \in \mathscr{H}$, $|i \cap h| = 1$). For example, in the above model m_2 and m_3 both occur at the first instant. We may be interested in an agent's knowledge at a particular instant: after the second card flip, the agent knows the color of the card: at each moment in the second instant, it is true that agent knows the color of the card.

9.4 Looking Inside the Dynamics of Knowledge: Dynamic Epistemic Logic

The models introduced in the previous section each provide a "grand stage" where histories of some social interaction unfold constrained by some underlying *protocol*. Temporal-epistemic models present the observable effect of the dynamics of knowledge over time but do not reflect the causes for that dynamics. In this section, we introduce an alternative framework to reason about the dynamics of knowledge. The focus in this section is on "epistemic actions" that transform models describing the agents' current information. A number of elegant logical systems have been devised to reason about such epistemic actions (see [14] and the chapters [28] and [1] for overviews).

Similar to the way relational structures are used to capture the information the agents have about a *fixed* social situation, an *event model* describes the agents' information about which actual events are currently taking place. The temporal evolution of the situation is then computed from some initial epistemic model through a process of successive model updates, effected by a product construction between the epistemic model and the event model.

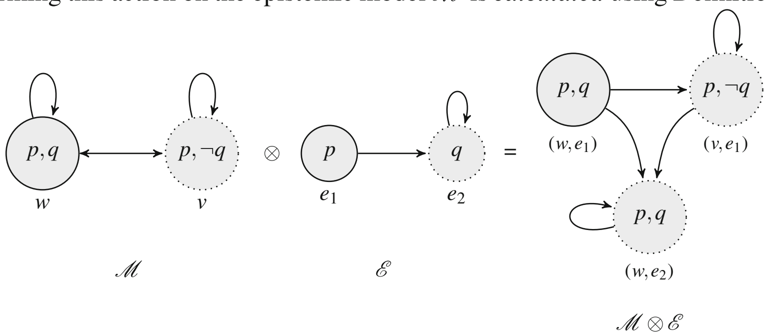
Definition 9.2 Suppose that $\mathscr{L}_{EL}$ is the epistemic modal language. An *event model* is a tuple $\langle S, \longrightarrow, \mathsf{pre}\rangle$, where S is a nonempty set of *primitive events*, for each $i \in \mathbb{A}$, $\longrightarrow \subseteq S \times S$ and $\mathsf{pre} : S \to \mathscr{L}_{EL}$ is the *precondition function*.

The only difference with a relational model is that the precondition function assigns a single formula to each primitive event. The intuition is that $\mathsf{pre}(e)$ describes what must be true in order for the event e to happen. Given two primitive events e and f, $e \longrightarrow f$ means "if event e takes place, then i *thinks* it is event f". The information provided by an event model can be incorporated into a relational structure using the following operation [3]:

Definition 9.3 (*Product Update*) The *product update* $\mathscr{M} \otimes \mathscr{E}$ of a relational model $\mathscr{M} = \langle W, R, V \rangle$ and event model $\mathscr{E} = \langle S, \longrightarrow, \mathsf{pre} \rangle$ is the relational model $\langle W', R', V' \rangle$ with:

1. $W' = \{(w, e) \mid w \in W, e \in S \text{ and } \mathscr{M}, w \models \mathsf{pre}(e)\}$;
2. $(w, e)R'(w', e')$ iff wRw' in $\mathscr{M}$ and $e \longrightarrow e'$ in $\mathscr{E}$; and
3. for all $p \in \mathsf{At}$, $(s, e) \in V'(p)$ iff $s \in V(p)$. □

The following abstract example illustrates this operation. Suppose that, initially, the agent knows that p is the case, but thinks that both q and $\neg q$ are (epistemically) possible. This epistemic model $\mathscr{M}$ is represented on the left in the picture below, with an edge from state w to state v provided the agent cannot distinguish between w and v. Suppose that p describes the actual event, but the agent (mistakenly) thinks she observes q. This can be described by the following event model $\mathscr{E}$. The result of performing this action on the epistemic model $\mathscr{M}$ is *calculated* using Definition 9.3:



The first thing to notice is that the model $\mathscr{M} \otimes \mathscr{E}$ is not an epistemic model since the relation is not an equivalence relation. But this makes sense since the agent was misinformed or uncertain about precisely what she observed.

9.4.1 Comparing ETL and DEL

Both ETL and DEL are logical frameworks that are intended to describe the flow of information in a social interactive situation. Summarizing the results found in [6, Sect. 3], this section shows how these two "competing" logical frameworks can be rigorously compared. We will

(i) illustrate how DEL product update (Definition 9.3) may be used to generate interesting ETL frames, and
(ii) describe the observational powers of the agents presupposed in the DEL setting.

The key observation is that by repeatedly updating an epistemic model with event models, the machinery of DEL (i.e., Definition 9.3) in effect creates ETL models. Note that an ETL model contains not only a description of how the agents' information changes over time, but also "protocol information" describing *when* each event *can* be performed. Thus, in rigorously comparing DEL with ETL models, the protocol information must be made explicit.

Let $\mathbb{E} = \{(\mathscr{E}, e) \mid \mathscr{E}$ an event model and $e \in \mathscr{E}\}$ be the class of all pointed event models. A *DEL protocol* (called a *uniform protocol* in [6]) is a set $\mathsf{P} \subseteq \mathbb{E}^*$ closed under the (non-empty) initial segment relations.[4] Given a DEL protocol P, let σ denote an element of P, i.e., σ is a sequence of pointed event models. We write σ_n for the initial segment of σ of length $n \leq \mathsf{len}(\sigma)$ and write $\sigma_{(n)}$ for the nth component of σ. For example, if $\sigma = (\mathscr{E}_1, e_1)(\mathscr{E}_2, e_2)(\mathscr{E}_3, e_3)\cdots(\mathscr{E}_n, e_n)$, then $\sigma_3 = (\mathscr{E}_1, e_1)(\mathscr{E}_2, e_2)(\mathscr{E}_3, e_3)$ and $\sigma_{(3)} = (\mathscr{E}_3, e_3)$. Given a sequence $\sigma \in \mathbb{E}^*$, we abuse notation and write $\mathsf{pre}(\sigma_{(n)})$ for $\mathsf{pre}(e_n)$ where $\sigma_{(n)} = (\mathscr{E}_n, e_n)$. Furthermore, we write $\sigma_{(n)} \longrightarrow_i \sigma'_{(n)}$ provided $\sigma_{(n)} = (\mathscr{E}, e)$ and $\sigma'_{(n)} = (\mathscr{E}, e')$ and $e \longrightarrow_i e'$ is in $\mathscr{E}$. Finally, let $Ptcl(\mathbb{E})$ be the class of all DEL protocols, i.e., $Ptcl(\mathbb{E}) = \{\mathsf{P} \mid \mathsf{P} \subseteq \mathbb{E}^*$ is closed under initial segments$\}$.

The main idea is to start from an initial (pointed) epistemic model and construct an ETL model by repeatedly applying product updates.

Definition 9.4 Given a pointed epistemic model $\mathscr{M}, w$ and a finite sequence of pointed event models σ, we define the *σ-generated epistemic model*, $(\mathscr{M}, w)^\sigma$ as $(\mathscr{M}, w) \otimes \sigma_{(1)} \otimes \sigma_{(2)} \otimes \cdots \otimes \sigma_{(\mathsf{len}(\sigma))}$. We will write $\mathscr{M}^\sigma$ for $(\mathscr{M}, w)^\sigma$ when the state w is clear from context.

Definition 9.5 Let $\mathscr{M}, w$ be a pointed epistemic model, and P a DEL protocol. The ETL model generated by $\mathscr{M}$ and P, $\mathsf{Forest}(\mathscr{M}, \mathsf{P})$, represents all possible evolutions of the system obtained by updating $\mathscr{M}$ with sequences from P. More precisely, $\mathsf{Forest}(\mathscr{M}, \mathsf{P}) = \langle \Sigma, \mathsf{H}, \{\sim_i\}_{i \in \mathbb{A}}, V \rangle$, where $\langle \mathsf{H}, \{\sim_i\}_{i \in \mathbb{A}}, V \rangle$ is the union of all models of the form $\mathscr{M}^\sigma$ with $\sigma \in \mathsf{P}$.

Since any DEL protocol P is closed under prefixes, for any epistemic model $\mathscr{M}$, $\mathsf{Forest}(\mathscr{M}, \mathsf{P})$ is indeed an ETL model. Now, given a class of DEL protocols $\mathbf{X}$, let

$$\mathbb{F}(\mathbf{X}) = \{\mathsf{Forest}(\mathscr{M}, \mathsf{P}) \mid \mathscr{M} \text{ an epistemic model and } \mathsf{P} \in \mathbf{X}\}$$

If $\mathbf{X} = \{\mathsf{P}\}$ then we write $\mathbb{F}(\mathsf{P})$ instead of $\mathbb{F}(\{\mathsf{P}\})$.

Note that not all ETL models can be generated by a DEL protocol. Indeed, such generated ETL models satisfy synchronicity, perfect recall and uniform no miracles (see Sect. 9.3.3 for definitions). The main result (Theorem 9.2 in this section is a characterization of the ETL models that are generated by some (uniform) DEL protocol. This is an improvement of an earlier characterization result from [12] and provides a precise comparison between the DEL and ETL frameworks.

Suppose that $\mathscr{H}$ is an ETL frame, which satisfies synchronicity, perfect recall and uniform no miracles. We can easily read off an epistemic *frame* with a set of states W and relations R_i for each agent $i \in \mathbb{A}$ on W, to serve as the initial model, where the histories of length 1 are the states and the uncertainty relations are simply

[4] The *preconditions of DEL* also encode protocol information of a 'local' character, and hence they can do some of the work of global protocols, as has been pointed out by van Benthem [12].

copied. Furthermore, we can define a "DEL-like" protocol $\mathsf{P}_{\mathscr{H}}$ with the construction given below in the proof of Theorem 9.2, consisting of sequences of event models where the precondition function assigns to the primitive events *sets of finite histories*. Intuitively, if e is a primitive event, i.e., a state in an event model, then $\mathsf{pre}(e)$ is the set of histories where e can "be performed". Thus, we have a comparison of the two frameworks at the level of frames provided we work with a modified definition of an event model. However, the representation theorem is stated in terms of models, so we need additional properties. In particular, at each level of the ETL model we will need to specify a *formula* of $\mathscr{L}_{EL}$ as a pre-condition for each primitive event e (cf. Definition 9.2). As usual, this requires that the set of histories preceding an event e be *bisimulation-closed* (see [20] for a definition of bisimulations and [6] for the precise definition needed here). However, as is well-known, bisimulation-invariance alone is typically not enough to guarantee the existence of such a formula. More specifically, there are examples of *infinite* sets that are bisimulation closed but not definable by any formula of $\mathscr{L}_{EL}$ (However, it will be definable by a formula of epistemic logic with *infinitary* conjunctions—see [20] for a discussion). Thus, if the set of histories at some level in which an event e can be executed is infinite, there may not be a formula of $\mathscr{L}_{EL}$ that defines this set to be used as a pre-condition for e. Such a formula will exist under an appropriate *finiteness assumption*: at each level there are only finitely many histories in which e can be executed,[5] i.e., for each n, the set $\{h \mid he \in \mathsf{H} \text{ and } \mathsf{len}(h) = n\}$ is finite.

One final assumption is needed since we are assuming that product update does not change the ground facts. An ETL model $\mathscr{H}$ satisfies *propositional stability* provided for all histories h in $\mathscr{H}$, events e with he in $\mathscr{H}$ and all propositional variables P, if P is true at h then P is true at he. We remark that this property is not crucial for the results in this section and can be dropped provided we allow product update to change the ground facts (cf. [5]).

Theorem 9.2 (Representation Theorem) *Let $\mathbf{X}_{DEL}$ be the class of uniform DEL protocols. An ETL model $\mathscr{H}$ is in $\mathbb{F}(\mathbf{X}_{DEL})$ if and only if $\mathscr{H}$ satisfies propositional stability, synchronicity, perfect recall, uniform no miracles, and bisimulation invariance.*

Consult [6, Theorem 1] for the proof. Note that the finiteness assumption can be dropped at the expense of allowing preconditions to come from a more expressive language (specifically, infinitary epistemic logic). Alternatively, as remarked above, we can define the preconditions to be *sets* of histories, instead of formulas of some logical language.

In [25] Dégremont, Löwe, and Witzel provide an alternative merging approach by mapping an epistemic model and a protocol of pointed models to an epistemic temporal structure. The resulting epistemic temporal structure need not be synchronous, so the authors argue that synchronicity is not an inherent property of DEL, but

[5] Note that this property may be violated even in an ETL model generated from only finitely many events.

rather of the translation used in [6]. They provide a different translation that produces asynchronous ETL models and discuss a minimal temporal extension of DEL that removes the ambiguities between the possible translations. In this context, they discuss the question of which epistemic-temporal properties are intrinsic to DEL and which ones are properties of the translation.

9.5 The Dynamics of Knowledge and Abilities in Multi-Player Games

In this section we discuss a particular aspect of the dynamics of knowledge in multi-agent systems, effected in the course of playing (abstract) multi-player games. Games offer a variety on perspectives on this topic, and we refer the reader to the chapter [23] for some of them. A most important specific issue arising in multi-player games on which we focus here is the *interaction between knowledge and abilities* of players to achieve their objectives in the play of the game. This interaction has two equally important directions. On the one hand, the abilities of the players to guarantee achievement of their objectives (i.e., to have a winning strategy) crucially depend on their information about the game as such, and about the particular play of the game. On the other hand, the players' knowledge changes dynamically in the course of playing the game. That dynamic interaction crucially affect the players' abilities in the play the game.

Building on [37], here we will outline a logical framework, capturing the dynamics of the interplay between knowledge and abilities of players in multi-player games. In order to avoid having to deal with fundamental issues of the concept of knowledge, we hereafter prefer to use the more neutral, and at the same time more general, notion of *information*, which refers not only to the knowledge or uncertainties, but also to beliefs and confusions of the agents/players.

9.5.1 A Priori vs Empirical Information of Players

First, some brief terminological remarks. Traditionally, in Game Theory the notion of 'incomplete information' refers to the knowledge or uncertainties of the player about the *structure and rules of the game*, while the notion of 'imperfect' information' refers to the knowledge or uncertainties of the player about *the course of the play* of the game, e.g. about the state in which the game currently is, or the history of the play, or the moves/actions taken by other players. Instead, we introduce the notions of 'a priori information' and 'empirical information'. Intuitively, a player's *a priori information* is the information (incl. knowledge, beliefs and uncertainties) that the player has about the game as such, *prior* to the actual play of the game about its *rules*, *protocol*, or *structure*. On the other hand,*empirical information* refers to the information that

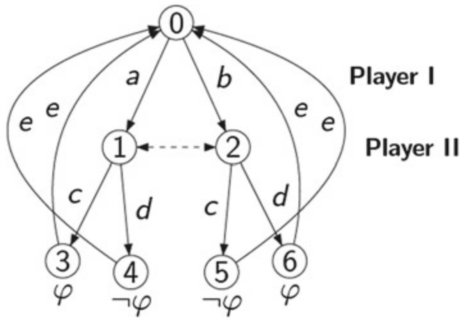


Fig. 9.2 Learning from experience

a player builds by way of observations, recollections, communication and reasoning made during the course of play. The a priori information plays its role only at the beginning of the game. Ultimately, it is the empirical information that determines the players' abilities in the game.

9.5.2 Some Examples

We now present several simple examples in order to illustrate the concepts of a priori information and the empirical information of a player. In all examples that follow we consider simple turn-based or concurrent games, played by two players, **I** and **II**, consisting in them making series of moves. States in the game are labelled with numbers, with 0 indicating the start state. Outcomes of the games are only qualitative, expressed in terms of the truth of certain propositions.

Example 9.1 (*Learning from experience*) In the game on Fig. 9.2 Player **I** moves first from state 0, then **II** moves, and then the game restarts from state 0. Player **II** has incomplete *a priori* information about the game: he is not able to distinguish a priori between states 1 and 2. For instance, that means that he cannot observe or recognize the actions a or b of Player **I**. For convenience we will usually refer to Player **I** as female and to Player **II** as male.

So, does Player **II** have the ability to eventually guarantee his desired outcome φ? If his uncertainty persists throughout the game, then clearly not. However, if Player **II** can observe the action of Player I and use some memory, he can learn from his experience: after choosing an action at random at the first round of the game if it does not lead to the desired outcome, then he can revise his strategy to achieve φ in the next round, by playing the other action if Player **I** repeats the same action and vice versa. Thus, the experience during the play can enrich the player's information and thus enhance his abilities.

Example 9.2 (*Getting confused*) In the game on Fig. 9.3 Player **II** has perfect a priori information about the game.

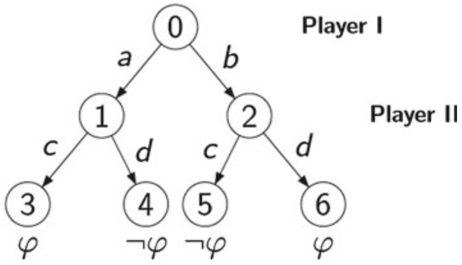


Fig. 9.3 Getting confused

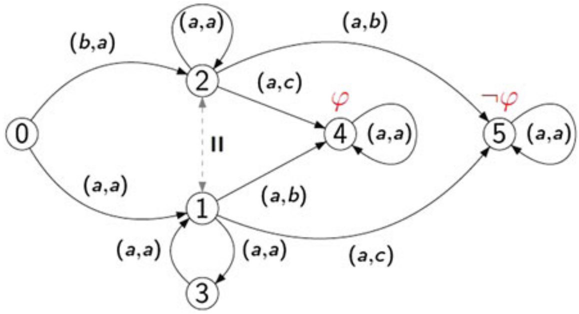


Fig. 9.4 Learning by experiments

So, does **II** have the a priori ability to guarantee the desired outcome φ? Yes, player **II** has a simple strategy for that. But, suppose that in the course of the actual play of the game, player **II** turns out unable to *observationally* distinguish states 1 and 2 after player **I** made her move, e.g., by failing to observe player **I**'s action, or due to malfunctioning sensors. If that happens, then Player **II** is no longer able to guarantee outcome φ. Thus, the experience in the play of the game can be negative, too and can lead to loss of information and abilities.

Example 9.3 (*Learning by experiments*) In the concurrent game on Fig. 9.4 essentially Player **I** determines the move from state 0, and thereafter **II** is in control. Player **II**'s objective is to reach a φ-state but she cannot distinguish a priori states 2 and 3 and cannot observe the action of Player **I** at state 0. However, Player **II** has all the information about the game which is provided on Fig. 9.4, except that he cannot see the labels 1 and 2.

So, does **II** can guarantee reaching the outcome φ? A priori, not. However, by performing a suitable experiment at the information set of states $\{2, 3\}$ player **II** can generate sufficient *empirical information* to enable himself distinguish between these states, by following the strategy: play action a and observe the result. If that is state 2, then play c, and if it is state 3 (which player **II** can distinguish from state 2), then play a again, and then b.

Remark 9.2 Note that the reasoning described in the last example can be naturally regarded as 'a priori reasoning', as it may take place before any actual play of the game. However, one should distinguish the results of 'a priori reasoning' in this intuitive sense from the notion of 'a priori information' we use here. Since the results of the 'a priori reasoning' in this example applies to possible *plays* of the game, this information is 'empirical information' in the sense in which we use this term here.

In summary: players' information changes dynamically during the play of the game. The main problem arising here is: how to formally compute that dynamics?

9.5.3 Formalizing the a Priori and Empirical Information

Here we propose a formal modeling framework incorporating the a priori information and computing the empirical information of the players.

The models upon which the framework is built are variation of the concurrent game structures with incomplete information [2, 36, 46, 60]. The key extension to these structures consists in two 'information relations' per player. The first, called 'a priori information relation' relates one state to another if the player considers the second state a possible *structural alternative* for the first, i.e. a priori alternative in the game structure. This relation can also be used to represent structural uncertainties and beliefs. The second, called 'empirical information relation', relates a *run*, i.e., initial segment of a play of the game, to (possibly) another one which the player considers a possible 'observational' or 'empirical alternative' to the first one. This relation can be used to represent empirical uncertainties and beliefs arising in the course of the play.

9.5.3.1 Concurrent Game Structures

Definition 9.6 (*Concurrent game structure*) A *concurrent game structure* (CGS) is a tuple $\langle \mathbb{A}, Q, \mathsf{Act}, d, \mathsf{out} \rangle$, where:

- $\mathbb{A} = \{\mathbf{1}, \mathbf{2}, \ldots, \mathbf{k}\}$ is a finite set of players.
- Q is a non-empty set of *states*.
- $d : Q \times \mathbb{A} \longrightarrow \mathbf{2}^{\mathsf{Act}}$ is a function that for every state q and a player i assigns the subset of actions available to player i at state q.
  The set of actions available to i at q will be denoted $\mathsf{Act}_i(q)$.
  A *joint action* at a given state q, denoted by σ_q (or simply by σ when q is fixed by the context), is a tuple $(\alpha_1, \alpha_2, \ldots, \alpha_k)$, where $\alpha_i \in \mathsf{Act}_i(q)$ for every $i \leq k$, consisting of a collection of actions, one for each player, that may be performed at state q. Given a joint action $\sigma = (\alpha_1, \alpha_2, \ldots, \alpha_k)$, we write σ^i to indicate α_i. We write $\mathsf{Act}(Q)$ for the set of all joint actions from states in Q.
- $\mathsf{out} : Q \times \mathsf{Act}(Q) \to Q$ is the *transition function*, that maps a state q and a joint action at q to a unique *successor state* in Q. The set of all successor states of q

will be denoted by $succ(q)$. Thus, $q' \in succ(q)$ if there is a move vector σ such that $\mathsf{out}\,(q, \sigma) = q'$.

One can think of a CGS as capturing the structure and the rules of a game.

Definition 9.7 (*Games, runs and plays*) A *game* is a pair $\langle \mathscr{S}, q \rangle$ consisting of a CGS $\mathscr{S}$ with a set of states Q and an *initial state* $q \in Q$. Given a game $\mathscr{G} = \langle \mathscr{S}, q \rangle$, a *play* of $\mathscr{G}$ is an infinite sequence $\lambda = q_0, \sigma_0; q_1, \sigma_1; q_2, \sigma_2 \ldots$ of alternating states and joint actions applied at them, such that $q_0 = q$ and $q_{i+1} = \mathsf{out}\,(q_i, \sigma_i)$ for every $i \in \mathbb{N}$. A *(finite) run* in $\mathscr{G}$ is a finite initial segment of a play ending with a state: $q_0, \sigma_0; q_1, \sigma_1; \ldots; q_n$. One-state runs will be identified with the respective states.

A *q-play (resp. q-run)* is a play (resp. run) where $q_0 = q$. We denote the set of q-plays by $Play(q)$ and the set of all q-runs by $Run(q)$. We also denote the set of all plays in $\mathscr{S}$ by $Play(\mathscr{S})$ and the set of all runs in $\mathscr{S}$ by $Run(\mathscr{S})$.

Example 9.4 The game on Fig. 9.4 defines a CIGS $\mathscr{S} = \langle \mathbb{A}\,, Q, \mathsf{Act}\,, d, \mathsf{out}\,\rangle$, where:

- $\mathbb{A}\, = \{\mathbf{I}, \mathbf{II}\}$, $Q = \{0, \ldots, 7\}$, $\mathsf{Act}\, = \{a, b, c\}$.
- $d : Q \times \mathbb{A} \longrightarrow \mathbf{2}^{\mathsf{Act}}$ and $\mathsf{out}\, : Q \times \mathsf{Act}\,(Q) \to Q$ are defined as on the figure, e.g.:
  $d(0, \mathbf{I}) = \{a, b\}, d(0, \mathbf{II}) = \{a\}$; $\mathsf{out}\,(0; (a, a)) = 1, \mathsf{out}\,(0; (b, a)) = 2$, etc.

Here are some plays in the game $(\mathscr{S}, 0)$:

- $0, (a, a); 1, (a, a); 3, (a, a); 1, (a, a); 3, (a, a) \ldots$
- $0, (a, a); 1, (a, c); 5, (a, a); 5, (a, a); 5, (a, a) \ldots$
- $0, (a, a); 1, (a, a); 3, (a, a); 1, (a, b); 4, (a, a); 4, (a, a) \ldots$
- $0, (b, a); 2, (a, a); 2, (a, a); 2, (a, c); 4, (a, a); 4, (a, a) \ldots$

9.5.3.2 Concurrent Informational Game Structures

Definition 9.8 (*Concurrent informational game structure*) A *concurrent informational game structure* (CIGS) is a tuple $\langle \mathscr{S}; \{\overset{a}{\rightsquigarrow}_{\mathbf{i}}\}_{\mathbf{i} \in \mathbb{A}}\,, \{\overset{e}{\rightsquigarrow}_{\mathbf{i}}\}_{\mathbf{i} \in \mathbb{A}}\,\rangle$, where $\mathscr{S} = \langle \mathbb{A}\,, Q, \mathsf{Act}\,, d, \mathsf{out}\,\rangle$ is a CGS and for every $\mathbf{i} \in \mathbb{A}\,$:

- $\overset{a}{\rightsquigarrow}_{\mathbf{i}} \subseteq Q \times Q$ is a *a priori information relation* for the player $\mathbf{i}$;
- $\overset{e}{\rightsquigarrow}_{\mathbf{i}} \subseteq Run(\mathscr{S}) \times Run(\mathscr{S})$ is an *empirical information relation* for the player $\mathbf{i}$, which coincides with $\overset{a}{\rightsquigarrow}_{\mathbf{i}}$ when restricted to one-state runs.

The intuition: $q_1 \overset{a}{\rightsquigarrow}_{\mathbf{i}} q_2$ holds if the player $\mathbf{i}$ considers the state q_2 as a possible *a priori* alternative of the state q_1 in the game structure $\mathscr{M}$. Likewise, $\rho_1 \overset{e}{\rightsquigarrow}_{\mathbf{i}} \rho_2$ if the player $\mathbf{i}$ considers the run ρ_2 a possible alternative of the run ρ_1 in $\mathscr{M}$ at its last state. Initially, the empirical information of the player about a play is simply her a priori information about the game. As the play progresses, the player may on one hand gain some additional information about the structure of the game, and on the other hand may acquire some uncertainties or wrong beliefs about the history and the current state of the play. Here are some particular cases:

- If $\overset{a}{\rightsquigarrow}_{\mathbf{i}}$ is the equality, then the player **i** has a complete (a priori) information about the game $\mathscr{M}$. Likewise, if $\overset{e}{\rightsquigarrow}_{\mathbf{i}}$ restricted to the runs of a given play in $\mathscr{M}$ is the equality (i.e. does not associate any run of the play with any different run), then the player **i** maintains a perfect empirical information throughout that play.
- If the player **i** has no wrong beliefs, but only uncertainties about the game, then $\overset{a}{\rightsquigarrow}_{\mathbf{i}}$ is an equivalence relation of *a priori indistinguishability*. Likewise, if the player **i** has no wrong beliefs, but only uncertainties about the states of a play, then $\overset{e}{\rightsquigarrow}_{\mathbf{i}}$ is an equivalence relation of *empirical indistinguishability*.
- Furthermore, if the player can keep a count of the number of moves made in the play, or has a 'clock' showing how many time units have passed since the beginning of the play (where every time unit corresponds to one transition from a state to a successor state), then $\overset{e}{\rightsquigarrow}_{\mathbf{i}}$ can only relate runs of the same length (recall the property of synchrony in temporal epistemic models). In general, however, it is conceivable that a player may 'forget' the length of the current run.
- In many cases it is reasonable to assume that the games considered are 'tree-like', where every state is associated with a unique run. Then the relations $\overset{e}{\rightsquigarrow}_{\mathbf{i}}$ can be regarded as relations on states, rather than runs.

A more refined approach towards the different types of information and abilities would be to distinguish purely *observational* information acquired in the course of a play by only observing the current states through which the play goes from the accumulative empirical information which also involves recollections and reasoning.

Example 9.5 The game on Fig. 9.4 defines a CIGS $\mathscr{T} = \langle \mathscr{S}; \{\overset{a}{\rightsquigarrow}_{\mathbf{i}}\}_{\mathbf{i}\in\mathbb{A}}, \{\overset{e}{\rightsquigarrow}_{\mathbf{i}}\}_{\mathbf{i}\in\mathbb{A}} \rangle$, where $\mathscr{S}$ is the CGS defined in the previous example, and the a priori information relations are defined as follows:

- $\overset{a}{\rightsquigarrow}_{\mathbf{I}}$ is the equality: $\{(q, q) \mid q \in Q\}$;
- $\overset{a}{\rightsquigarrow}_{\mathbf{II}} = \{(q, q) \mid q \in Q\} \cup \{(1, 2), (2, 1)\}$.

The empirical relations:

- $\overset{e}{\rightsquigarrow}_{\mathbf{I}}$ is the equality of runs.
- $\overset{e}{\rightsquigarrow}_{\mathbf{II}}$ extends $\overset{a}{\rightsquigarrow}_{\mathbf{II}}$ by relating every run to itself, but it also relates the run $0, (a, a); 1$ with $0, (b, a); 2$ and no other runs, because once the game is past any of these 2 runs, the a priori uncertainty of player **II** disappears (assuming that he has basic observational abilities and memory).

9.5.3.3 Computing the Empirical Information

Note that, while both sets of relations, $\overset{a}{\rightsquigarrow}$ and $\overset{e}{\rightsquigarrow}$ are part of the definition of a CIGS, only the former should be assumed to be given explicitly a priori, while the latter is to be computed in the course of the play. It is not possible to give a general rule of how the empirical relations of the players are computed, as that would depend on their observational abilities, memory, reasoning skills, etc. Computing the empirical information is one of the main problems in the development of this framework. Here we only outline a conceptual proposal for a mechanism computing the empirical information during the play of the game, as follows:

- Before the play begins, the empirical information of the players in the CIGS is their a priori information. It determines an “a priori” multi-agent epistemic model associated with the CIGS.
- Every transition in the CIGS generates an “information update model” à la DEL, which represents the epistemic updates for the payers generated by that transition.
- That update model is applied to the current epistemic model associated with the CIGS, to produce an updated epistemic model, which represents the empirical information relations between all runs of length being at most the length of the current history of the play.
- The players use the so obtained empirical information relations to determine their next actions, possibly following an ’empirical strategy’ based on the empirical information represented by these relations.
- The collective action determines the next transition, and the cycle repeats.

The procedure of computing the information update models is the engine of the entire mechanism and depends on the abilities of the players to observe, memorize and recall, communicate, reason, etc. Some simple cases of that procedure are being developed in [35].

9.5.3.4 Logical Framework for Computing Empirical Strategic Abilities

Using the a priori and empirical relations, one can refine the notions of strategies and strategic abilities of players, underlying the semantics of the Alternating-time temporal logic ATL, [2] in order to distinguish between ‘objective’ abilities (what the player can achieve *if* they had perfect information), ‘a priori’ abilities (what the player can achieve based on her a priori information about the game), and ‘empirical’ abilities (what the player can achieve given that the player can take advantage of, or suffer disadvantage from, experience of actual play). Furthermore, these can be used to provide a formal semantics of an enrichment of ATL with incomplete information, with separate operators for stating objective, a priori, and empirical abilities of players and coalitions. For further detail on all these we refer the reader to [37] and the work in preparation [35].

9.6 Putting the Temporal, Dynamic and Epistemic Frameworks Together

Besides [17] and [6], a few other publications have appeared recently that propose combining temporal, dynamic and epistemic frameworks. We briefly survey the more popular of them here.

In his Ph. D. thesis [57] and in the subsequent chapter [58] Sack combines temporal logic with public announcement logic (PAL) and dynamic epistemic logic (DEL). Adding next-time and previous-time operators to PAL allows formalizing the muddy children and the 'sum and product' puzzles. He also discusses relationships between the announcements and the new knowledge that agents acquire. Adding a full past-time operator to DEL also helps obtaining a complete axiomatization. In [59] Sack proposes a new version of temporal DEL (TDEL) with (mostly) unparametrized past operators in a language with DEL-action signatures. This TDEL does not involve protocols and the update modality semantics explicitly changes the epistemic temporal structure.

In [47] Hoshi and Yap consider a version of temporal DEL (TDEL) with a parametrized past operator. In order to axiomatize that extension, they develop transformation a given model into a certain normal form. The authors suggest further applications of such extensions of DEL to the theories of agency and learning.

In [55] Renne, Sack, and Yap introduce a new type of arrow in the DEL action models in order to enable reasoning about epistemic temporal dynamics in multi-agent systems that need not be synchronous. Their framework provides a new perspective on the work in [6], in particular, while in each of the two approaches the epistemic temporal models generated by standard update frames necessarily satisfy certain structural properties such as synchronicity, [55] discusses which these structural properties are due to the inherent structure of the update models themselves. In the extended version [56] they relate DETL and TDEL and provide a completeness theorem for DETL with respect to well-behaved epistemic temporal models.

In [27] van Ditmarsch, van der Hoek and Ruan discuss a relation between DEL with the usual semantics on relational models, and a temporal epistemic logic with semantics in interpreted systems à la [30]. In particular, from a given 'epistemic state', i.e. pointed epistemic model and a DEL formula they construct an interpreted system that satisfies the translation of the formula in the respective temporal epistemic logic.

9.7 Concluding Remarks

The dynamics of agents' knowledge is a conceptually rich, deep and multi-faceted topic. Here we have discussed only some aspects of that dynamics, mainly related to its temporality rather than its causes and effects. Furthermore, we have focused on the dynamics of *knowledge* rather than other informational attitudes, such as *beliefs*. Consult [22, 24] and the chapter [34] for a discussion of the temporal aspects of beliefs.

In summary, while a number of models and logics have been proposed in the past. This chapter has shown that there is much more in common between these different logical systems than once thought. Yet, a "Unified Theory" of the dynamics of knowledge over time is still to be developed, if ever. However, our overall goal was not to argue that any one framework is the "right model", or even that there is a single such "Unified Theory", but rather that there is a coherent collection of logical systems each focused on modeling the dynamics of knowledge from a different perspective. Actually, we are more inclined to believe that there is no unique model, not even unique 'right' methodology for modeling that dynamics, but that the pluralism of relevant approaches is its inherent valuable feature.

This pluralistic viewpoint of ours, together with the natural limitations of time and space in which this chapter had to be placed, are our excuses for leaving untouched a number of relevant studies and approaches, including: learning theory, interactive epistemology, situation calculus, etc. We do, however, refer the unslaked reader to Johan van Benthem's recent collection of though-provoking essays on related topics in [14, 16].

Acknowledgments We thank Johan van Benthem, Joshua Sack and Dominik Klein for useful comments and references. Valentin Goranko's work on this chapter was completed during his sabbatical visit to the Centre International de Mathématiques et Informatique de Toulouse. Eric Pacuit's work was supported by an NWO Vidi Grant 016.094.345.

References

1. Ågotnes T, van Ditmarsch H (2014) Knowledge games and coalitional abilities. In: Baltag A, Smets S (eds) Johan van Benthem on Logic and Information Dynamics, Springer, Dordrecht, pp 451–485 (Chapter 16 in this volume)
2. Alur R, Henzinger TA, Kuperman O (2002) Alternating-time temporal logic. J ACM 49(5):672–713
3. Baltag A, Moss L, Solecki S (1998) The logic of common knowledge, public announcements and private suspicions. In: Gilboa I (ed) Proceedings of the 7th conference on theoretical aspects of rationality and knowledge (TARK 98), pp 43–56
4. van Benthem J (1984) Tense logic and time. Notre Dame J. Formal Logic 25:1–16
5. van Benthem J, van Eijck J, Kooi B (2006) Logics of communication and change. Inf Comput 204(11):1620–1662
6. van Benthem J, Gerbrandy J, Hoshi T, Pacuit E (2009) Merging frameworks of interaction. J Philos Logic 38(5):491–526
7. van Benthem J (1993) The logic of time, 2nd edn. Kluwer Academic Publishers, Dordrecht
8. van Benthem J (1995) Temporal logic, Oxford University Press, Oxford, pp 241–350
9. van Benthem J (1996) Exploring logical dynamics. Center for the study of language and information, Stanford
10. van Benthem J (2001) Games in dynamic-epistemic logic. Bull Econ Res 53(4):219–248
11. van Benthem J (2001) Logics for information update. In: van Benthem J (ed) Proceedings of TARK 2001, Morgan Kaufmann, San Francisco, pp 51–67
12. van Benthem J (2002) 'One is a lonely number': on the logic of communication. In: Chatzidakis Z, Koepke P, Pohlers W (eds) Logic colloquium '02, pp 96–129. ASL and A. K. Peters, Available at http://staff.science.uva.nl/~johan/Muenster.pdf
13. van Benthem J (2004) What one may come to know. Analysis 64(2):95–105
14. van Benthem J (2011) Logical dynamics of information and interaction. Cambridge University Press, Cambridge
15. van Benthem J (2013) Reasoning about strategies. In: Coecke B, Ong L, Panagaden P (eds) Abramsky festrschrift, LNCS, vol 7860, pp 336–347
16. van Benthem J (2013) Logic in games. The MIT Press, Cambridge
17. van Benthem J, Pacuit E (2006) The tree of knowledge in action: towards a common perspective. In: Advances in modal logic, pp 87–106

18. van Benthem J, Pacuit E (2014) Connecting logics of choice and change. In: Nuel Belnap on indeterminism and free action, outstanding contributions to logic, vol 2, pp 291–314
19. Bezhanishvili N, van der Hoek W (2014) Structures for epistemic logic. In: Baltag A, Smets S (eds) Johan van Benthem on Logic and Information Dynamics, Springer, Dordrecht, pp 339–380 (Chapter 12 in this volume)
20. Blackburn P, de Rijke M, Yde V (2002) Modal logic, Campbridge University Press, Campbridge
21. Bonanno G (2004) Memory and perfect recall in extensive games. Games Econ Behav 47:237–256
22. Bonanno G (2012) Belief change in branching time: AGM-consistency and iterated revision. J Philos Logic 41(1):201–236
23. Bonanno G, Dégremont C (2014) Logic and game theory. In: Baltag A, Smets S (eds) Johan van Benthem on Logic and Information Dynamics, Springer, Dordrecht, pp 421–449 (Chapter 15 in this volume)
24. Dégremont C (2010) The temporal mind. Observations on the logic of belief change in interactive systems. Ph. D. thesis, Institute for Logic, Language and Computation (DS-2010-03)
25. Dégremont C, Löwe B, Witzel A (2011) The synchronicity of dynamic epistemic logic. In: Proceedings of TARK'2011, pp 145–152
26. van Ditmarsch H, van der Hoek W, Kooi B (2008) Dynamic epistemic logic. Springer, Dordecht
27. van Ditmarsch H, van der Hoek W, Ruan J (2012) Connecting dynamic epistemic and temporal epistemic logics. J IGPL (to appear)
28. van Eijck J (2014) Dynamic epistemic logics. In: Baltag A, Smets S (eds) Johan van Benthem on Logic and Information Dynamics, Springer, Dordrecht, pp 175–202 (Chapter 7 in this volume)
29. Engelfriet J (1996) Minimal temporal epistemic logic. Notre Dame J Formal Logic 37(2):233–259
30. Fagin R, Halpern JY, Moses Y, Vardi MY (1995) Reasoning about knowledge. The MIT Press, Boston
31. Fagin R, Halpern JY, Vardi MY (1992) What can machines know? On the properties of knowledge in distributed systems. J ACM 39(2):328–376
32. Finger M, Gabbay DM (1992) Adding a temporal dimension to a logic system. J Logic, Lang Inf 1:203–233
33. Gabbay DM, Kurucz A, Wolter F, Zakharyaschev M (2003) Many-dimensional modal logics: theory and applications, Vol 148. Elsevier, Amsterdam
34. Girard P, Rott H (2014) Belief revision and dynamic logic. In: Baltag A, Smets S (eds) Johan van Benthem on Logic and Information Dynamics, Springer, Dordrecht, pp 203–233 (Chapter 8 in this volume)
35. Goranko V (2012) Modelling the dynamics of knowledge and strategic abilities of players in multi-player games. In: Work in progress, Technical University of Denmark, Kongens Lyngby
36. Goranko V, Jamroga W (2004) Comparing semantics of logics for multi-agent systems. Synthese 139(2):241–280
37. Goranko V, Hawke P (2010) On the dynamics of information and abilities of players in multi-player games. In: Proceedings of LOFT'2010
38. Goranko V, Shkatov D (2009) Tableau-based decision procedure for full coalitional multiagent temporal-epistemic logic of linear time. In: Decker K, Sichman J, Sierra C, Castelfranchi C (eds) Proceedings of AAMAS'2009
39. Goranko V, Shkatov D (2009) Tableau-based decision procedure for the full coalitional multi-agent logic of branching time. In: Baldoni et al M (eds) MALLOW'2009
40. Halpern JY, Vardi MY (1988) Reasoning about knowledge and time in asynchronous systems. In: Simon J (ed) Proceedings of STOC'1988, ACM, pp 53–65
41. Halpern JY (1987) Using reasoning about knowledge to analyze distributed systems. Ann Rev Comput Sci 2:37–68
42. Halpern JY, Vardi MY (1989) The complexity of reasoning about knowledge and time i: lower bounds. J Comput Syst Sci 38(1):195–237
43. Halpern JY, Moses Yoram (1990) Knowledge and common knowledge in a distributed environment. J ACM 37(3):549–587

44. Halpern JY, van der Meyden R, Vardi M (2004) Complete axiomatizations for reasoning about knowledge and time. SIAM J Comput 33(2):674–703
45. Hintikka J (1962) Knowledge and belief: an introduciton to the logic of the two notions. Cornell University Press, Ithaca
46. van der Hoek W, Wooldridge M (2003) Cooperation, knowledge, and time: alternating-time temporal epistemic logic and its applications. Stud Logica 75(1):125–157
47. Hoshi T, Yap A (2009) Dynamic epistemic logic with branching temporal structures. Synthese 169(2):259–281
48. van der Meyden R, Wong K-S (2003) Complete axiomatizations for reasoning about knowledge and branching time. Stud Logica 75(1):93–123
49. van Otterloo S (2005) A strategic analaysis of multi-agent protocols. Ph. D. thesis, Institute for Logic, Language and Computation
50. Pacuit E (2007) Some comments on history based structures. J Appl Logic 5(4):613–624
51. Pacuit E, Simon S (2012) Reasoning with protocols under imperfect information. Rev Symbolic Logic 4:412–444
52. Parikh R, Ramanujam R (2003) A knowledge based semantics of messages. J Logic, Lang Inf 12:453–467
53. Parikh R, Ramanujam R (1985) Distributed processes and the logic of knowledge. In: Logic of programs. Lecture notes in computer science, Vol 193. Springer, New York, pp 256–268
54. Prior AN (1967) Past, present and future. Oxford University Press, Oxford
55. Renne B, Sack J, Yap A (2009) Dynamic epistemic temporal logic. In: Proceedings of LORI, pp 263–277
56. Renne B, Sack J, Yap A (2010) Dynamic epistemic temporal logic
57. Sack J (2007) Adding temporal logic to dynamic epistemic logic. PhD thesis, Indiana University, Bloomington
58. Sack J (2008) Temporal languages for epistemic programs. J Logic, Lang Inf 17(2):183–216
59. Sack J (2010) Logic for update products and steps into the past. Ann Pure Appl Logic 161(12):1431–1461
60. Schobbens P-Y (2004) Alternating-time logic with imperfect recall. Electron Notes Theoret Comput Sci, 85(2):82–93