Convergence, Continuity, Recurrence and Turing Completeness in Dynamic Epistemic Logic*

Dominik Klein[1] and Rasmus K. Rendsvig[2]

[1] Department of Philosophy, Bayreuth University, and
Department of Political Science, University of Bamberg
dominik.klein@uni-bayreuth.de
[2] Center for Information and Bubble Studies, University of Copenhagen
rasmus@hum.ku.dk

Abstract. The paper analyzes dynamic epistemic logic from a topological perspective. The main contribution consists of a framework in which dynamic epistemic logic satisfies the requirements for being a topological dynamical system thus interfacing discrete dynamic logics with continuous mappings of dynamical systems. The setting is based on a notion of logical convergence, demonstratively equivalent with convergence in Stone topology. Presented is a flexible, parametrized family of metrics inducing the Stone topology, used as an analytical aid. We show maps induced by action model transformations continuous with respect to the Stone topology and present results on the recurrent behavior of said maps. Among the recurrence results, we show maps induced by finite action models may have uncountably many recurrent points, even when initiated on a finite input model. Several recurrence results draws on the class of action models being Turing complete, for which the paper provides proof in the postcondition-free case. As upper bounds, is shown that either 1 atom, 3 agents and preconditions of modal depth 18, or 1 atom, 7 agents and preconditions of modal depth 3 suffices for Turing completeness.

Keywords: dynamic epistemic logic, limit behavior, convergence, recurrence, dynamical systems, metric spaces, general topology, modal logic, Turing machine, Turing completeness

1 Introduction

Dynamic epistemic logic is a framework for modeling information dynamics. In it, systematic change of Kripke models are punctiliously investigated through model transformers mapping Kripke models to Kripke models. The iterated application of such a map may constitute a model of information dynamics, or be may be analyzed purely for its mathematical properties [6, 8, 10, 11, 13, 16, 18, 42–45, 47].

Dynamical systems theory is a mathematical field studying the long-term behavior of spaces under the action of a continuous function. In case of discrete time, this amounts to investigating the space under the iterations of a continuous map. The field is rich in concepts, methodologies and results developed with the aim of understanding general dynamics.

The two fields find common ground in the iterated application of maps. With dynamic epistemic logic analyzing very specific map types, the hope is that general results from dynamical systems theory may shed light on properties of the former. There is, however, a chasm between the two: Dynamical systems theory revolves around spaces imbued with metrical or topological structure with respect to which maps are continuous. No such structure is found in dynamic epistemic logic.

This chasm has not gone unappreciated: In his 2011 *Logical Dynamics of Information and Interaction* [10], van Benthem writes

> **From discrete dynamic logics to continuous dynamical systems**
> "We conclude with what we see as a major challenge. Van Benthem [7, 8] pointed out how update evolution suggests a long-term perspective that is like the evolutionary dynamics found in dynamical systems. [...] Interfacing current dynamic and temporal logics with the continuous realm is a major issue, also for logic in general." [10, Sec. 4.8. Emph. is org. heading]

This paper takes on the challenge and attempts to bridge this chasm.

We proceed as follows. Section 2 presents what we consider natural spaces when working with modal logic, namely sets of pointed Kripke models *modulo* logical equivalence. These are referred to

* This is a revised and extended version of the paper [33]. Terminology and setting has been aligned with [34].

as *modal spaces*. A natural notion of "logical convergence" on modal spaces is provided. Section 3 seeks a topology on modal spaces for which topological convergence coincides with logical convergence. We consider a metric topology based on n-bisimulation and prove it insufficient, but show an adapted Stone topology satisfactory. Saddened by the loss of a useful aid, the metric inducing the n-bisimulation topology, a family of metrics is introduced that all induce the Stone topology, yet allow a variety of subtle modelling choices. Sets of pointed Kripke models are thus equipped with a structure of compact metric spaces. Section 4 considers maps on modal spaces based on multi-pointed action models using product update. Restrictions are imposed to ensure totality, and the resulting *clean maps* are shown continuous with respect to the Stone topology. With that, we present our main contribution: A modal space under the action of a clean map satisfies the standard requirements for being a topological dynamical system. Section 5 applies the now-suited terminology from dynamical systems theory, and present some initial results pertaining to the recurrent behavior of clean maps on modal spaces. Several recurrence results draws on the class of action models being Turing complete, for which the paper's Section 6 provides proof in the postcondition-free case. Section 7 concludes the paper by pointing out a variety of future research venues. Throughout, we situate our work in the literature.

Remark 1. To make explicit what may be apparent, note that the primary concern is the *semantics* of dynamic epistemic logic, i.e., its models and model transformation. Syntactical considerations are briefly touched upon in Section 7.

Remark 2. The paper is not self-contained. For notions from modal logic that remain undefined here, refer to e.g. [14,27]. For topological notions, refer to e.g. [39]. For more on dynamic and epistemic logic than the bare minimum of standard notions and notations rehearsed, see e.g. [2–5,10,20,22,30,40,41]. Finally, a background document containing generalizations and omitted proofs is our [34].

2 Modal Spaces and Logical Convergence

Let there be given a countable set Φ of **atoms** and a finite set I of **agents**. Where $p \in \Phi$ and $i \in I$, define the **language** $\mathcal{L}$ by

$$\varphi := \top \mid p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \Box_i \varphi.$$

Modal logics may be formulated in $\mathcal{L}$. By a **logic** Λ we refer only to extensions of the **minimal normal modal logic** K over the language $\mathcal{L}$.

We use relational semantics to evaluate formulas. A **Kripke model** for $\mathcal{L}$ is a tuple $M = (\llbracket M \rrbracket, R, \llbracket \cdot \rrbracket)$ where $\llbracket M \rrbracket$ is a countable, non-empty set of **states**, $R : I \longrightarrow \mathcal{P}(\llbracket M \rrbracket \times \llbracket M \rrbracket)$ assigns to each $i \in I$ an **accessibility relation** R_i, and $\llbracket \cdot \rrbracket : \Phi \longrightarrow \mathcal{P}(\llbracket M \rrbracket)$ is a **valuation**, assigning to each atom a set of states. With $s \in \llbracket M \rrbracket$, call $Ms = (\llbracket M \rrbracket, R, \llbracket \cdot \rrbracket, s)$ a **pointed Kripke model**. The used semantics are standard, including the modal clause:

$$Ms \vDash \Box_i \varphi \text{ iff for all } t : sR_i t \text{ implies } Mt \vDash \varphi.$$

Throughout, we work with pointed Kripke models, and when referring to a set of pointed Kripke models X alongside a modal language, we tacitly assume that all models in X share the signature of $\mathcal{L}$.

Working with modal logics, we find it natural to identify pointed Kripke models that are considered equivalent by the language used. The domains of interest are thus the following type of quotient spaces:

Definition 1. *The $\mathcal{L}$ **modal space** of a set of pointed Kripke models X is the set $\boldsymbol{X} = \{\boldsymbol{x} : x \in X\}$ for $\boldsymbol{x} = \{y \in X : y \vDash \varphi$ iff $x \vDash \varphi$ for all $\varphi \in \mathcal{L}\}$.*

When clear from context we will drop reference of $\mathcal{L}$, simply speaking of a modal space instead. Working with a modal space portrays that we only are interested in differences between pointed Kripke models insofar as these are modally expressible.[3]

In a modal space, how may we conceptualize that a sequence $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ converges to some point $\boldsymbol{x}$? Focusing on the concept from which we derive the notion of identity in modal spaces, namely

[3] This approach is slightly simpler, but essentially equivalent to that used in the shorter version of this paper, [33].

equivalence in $\mathcal{L}$, we find it natural to think of $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ as converging to $\boldsymbol{x}$ just in case $\boldsymbol{x_n}$ moves towards satisfying all the same formulas as $\boldsymbol{x}$ as n goes to infinity. We thus offer the following definition:

Definition 2. *A sequence of points $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ in a modal space $\boldsymbol{X}$ is said to* ***logically converge*** *to the point $\boldsymbol{x}$ in $\boldsymbol{X}$ iff for every $\varphi \in \mathcal{L}$ for which $x \vDash \varphi$, there is an $N \in \mathbb{N}$ such that $x_n \vDash \varphi$ for all $n \geq N$.*

To avoid re-proving useful results concerning this notion of convergence, we next turn to seeking a topology for which logical convergence coincides with topological convergence. Recall that for a topology $\mathcal{T}$ on a set X, a sequence of points $x_1, x_2, \ldots$ is said to **converge** to x in the **topological space** $(X, \mathcal{T})$ iff for every open set $U \in \mathcal{T}$ containing x, there is an $N \in \mathbb{N}$ such that $x_n \in U$ for all $n \geq N$.

3 Topologies on Modal Spaces

One way of obtaining a **topology** on a space is to define a **metric** for said space. Several metrics have been suggested for sets of pointed Kripke models [1,17]. These metrics are only defined for finite pointed Kripke models, but incorporating ideas from the metrics of [38] on **shift spaces** and [26] on sets of **first-order logical theories** allows us to simultaneously generalize and simplify the n-**Bisimulation-based Distance** of [17] to the degree of applicability:

Let $\boldsymbol{X}$ be a modal space for which modal equivalence and **bisimilarity** coincide[4] and let $\leftrightarroweq_n$ relate $x, y \in X$ iff x and y are n-**bisimilar**. Then proving

$$d_B(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} 0 & \text{if } x \leftrightarroweq_n y \text{ for all } n \\ \frac{1}{2^n} & \text{if } n \text{ is the least intenger such that } x \not\leftrightarroweq_n y \end{cases}$$

a metric on $\boldsymbol{X}$ is trivial. We refer to d_B as the n-**bisimulation metric**, and to the induced **metric topology** as the n-**bisimulation topology**, denoted $\mathcal{T}_B$. A basis of the topology $\mathcal{T}_B$ is given by the set of elements $B_{\boldsymbol{x}n} = \{\boldsymbol{y} \in \boldsymbol{X} \colon y \leftrightarroweq_n x\}$.

Considering the intimate link between modal logic and bisimulation, we consider both n-bisimulation metric and topology highly natural.[5] Alas, logical convergence does not:

Proposition 1. *Logical convergence in arbitrary modal space $\boldsymbol{X}$ does not imply convergence in the topological space $(\boldsymbol{X}, \mathcal{T}_B)$.*

Proof. Let $\boldsymbol{X}$ be an $\mathcal{L}$ modal space with $\mathcal{L}$ based on the atoms $\Phi = \{p_k \colon k \in \mathbb{N}\}$. Let $x \in X$ satisfy $\Box\bot$ and p_k for all $k \in \mathbb{N}$. Let $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ be a sequence in $\boldsymbol{X}$ such that for all $k \in \mathbb{N}$, x_k satisfies $\Box\bot$, p_m for all $m \leq k$, and $\neg p_l$ for all $l > k$. Then for all $\varphi \in \mathcal{L}$ for which $x \vDash \varphi$, there is an N such that $x_n \vDash \varphi$ for all $n \geq N$, hence the sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ converges to $\boldsymbol{x}$. There does not, however, exist any N' such that $\boldsymbol{x}_{n'} \in B_{\boldsymbol{x}0}$ for all $n' \geq N'$. Hence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ does not converge to $\boldsymbol{x}$ in $\mathcal{T}_B$. $\square$

Proposition 1 implies that the n-bisimulation topology may not straight-forwardly be used to establish negative results concerning logical convergence. That it may be used for positive cases is a corollary to Propositions 2 and 6 below. On the upside, logical convergence coincides with convergence in the n-bisimulation topology – i.e. Proposition 1 fails – when $\mathcal{L}$ has finite atoms. This is a corollary to Proposition 5.

An alternative to a metric-based approach to topologies is to construct the set of all open sets directly. Comparing the definition of logical convergence with that of convergence in topological spaces is highly suggestive: Replacing every occurrence of the formula φ with an open set U while replacing satisfaction ($\vDash$) with inclusion ($\in$) transforms the former definition into the latter. Hence the collection of sets $U_\varphi = \{\boldsymbol{x} \in \boldsymbol{X} : x \vDash \varphi\}$ for $\varphi \in \mathcal{L}$, seems a reasonable candidate for a topology. Alas, this collection is not closed under arbitrary unions, as all formulas are finite. Hence it is not a topology. It does however constitute the basis for a topology, in fact for the somewhat influential **Stone topology**, $\mathcal{T}_S$.

[4] That all models in X are **image-finite** is a sufficient condition, cf. the Hennessy-Milner Theorem. See e.g. [14] or [27].

[5] Space does not allow for a discussion of the remaining metrics of [1,17], but see [34].

The Stone topology is traditionally defined on the collection of complete theories for some propositional, first-order or modal logic, but is straightforwardly applicable to modal spaces. Moreover, it satisfies our *desideratum*:

Proposition 2. *For any modal space $\boldsymbol{X}$, a sequence $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ logically converges to the point $\boldsymbol{x}$ if, and only if, it converges to $\boldsymbol{x}$ in $(\boldsymbol{X}, \mathcal{T}_S)$.*

Proof. Assume $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ logically converges to $\boldsymbol{x}$ in $\boldsymbol{X}$ and that U containing $\boldsymbol{x}$ is open in $\mathcal{T}_S$. Then there is a basis element $U_\varphi \subseteq U$ with $\boldsymbol{x} \in U_\varphi$. So $x \vDash \varphi$. By assumption, there exists an N such that $x_n \vDash \varphi$ for all $n \geq N$. Hence $\boldsymbol{x_n} \in U_\varphi \subseteq U$ for all $n \geq N$.

Assume $\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots$ converges to $\boldsymbol{x}$ in $(\boldsymbol{X}, \mathcal{T}_S)$ and let $x \vDash \varphi$. Then $\boldsymbol{x} \in U_\varphi$, which is open. As the sequence converges, there exists an N such that $\boldsymbol{x_n} \in U_\varphi$ for all $n \geq N$. Hence $x_n \vDash \varphi$ for all $n \geq N$. □

If one accepts logical convergence as capturing the natural notion of convergence, then Proposition 2 provides an indication that the Stone topology is a natural candidate for analyzing logical dynamics. It is, moreover, the *unique* such candidate, as follows from:

Theorem 1. *Let $\boldsymbol{X}$ be a modal space and $\mathcal{T}$ a topology on $\boldsymbol{X}$. Then the following are equivalent:*

1. *A sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ of points from $\boldsymbol{X}$ converges to $\boldsymbol{x}$ in $(\boldsymbol{X}, \mathcal{T})$ if, and only if, $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots$ logically converges to $\boldsymbol{x}$ in $\boldsymbol{X}$.*
2. *$\mathcal{T}$ is the Stone-like topology $\mathcal{T}_S$ on $\boldsymbol{X}$.*

Proof. That 2. implies 1. is Proposition 2. That 1. implies 2. is shown in [34]. □

Apart from its attractive characteristic concerning convergence, by working on the basis of a logic, the Stone topology imposes a natural structure. As is evident from its basis, every subset of $\boldsymbol{X}$ characterizable by a single formula $\varphi \in \mathcal{L}$ is clopen. If the logic Λ is compact and $\boldsymbol{X}$ is 'Λ-complete', also the converse is true: every clopen set is of the form U_φ for some φ.[6] Here, the Λ-completeness of $\boldsymbol{X}$ is a requirement to the effect that $\boldsymbol{X}$ is sufficiently rich in model variety.[7] Formally:

Definition 3. *Let $\boldsymbol{X}$ be an $\mathcal{L}$ modal space and $\Lambda \subseteq \mathcal{L}$ sound with respect to X. Then $\boldsymbol{X}$ is Λ-**complete** if for each Λ-consistent set of formulas A, there is an $\boldsymbol{x} \in \boldsymbol{X}$ such that $x \vDash A$.*

When $\boldsymbol{X}$ is Λ-complete, one obtains a very natural modal space, containing for each maximal Λ-consistent set of formulas a unique point satisfying this set. It is thus homeomorphic to the space of all complete Λ-theories under the Stone topology of $\mathcal{L}$. Such spaces have been widely studied, see e.g. [26, 31, 48]. Calling such modal spaces Λ-complete reflects that the requirement ensures that the logic Λ is complete with respect to the set X, but that the obligation of sufficiency lies on the set X to be inclusive enough for Λ, not on Λ to be restrictive enough for X.[8]

In the following, we are interested mainly in spaces that are both compact and Λ-complete. In this case, a subset is open, but not closed, iff it is characterizable only by an infinitary disjunction of $\mathcal{L}$ formulas, and a subset if closed, but not open, iff it is characterizable only by an infinitary conjunction of $\mathcal{L}$ formulas. The Stone topology thus transparently reflects the properties of logic, language and topology. Moreover, it enjoys practical topological properties:

Proposition 3. *For any $\mathcal{L}$ modal space $\boldsymbol{X}$, $(\boldsymbol{X}, \mathcal{T}_S)$ is **Hausdorff** and **totally disconnected**. If Λ is (logically) **compact**[9] and $\boldsymbol{X}$ is Λ-complete, then $(\boldsymbol{X}, \mathcal{T}_S)$ is also (topologically) **compact**.*

Proof. These properties are well-known for the Stone topology applied to complete theories. For the topology applied to modal spaces, we defer to [34]. □

---

[6] See [34] for a proof.

[7] Λ-completeness (Def. 3) was called 'saturation' in [33]. In [34] 'Λ-saturation' refers to a weaker, more general notion than 'saturation'. Under the present assumptions, 'Λ-saturation' and the stronger notion 'Λ-completeness' are—as used in [33]—equivalent and equivalent to 'Λ-completeness' of Def. 3. We adopt 'Λ-completeness' to underline our stronger assumptions in comparison to the general setting of [33].

[8] Compare with the notion of *strong completeness*, cf. e.g. [14, Prop.4.12].

[9] A logic Λ is logically compact if any arbitrary set A of formulas is Λ-consistent iff every finite subset of A is Λ-consistent.

One may interject that, as having a metric may facilitate obtaining results and the n-bisimulation topology is induced by a metric, it may cause a loss of tools to move away from it. The Stone topology, however, is **metrizable**. A family of metrics inducing it, generalizing the Hamming distance to infinite strings by using weighted sums, was introduced in [34]. We here present a sub-family, suited for modal spaces:

Definition 4. *With X a set of pointed Kripke models for the language $\mathcal{L}$, let a **descriptor** be a set $D \subseteq \mathcal{L}$ such that for every $\varphi \in \mathcal{L}$ any valuation of D semantically entails either φ or $\neg\varphi$ over X.*[10]

For each descriptor, we can define a family of metrics on the modal space $\boldsymbol{X}$ as follows.

Definition 5. *Let $\boldsymbol{X}$ be the $\mathcal{L}$ modal space of X, D a descriptor, and $\varphi_1, \varphi_2 ...$ an enumeration of D.*
*i) For each $\varphi_k \in D$, define a **disagreement map** $d_k : \boldsymbol{X} \times \boldsymbol{X} \longrightarrow \{0, 1\}$ by*

$$d_k(\boldsymbol{x}, \boldsymbol{y}) = \begin{cases} 0 & \text{if } x \vDash \varphi_k \Leftrightarrow y \vDash \varphi_k \\ 1 & \text{else} \end{cases}$$

*ii) Call $w : D \longrightarrow \mathbb{R}_{>0}$ a **weight function** if it assigns a strictly positive **weight** to each $\varphi \in D$ such that $\sum_{\varphi \in D} w(\varphi) < \infty$.*
*iii) For weight function w, the **distance function** $d_w : \boldsymbol{X} \times \boldsymbol{X} \longrightarrow \mathbb{R}$ is defined by, for each $\boldsymbol{x}, \boldsymbol{y} \in \boldsymbol{X}$*

$$d_w(\boldsymbol{x}, \boldsymbol{y}) = \sum_{k=1}^{|D|} w(\varphi_k) d_k(\boldsymbol{x}, \boldsymbol{y}).$$

*iv) The set of distance functions d_w is denoted by $\mathcal{D}_{(\boldsymbol{X},D)}$, the set of D**-metrics** over $\boldsymbol{X}$. Finally, $\mathcal{D}_{\boldsymbol{X}} := \bigcup_{\{D \subseteq \mathcal{L}\colon D \text{ is a descriptor}\}} \mathcal{D}_{(\boldsymbol{X},D)}$ is the set of **descriptor-induced metrics** over $\boldsymbol{X}$.*

We refer to [34] for the proof establishing the following proposition:

Proposition 4. *Let $\boldsymbol{X}$ be any modal space and $\mathcal{D}_{\boldsymbol{X}}$ as in Definition 4. Then any $d_w \in \mathcal{D}_{\boldsymbol{X}}$ is a metric on $\boldsymbol{X}$ and the metric topology $\mathcal{T}_w$ induced by d_w on $\boldsymbol{X}$ is the Stone topology of $\mathcal{L}$.*

For a metric space $(\boldsymbol{X}, d)$, we will also write $\boldsymbol{X}_d$.

With variable parameters D and w, $\mathcal{D}_{\boldsymbol{X}}$ allows one to vary the choice of metric with the problem under consideration. E.g., if the n-bisimulation metric seems apt, one could choose that, with one restriction:

Proposition 5. *If $\boldsymbol{X}$ is an $\mathcal{L}$ modal space with $\mathcal{L}$ based on a finite atom set, then $\mathcal{D}_{\boldsymbol{X}}$ contains a topological equivalent to the n-bisimulation metric.*

Proof (sketch). As $\mathcal{L}$ is based on a finite set of atoms, for each $\boldsymbol{x} \in \boldsymbol{X}, n \in \mathbb{N}_0$, there exists a characteristic formula $\varphi_{x,n}$ such that $y \vDash \varphi_{x,n}$ iff $y \leftrightarroweq_n x$, cf. [27]. Let $D_n = \{\varphi_{x,n} \colon \boldsymbol{x} \in \boldsymbol{X}\}$ and $D = \cup_{n \in \mathbb{N}_0} D_n$. Then each D_n is finite and D satisfies Definition 4. Finally, let $w(\varphi) = \frac{1}{|D_n|} \cdot \frac{1}{2^{n+1}}$ for $\varphi \in D_n$. Then $d_w \in \mathcal{D}_{\boldsymbol{X}}$ and is equivalent to the n-bisimulation metric d_b. □

As corollary to Proposition 5, it follows that, for finite atom languages, the n-bisimulation topology is the Stone topology. This is not true in general, as witnessed by Proposition 1 and the following:

Proposition 6. *If $\boldsymbol{X}$ is a modal space with $\mathcal{L}$ based on a countably infinite atom set, then the n-bisimulation metric topology on $\boldsymbol{X}$ is strictly finer than the Stone topology on $\boldsymbol{X}$.*

Proof (sketch). We refer to [34] for details, but for $\mathcal{T}_B \not\subseteq \mathcal{T}_S$, note that the set $B_{\boldsymbol{x}0}$ used in the proof of Prop. 1, is open in $\mathcal{T}_B$, but not in $\mathcal{T}_S$. □

With this comparison, we end our exposition of topologies on modal spaces.

[10] The requirements on $D \subseteq \mathcal{L}$ make it what in [34] is called an '$\mathcal{L}$-representative descriptor', a special case of a more general class.

4 Clean Maps on Modal Spaces

We focus on a class of maps induced by action models applied using product update. Action models are a popular and widely applicable class of model transformers, generalizing important constructions such as public announcements. An especially general version of action models is *multi-pointed* action models with *postconditions*. Postconditions allow for ontic change, i.e. action states in an action model changing the valuation of atoms [12, 19], thereby also allowing the representation of information dynamics concerning situations that are not factually static. Permitting multiple points allows the actual action states executed to depend on the pointed Kripke model to be transformed, thus generalizing single-pointed action models.[11]

A **multi-pointed action model** is a tuple $\Sigma\Gamma = (\llbracket\Sigma\rrbracket, \mathsf{R}, pre, post, \Gamma)$ where $\llbracket\Sigma\rrbracket$ is a countable, non-empty set of **actions**. The map $\mathsf{R} : I \to \mathcal{P}(\llbracket\Sigma\rrbracket \times \llbracket\Sigma\rrbracket)$ assigns an **accessibility relation** R_i on $\llbracket\Sigma\rrbracket$ to each agent $i \in I$. The map $pre : \llbracket\Sigma\rrbracket \to \mathcal{L}$ assigns to each action a **precondition**, and the map $post : \llbracket\Sigma\rrbracket \to \mathcal{L}$ assigns to each action a **postcondition**,[12] which must be $\top$ or a conjunctive clause[13] over Φ. Finally, $\emptyset \neq \Gamma \subseteq \llbracket\Sigma\rrbracket$ is the set of designated actions.

To obtain well-behaved total maps on a modal spaces, we must invoke a set of mild, but non-standard, requirements: Let X be a set of pointed Kripke models. Call $\Sigma\Gamma$ **precondition finite** if the set $\{pre(\sigma) \in \mathcal{L} \colon \sigma \in \llbracket\Sigma\rrbracket\}$ is finite up to logical equivalence. This is needed for our proof of continuity. Call $\Sigma\Gamma$ **exhaustive over** X if for all $x \in X$, there is a $\sigma \in \Gamma$ such that $x \vDash pre(\sigma)$. This conditions ensures that the action model $\Sigma\Gamma$ is universally applicable on X. Finally, call $\Sigma\Gamma$ **deterministic over** X if $X \vDash pre(\sigma) \wedge pre(\sigma') \to \bot$ for each $\sigma \neq \sigma' \in \Gamma$. Together with exhaustivity, this condition ensures that the product of $\Sigma\Gamma$ and any $Ms \in X$ is a (single-)pointed Kripke model, i.e., that the actual state after the updates is well-defined and unique.

Let $\Sigma\Gamma$ be exhaustive and deterministic over X and let $Ms \in X$. Then the **product update** of Ms with $\Sigma\Gamma$, denoted $Ms \otimes \Sigma\Gamma$, is the pointed Kripke model $(\llbracket M\Sigma\rrbracket, R', \llbracket\cdot\rrbracket', s')$ with

$$
\begin{aligned}
\llbracket M\Sigma\rrbracket &= \{(s,\sigma) \in \llbracket M\rrbracket \times \llbracket\Sigma\rrbracket : (M,s) \vDash pre(\sigma)\} \\
R' &= \{((s,\sigma),(t,\tau)) : (s,t) \in R_i \text{ and } (\sigma,\tau) \in \mathsf{R}_i\}\,, \text{ for all } i \in N \\
\llbracket p\rrbracket' &= \{(s,\sigma) \colon s \in \llbracket p\rrbracket, post(\sigma) \nvDash \neg p\} \cup \{(s,\sigma) \colon post(\sigma) \vDash p\}\,, \text{ for all } p \in \Phi \\
s' &= (s,\sigma) : \sigma \in \Gamma \text{ and } Ms \vDash pre(\sigma)
\end{aligned}
$$

Call $\Sigma\Gamma$ **closing over** X if for all $x \in X$, $x \otimes \Sigma\Gamma \in X$. With exhaustivity and determinicaty, this ensures that $\Sigma\Gamma$ and $\otimes$ induce well-defined total map on X.

The class of maps of interest in the present is then the following:

Definition 6. *Let $\boldsymbol{X}$ be a modal space. A map $\boldsymbol{f} : \boldsymbol{X} \to \boldsymbol{X}$ is called* ***clean*** *if there exists a precondition finite, multi-pointed action model $\Sigma\Gamma$ closing, deterministic and exhaustive over X such that $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{y}$ iff $x \otimes \Sigma\Gamma \in \boldsymbol{y}$ for all $\boldsymbol{x} \in \boldsymbol{X}$.*

Clean maps are total by the assumptions of being closing and exhaustive. They are well-defined as $\boldsymbol{f}(\boldsymbol{x})$ is independent of the choice of representative for $\boldsymbol{x}$: If $x' \in \boldsymbol{x}$, then $x' \otimes \Sigma\Gamma$ and $x \otimes \Sigma\Gamma$ are modally equivalent and hence define the same point in $\mathbf{X}$. The latter follows as multi-pointed action models applied using product update preserve bisimulation [2], which implies modal equivalence. Clean maps moreover play nicely with the Stone topology:

Proposition 7. *Let $\boldsymbol{f}$ be a clean map on an $\mathcal{L}$ modal space $\boldsymbol{X}$. Then $\boldsymbol{f}$ is continuous with respect to the Stone topology of Λ.*

Proof (sketch). We defer to [34] for details, but offer a sketch: The map $\boldsymbol{f}$ is shown uniformly continuous using the ε-δ formulation of continuity. The proof relies on a lemma stating that for every $d_w \in \mathcal{D}_{\boldsymbol{X}}$ and every $\epsilon > 0$, there are formulas $\chi_1, \ldots, \chi_l \in \mathcal{L}$ such that every $x \in X$ satisfies some χ_i and whenever $y \vDash \chi_i$ and $z \vDash \chi_i$ for some $i \leq l$, then $d_w(y,z) < \epsilon$. The main part of the proof establishes the claim that there is a function $\boldsymbol{\delta} : \mathcal{L} \to (0,\infty)$ such that for any $\varphi \in \mathcal{L}$, if $f(x) \vDash \varphi$ and $d_{\boldsymbol{a}}(x,y) < \boldsymbol{\delta}(\varphi)$, then $f(y) \vDash \varphi$. Setting $\delta = \min\{\boldsymbol{\delta}(\chi_i) \colon i \leq l\}$ then yields a δ with the desired property. $\square$

[11] Multi-pointed action models are also referred to as *epistemic programs* in [2], and allow encodings akin to *knowledge-based programs* [22] of interpreted systems, cf. [44].

[12] The precondition of σ specify the conditions under which σ is executable, while its postcondition may dictate the posterior values of a finite, possibly empty, set of atoms.

[13] I.e. a conjunction of literals, where a literal is an atom or a negated atom.

With Proposition 7, we are positioned to state our main theorem:

Theorem 2. *Let $\boldsymbol{f}$ be a clean map on a Λ-complete $\mathcal{L}$ modal space $\boldsymbol{X}$ with Λ compact and let $d \in \mathcal{D}_{\boldsymbol{X}}$. Then $(\boldsymbol{X}_d, \boldsymbol{f})$ is a* ***topological dynamical system****.*

Proof. Propositions 2, 3, 4 and 7 jointly imply that $\boldsymbol{X}_d$ is a compact metric space on which $\boldsymbol{f}$ is continuous, thus satisfying the requirements of e.g. [21, 29, 49]. $\square$

With Theorem 2, we have, in what we consider a natural manner, situated dynamic epistemic logic in the mathematical discipline of dynamical systems. A core topic in this discipline is to understand the long-term, qualitative behavior of maps on spaces. Central to this endeavor is the concept of *recurrence*, i.e., understanding when a system returns to previous states as time goes to infinity.

5 Recurrence in the Limit Behavior of Clean Maps

We represent results concerning the limit behavior of clean maps on modal spaces. In establishing the required terminology, we follow [29]: Let f be a continuous map on a metric space X_d and $x \in X_d$. A point $y \in X$ is a **limit point**[14] for x under f if there is a strictly increasing sequence $n_1, n_2, \ldots$ such that the subsequence $f^{n_1}(x), f^{n_2}(x), \ldots$ of $(f^n(x))_{n\in\mathbb{N}_0}$ converges to y. The **limit set** of x under f is the set of all limit points for x, denoted $\omega_f(x)$. Notably, $\omega_f(x)$ is closed under f: For $y \in \omega_f(x)$ also $f(y) \in \omega_f(x)$. We immediately obtain that any modal system satisfying Theorem 2 has a nonempty limit set:

Proposition 8. *Let $(\boldsymbol{X}_d, \boldsymbol{f})$ be as in Theorem 2. For any point $\boldsymbol{x} \in \boldsymbol{X}$, the limit set of $\boldsymbol{x}$ under $\boldsymbol{f}$ is non-empty.*

Proof. Since $\boldsymbol{X}$ is is compact, every sequence in $\boldsymbol{X}$ has a convergent subsequence, cf. e.g. [39, Thm. 28.2]. $\square$

Proposition 8 does not inform us of the *structure* of said limit set. In the study of dynamical systems, such structure is often sought through classifying the possible repetitive behavior of a system, i.e., through the system's *recurrence* properties. For such studies, a point x is called (positively) **recurrent** if $x \in \omega_f(x)$, i.e., if it is a limit point of itself.

The simplest structural form of recurrence is *periodicity*: For a point $x \in X$, call the set $\mathcal{O}_f(x) = \{f^n(x) \colon n \in \mathbb{N}_0\}$ its **orbit**. The orbit $\mathcal{O}_f(x)$ is **periodic** if $f^{n+k}(x) = f^n(x)$ for some $n \geq 0, k > 0$; the least such k is the **period** of $\mathcal{O}_f(x)$. Periodicity is thus equivalent to $\mathcal{O}_f(x)$ being finite. Related is the notion of a **limit cycle**: a periodic orbit $\mathcal{O}_f(x)$ is a limit cycle if it is the limit set of some y not in the period, i.e., if $\mathcal{O}_f(x) = \omega_f(y)$ for some $y \notin \mathcal{O}_f(x)$.

It was conjectured by van Benthem that certain clean maps—those based on finite action models and without postconditions—would, whenever applied to a finite x, have a periodic orbit $\mathcal{O}_f(x)$. I.e., after finite iterations, the map would oscillate between a finite number of states. This was the content of van Benthem's "*Finite Evolution Conjecture*" [8]. The conjecture was refuted using a counterexample by Sadzik in his 2006 paper, [47].[15] The example provided by Sadzik (his Example 33) uses an action model with only Boolean preconditions. Interestingly, the orbit of the corresponding clean map terminates in a limit cycle of length 1, i.e., a unique limit point. This is a corollary to Proposition 9 below.

Before we can state the proposition, we need to introduce some terminology. Call a multi-pointed action model $\Sigma\Gamma$ **finite** if $[\![\Sigma]\!]$ is finite, **Boolean** if $pre(\sigma)$ is a Boolean formula for all $\sigma \in [\![\Sigma]\!]$, and **static** if $post(\sigma) = \top$ for all $\sigma \in [\![\Sigma]\!]$. We apply the same terms to a clean map $\boldsymbol{f}$ based on $\Sigma\Gamma$. In this terminology, Sadzik showed that for any finite, Boolean, and static clean map $\boldsymbol{f} \colon \boldsymbol{X} \to \boldsymbol{X}$, if the orbit $\mathcal{O}_{\boldsymbol{f}}(\boldsymbol{x})$ is periodic, then it has period 1.[16] In showing this, Sadzik shows that for every $\boldsymbol{x} \in \boldsymbol{X}$, for every $n \in \mathbb{N}$, there exists an $N \in \mathbb{N}$ such that $\boldsymbol{f}^N(\boldsymbol{x})$ is n-bisimilar to $\boldsymbol{f}^{N+1}(\boldsymbol{x})$—i.e., in present terms, that $(\boldsymbol{f}^n(\boldsymbol{x}))_{n\in\mathbb{N}_0}$ converges in the n-bisimulation topology. This insightful result immediates the following:

[14] Or ω-limit point. The ω is everywhere omitted as time here only moves forward.
[15] We paraphrase van Benthem and Sadzik using the terminology introduced.
[16] See [16] for an elegant and generalizing exposition.

Proposition 9. *Let $(\boldsymbol{X}_d, \boldsymbol{f})$ be as in Theorem 2 with $\boldsymbol{f}$ finite, Boolean, and static. For all $\boldsymbol{x} \in \boldsymbol{X}$, the limit set $\omega_{\boldsymbol{f}}(\boldsymbol{x})$ is a singleton.*[17]

Proof. By Prop. 8, the limit set $\omega_{\boldsymbol{f}}(\boldsymbol{x})$ of $\boldsymbol{x}$ under $\boldsymbol{f}$ is non-empty. Sadzik's result shows that it contains a single point, as convergence in the n-bisimulation topology implies convergence in the Stone topology. Hence $(\boldsymbol{f}^n(\boldsymbol{x}))_{n \in \mathbb{N}_0}$ converges to this point. As the limit set $\omega_{\boldsymbol{f}}(\boldsymbol{x})$ is closed under $\boldsymbol{f}$, its unique point is a fix-point. $\square$

Proposition 9 may be seen as a partial vindication of van Benthem's conjecture: Forgoing the requirement of reaching the limit set in finite time and the possibility of modal preconditions, the conjecture holds, even if the initial state has an infinite set of worlds $[\![x]\!]$. This simple recurrent behavior is, however, not the general case. More complex clean maps may exhibit **nontrivial recurrence**, i.e., produce non-periodic orbits with recurrent points:

Proposition 10. *There exist finite, static, but non-Boolean, clean maps that exhibit nontrivial recurrence.*

Proposition 11. *There exist finite, Boolean, but non-static, clean maps that exhibit nontrivial recurrence.*

We show these propositions below, building a clean map which, from a selected initial state, has uncountably many limit points, despite the orbit being only countable. Moreover, said orbit also contains infinitely many recurrent points. In fact, every element of the orbit is recurrent and hence a limit point. The same holds for all but one elements of the limit set. We rely on Theorem 3 in the proof. Theorem 3.1 is proved in Section 6, a proof of Theorem 3.2 in [15].

Theorem 3. *Any Turing machine can be emulated using a set X of finite pointed $S5$ Kripke models for finite atoms and a finite multi-pointed action model $\Sigma\Gamma$ deterministic over X. Moreover, $\Sigma\Gamma$ may be chosen*

1. *static, but non-Boolean, or*
2. *Boolean, but non-static.*

Proof (of Propostitions 10 and 11). For both propositions, we use a Turing machine *ad infinitum* iterating the successor function on the natural numbers. Numbers are represented in mirrored base-2, i.e., with the *leftmost* digit the *lowest*. Such a machine may be build with alphabet $\{\triangleright, 0, 1, \sqcup\}$, where the symbol $\triangleright$ is used to mark the starting cell and $\sqcup$ is the *blank* symbol. We omit the exact description of the machine here. Of importance is the content of the tape: Omitting blank ($\sqcup$) cells, natural numbers are represented as illustrated in Figure 1.

0 :	$\triangleright$ 0	2 :	$\triangleright$ 0 1	4 :	$\triangleright$ 0 0 1	6 :	$\triangleright$ 0 1 1	8 :	$\triangleright$ 0 0 0 1
1 :	$\triangleright$ 1	3 :	$\triangleright$ 1 1	5 :	$\triangleright$ 1 0 1	7 :	$\triangleright$ 1 1 1	9 :	$\triangleright$ 1 0 0 1

Fig. 1. Mirrored base-2 Turing tape representation of $0, .., 9 \in \mathbb{N}_0$, blank cells omitted. Notice that the mirrored notation causes perpetual change close to the start cell, $\triangleright$.

Initiated with its read/write head on the cell with the start symbol $\triangleright$ of a tape with content n, the machine will go through a number of configurations before returning the read/write head to the start cell with the tape now having content $n+1$. Auto-iterating, the machine will thus, over time, produce a tape that will have contained every natural number in order.

This Turing machine may be emulated by a finite $\Sigma\Gamma$ on a set X cf. Theorem 3. To give a general intuition:[18], the idea is that the Turing tape, or a finite fragment, thereof may be encoded as a pointed Kripke model: Each cell of the tape corresponds to a state, with the cell's content encoded by additional

[17] The statement of this proposition in the conference version of this paper [33] contained an embarrassing conflation of concepts.

[18] The details differ depending on whether $\Sigma\Gamma$ must be static, but non-Boolean for Prop. 10, or Boolean, but non-static for Prop. 11. See Section 6 and [15].

structure,[19] which is modally expressible. By structuring the cell states with two equivalence relations and atoms u and e true at cells with odd (even) index respectively, (cf. Figure 2), also the position of a cell is expressible. The designated state corresponds to the start cell, marked $\triangleright$.

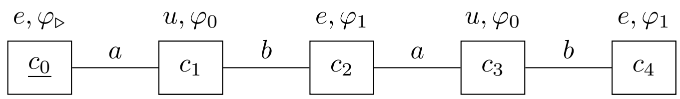

Fig. 2. A pointed Kripke model emulating the configuration of the Turing machine with cell content representing the number 10. The designated state is the underlined c_0. Each state is labeled with a formula $\varphi_\triangleright$, φ_0 or φ_1 expressing its content. Relations a and b allow expressing distance of cells: That c_0 satisfies $\Diamond_a(u \wedge \Diamond_b(e \wedge \varphi_1))$ exactly expresses that cell c_2 contains a 1. Omitted are reflexive loops for relations, and the additional structure marking cell content and read/write head position.

Let $(c_n)_{n\in\mathbb{N}_0}$ be the sequence of configurations of the machine when initiated on a tape with content 0. Each c_n may be represented by a pointed Kripke model, obtaining a sequence $(x_n)_{n\in\mathbb{N}_0}$. By Theorem 3, there thus exists a $\Sigma\Gamma$ such that for all n, $x_n \otimes \Sigma\Gamma = x_{n+1}$. Hence, moving to the full modal space $\boldsymbol{X}$ for the language used, a clean map $\boldsymbol{f}\colon \boldsymbol{X} \to \boldsymbol{X}$ based on $\Sigma\Gamma$ will satisfy $\boldsymbol{f}(\boldsymbol{x}_n) = \boldsymbol{x}_{n+1}$ for all n. The Turing machine's run is thus emulated by $(\boldsymbol{f}^k(\boldsymbol{x}_0))_{k\in\mathbb{N}_0}$.

Let $(c'_n)_{n\in\mathbb{N}_0}$ be the subsequence of $(c_n)_{n\in\mathbb{N}_0}$ where the machine has finished the successor operation and returned its read/write head to its starting position $\triangleright$, ready to embark on the next successor step. The tape of the first 9 of these c'_n are depicted in Figure 1. Let $(\boldsymbol{x}'_n)_{n\in\mathbb{N}_0}$ be the corresponding subsequence of $(\boldsymbol{f}^k(\boldsymbol{x}_0))_{k\in\mathbb{N}_0}$. We show that $(\boldsymbol{x}'_n)_{n\in\mathbb{N}_0}$ has uncountably many limit points:

For each subset Z of $\mathbb{N}$, let c^Z be a tape with content 1 on cell i iff $i \in Z$ and 0 else. On the Kripke model side, let the corresponding $x^Z \in X$ be a model structurally identical to those of $(x'_n)_{n\in\mathbb{N}_0}$, but satisfying φ_1 on all "cell states" distance $i \in Z$ from the designated "$\triangleright$" state, and φ_0 on all other.[20] The set $\{x^{\boldsymbol{Z}} : Z \subseteq \mathbb{N}\}$ is uncountable, and each $\boldsymbol{x}^{\boldsymbol{Z}}$ is a limit point of $\overline{\boldsymbol{x}}$: For each $Z \subseteq \mathbb{N}$ and $n \in \mathbb{N}$, there are infinitely many k for which $x_k \vDash \varphi$ iff $x^Z \vDash \varphi$ for all φ of modal depth at most n. Hence, for every n, the set $\{\boldsymbol{x}_k : d_b(\boldsymbol{x}_k, \boldsymbol{x}^{\boldsymbol{Z}}) < 2^{-n}\}$ is infinite, with d_b the equivalent of the n-bisimulation metric, cf. Prop. 5. Hence, for each of the uncountably many $Z \subseteq \mathbb{N}$, $\boldsymbol{x}^{\boldsymbol{Z}}$ is a limit point of the sequence $\overline{\boldsymbol{x}}$.

Finally, every $\boldsymbol{x}'_k \in (\boldsymbol{x}'_n)_{n\in\mathbb{N}_0}$ is recurrent: That $\boldsymbol{x}'_k \in \omega_{\boldsymbol{f}}(\boldsymbol{x}'_k)$ follows from $\boldsymbol{x}'_k$ being a limit point of $(\boldsymbol{x}'_n)_{n\in\mathbb{N}_0}$, which it is as $\boldsymbol{x}'_k = \boldsymbol{x}^{\boldsymbol{Z}}$ for some $Z \subseteq \mathbb{N}$.[21] As the set of recurrent points is thus infinite, it cannot be periodic. □

As a final result on the orbits of clean maps, we answer an open question: After having exemplified a period 2 system, Sadzik [47] notes that it is unknown whether finite, static, but non-Boolean, clean maps exhibiting longer periods exist. They do:

Proposition 12. *For any $n \in \mathbb{N}$, there exists finite, static, but non-Boolean clean maps with periodic orbits of period n. This is also true for finite Boolean, but non-static, clean maps.*

Proof. For the given n, find a Turing machine that, from some configuration, loops with period n. From here, Theorem 3 does the job. □

Finally, we note that brute force determination of a clean map's orbit properties is not in general a feasible option:

Proposition 13. *The problems of determining whether a Boolean and non-static, or a static and non-Boolean, clean map, a) has a periodic orbit or not, and b) contains a limit cycle or not, are both undecidable.*

Proof. The constructions from the proofs of Theorem 3 allows encoding the halting problem into either question. □

---

[19] For Prop. 11, tape cell content may be encoded using atomic propositions, changable through postconditions, cf. [15]; for Prop. 10, cell content is written by adding and removing additional states, cf. Section 6.

[20] The exact form is straightforward from the constructions used in Section 6 and [15].

[21] A similar argument shows that all x^Z with $Z \subseteq \mathbb{N}$ co-infinite are recurrent points. Hence $\omega_f(\boldsymbol{x}'_k)$ for any $\boldsymbol{x}'_k \in (\boldsymbol{x}'_n)_{n\in\mathbb{N}_0}$ contains uncountably many recurrent points.

6 Turing Completeness of Finite Static Non-Boolean Action Models

This section presents the proof of Theorem 3.1. As all action models of interest for that theorem have trivial postconditions for all action—i.e., have $post(\sigma) = \top$ for all $\sigma \in [\![\Sigma]\!]$—this section omits reference to postcondition maps, identifying an action model $\Sigma\Gamma = ([\![\Sigma]\!], \mathsf{R}, pre, post, \Gamma)$ with the sub-tuple $([\![\Sigma]\!], \mathsf{R}, pre, \Gamma)$. Moreover, all action models involved are finite and deterministic. In parlance, we follow the seminal [2] and refer to such action models as **epistemic programs**.[22]

6.1 Preliminaries

Define a **Turing machine** as a 6-tuple

$$\mathsf{M} = (Q, q_0, q_h, L, b, \delta)$$

where Q is a finite set of **states** with $q_0 \in Q$ the **start state** and $q_h \in Q$ the **halt state**, $L = \{\lambda_1, ..., \lambda_n\}$ a finite set of **tape symbols** (or 'labels') with $b \in L$ the **blank symbol** and δ a partial function

$$\delta : Q \times L \to Q \times L \times \{l, h, r\}$$

with $\delta(q_h, \lambda)$ undefined for all $\lambda \in L$, called the **transition function**. If $\delta(q, \lambda)$ is undefined, the machine will halt.

A Turing machine acts on a bi-infinite **tape** with cells indexed by $\mathbb{Z}$ and labeled with L such that only b occurs on the tape infinitely often. With the machine in state $q \in Q$ and reading label $\lambda \in L$, the transition function determines a possibly new state of the machine $q' \in Q$, a symbol s' to replace s at the current position on the tape, and a movement of the metaphorical "read/write head": Either one cell to the left (l), none (stay here, h), or one cell to the right (r).

A **configuration** of a machine is fully given by *i)* the current labeling of the tape, *ii)* the position of the $^r\!/_w$-head on the tape, and *iii)* the state of the machine. The space of possible configurations of a machine M is thus $\mathfrak{C} = \mathfrak{T} \times \mathbb{Z} \times Q$, where $\mathfrak{T}$ is the set of bi-infinite strings $\mathfrak{t} = (\dots, \lambda_{-2}, \lambda_{-1}, \lambda_0, \lambda_1, \lambda_2, \dots)$ over L such that only b occurs infinitely often in $\mathfrak{t}$. The transition function δ of M may thus be recast as a partial function $\delta : \mathfrak{C} \to \mathfrak{C}$.

Below, we show that a dynamic run of a Turing machine can be completely represented by a *finite* epistemic program acting on a set of *finite* Kripke models. Intuitively, the Kripke models encodes the physical realization of the machine, and the epistemic program its behavior. More specifically, each Kripke model represents an entire configuration of the machine, i.e., the current labelling of the tape, together with the position of the $^r\!/_w$-head and the state of the machine. For the state space to be finite, however, it cannot represent the full tape. Rather, we only take a finite section thereof: the furthest apart cells with non-blank symbols and all in between, plus finite blank segments on each end. Beyond writing on the tape, moving the $^r\!/_w$-head and changing the machine state, the epistemic program additionally extends the tape in each step in order to ensure that the $^r\!/_w$-head never reaches the tape's end.

To succeed with this construction, we recast the transition function δ in a manner that ignores blank ends of the tape. Each tape $\mathfrak{t}$ has infinite head and tail consisting solely of bs. Ignoring all but a finite segment of these yields a finite non-unique representation of the tape. Formally, for a string $\mathfrak{t} = (\dots, \lambda_{-2}, \lambda_{-1}, \lambda_0, \lambda_1, \lambda_2, \dots)$ and $k < k'$, let $\mathfrak{t}_{\restriction[k,k']}$ be the substring $(\lambda_k, \dots, \lambda_{k'})$. The set of all finite representations of $\mathfrak{T}$ is then given by

$$T = \{t = (\lambda_k, \dots, \lambda_{k'}) \colon k, k' \in \mathbb{N} \text{ and } \exists \mathfrak{t} \in \mathfrak{T} \text{ s.t. } t = \mathfrak{t}_{\restriction[k,k']} \text{ and } \forall j < k, \forall j' > k', \mathfrak{t}_j = \mathfrak{t}_{j'} = b\}.$$

Each $t \in T$ corresponds to a unique $\bar{\mathfrak{t}}(t) \in \mathfrak{T}$. Conversely, each configuration $\mathfrak{c} = (\mathfrak{t}, i, q) \in \mathfrak{C}$ may be represented by the equivalence class $\{(\mathfrak{t}_{\restriction[k,k']}, i, q) \colon k < k', \bar{\mathfrak{t}}(t) = \mathfrak{t}\}$ of its finite approximations. In each such equivalence class, there exists representatives for which the position i of the read/write head is "on the tape", i.e., satisfies that $\lambda_i \in t$. We impose this as a requirement and define a restricted equivalence class for each $\mathfrak{c} = (\mathfrak{t}, i, q) \in \mathfrak{C}$ by

$$[\mathfrak{c}] = \{(\mathfrak{t}_{\restriction[k,k']}, i, q) \colon k \le i \le k', \bar{\mathfrak{t}}(t) = \mathfrak{t}\}.$$

[22] In Baltag and Moss' [2], epistemic programs (multi-pointed action models) are not necessarily deterministic.

With $\mathsf{C} = \{[\mathfrak{c}] \colon \mathfrak{c} \in \mathfrak{C}\}$, i.e., the set of equivalence classes of finite representations of configurations for which the read/write head is on the finite tape, the transition function may finally be recast as a partial function $\delta : \mathsf{C} \to \mathsf{C}$.

6.2 Proof

To prove Theorem 3.1, it must be shown that any Turing machine $\mathsf{M} = (Q, q_0, q_h, L, b, \delta)$ can be simulated by an epistemic program. We show that as follows: First, we define an invertible operator $\mathtt{K}$ that for any finite representation of a configuration $\mathsf{c} \in [\mathfrak{c}] \in \mathsf{C}$ produces a pointed Kripke model $\mathtt{K}(\mathsf{c})$. Second, we define an epistemic program $\Sigma\Gamma$ that corresponds to the Turing machine M, i.e., which satisfies that

$$\mathtt{K}^{-1}(\mathtt{K}(\mathsf{c}) \otimes \Sigma\Gamma) \in \delta([\mathfrak{c}]), \tag{1}$$

for any $[\mathfrak{c}] \in \mathsf{C}$. Hence $\Sigma\Gamma$ may be used to calculate the trajectory of M.

Remark 3. The class of Turing machines with $L = \{0, 1\}, b = 0$, is Turing complete, cf. [46]. For simplicity, the proof restricts attention to this class. Nothing prevents using the constructions with additional symbols. The construction uses $|Q| + 3$ agents to model a Turing machine with states Q. In Section 6.3, we sketch a construction that for any Q requires only three $S5$ agents. The main proof, however, is carried out with the less economical construction as it is overall simpler.

Machine, Language and Logic. Fix a Turing machine M with states Q, and fix from this a set of relation indices $Q' = Q \cup \{a, b, 1\}$. Let the modal language $\mathcal{L}$ be based on a single atom p and operators $\Box_i, i \in Q'$.

Configuration Space. First, we fix the set of Kripke models the machine (epistemic program) works on. Each represents a configuration, i.e., a finite but sufficiently large section of the tape, together with the tape's content, the current position of the ${}^r\!/_w$-head and the state of the machine. We exemplify the construction to be in Figure 3.

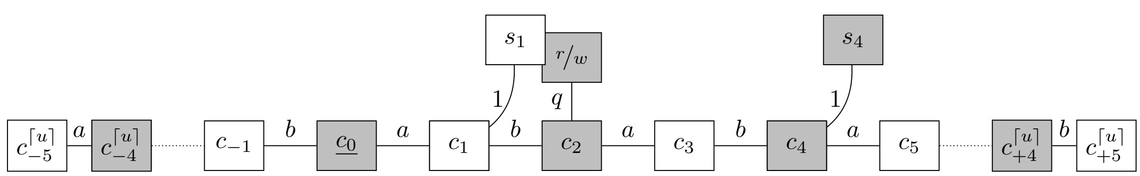

Fig. 3. An emulation of a Turing machine. All boxes are worlds; the underlined c_0 is the actual world. Worlds labeled with c represent tape cells, worlds labeled with s represent symbols written on tape cells, and world ${}^r\!/_w$ represents the read/write head. The cell-worlds labeled with $\lceil u \rceil$ represent the outermost cells of the (finite) tape: $c_{-5}^{\lceil u \rceil}$ and $c_{+5}^{\lceil u \rceil}$ represent respectively the cells in position $-(\lceil u \rceil + 5)$ and $\lceil u \rceil + 5$. Cells 1 and 4 are marked with symbol 1, cf. the 1 relation between c_1 and s_1 and between c_4 and s_4. The remaining cells are blank. The q relation connecting ${}^r\!/_w$ to c_2 represents the machine's state, viz the machine is in state q with the ${}^r\!/_w$-head at cell 2. For relations, reflexive and transitive links are omitted. In the dark worlds, the atom p is true: the atom bears no Turing machine interpretation, but allows differentiating between going left or right along the tape when moving along a, b relations.

The tape itselfs consists of a number of worlds c_i arranged into a linear chain by two $S5$-relations, a and b (cf. Figure 3). These worlds by default represent blank cells. To represent that a cell c_i has content 1 (the unique non-empty symbol), it is connected to a dedicated world s_i with a relation labeled by the symbol 1. Finally, to represent that cell c_j is the unique current position of the ${}^r\!/_w$-head, c_j is connected to a dedicated world h_j by a relation labeled by the machine's current state, $q \in Q$.

Formally, let $\mathsf{C} = \{[\mathfrak{c}] \colon \mathfrak{c} \in \mathfrak{C}\}$ be the set of equivalence classes of finite representations of configurations for which the read/write head is on the finite tape for M and let $\mathsf{c} = (t, i_{r/w}, q) \in \mathsf{C}$. We construct a pointed Kripke model $\mathtt{K}(\mathsf{c})$ representing $(t, i_{r/w}, q)$.

First, in three steps, we construct the set of worlds, cf. Figure 3: *i)* Construct slightly too many "tape cells": Let $\lceil u \rceil = 2 \cdot \max\{|k|, |k'|\}$ and take a set of worlds $C = \{c_j \colon -(\lceil u \rceil + 5) \leq j \leq \lceil u \rceil + 5\}$.

Property	Formula
Being a cell†	$c := (\Diamond_a p \lor \Diamond_b p) \land \Diamond_a \neg p$
Being a 1 symbol	$s := \neg c \land \Diamond_1 c$
Being a cell with symbol 1	$1 := c \land \Diamond_1 \neg c$
Being a cell with symbol 0	$0 := c \land \neg \Diamond_1 \neg c$
Being the cell of the r/w-head is while the machine is in state q	$h_q := c \land \Diamond_q \neg c$
Being the cell immediately left of the r/w-head while the machine is in state q	$l_q := c \land \neg h_q \land ((p \land \Diamond_a h_q) \lor (\neg p \land \Diamond_b h_q))$
Being the cell immediately right of the r/w-head while the machine is in state q	$r_q := c \land \neg h_q \land ((p \land \Diamond_b h_q) \lor (\neg p \land \Diamond_a h_q))$
Being the cell of/im. left of/im. right of the r/w-head while the machine is in state j and the cell of the r/w-head contains a 1/0	$h_{q1}/l_{q1}/r_{q1}/h_{q0}/l_{q0}/r_{q0}$: Replace c in $h_q/l_q/r_q$ with formula 1/0.
Being a cell at least two cells away from the r/w-head	$h_{\geq 2} := c \land \bigwedge_{q \in Q} (\neg h_q \land \neg l_q \land \neg r_q)$
Being the rightmost† cell	$R := c \land \Box_b \neg p$
Being the leftmost† cell	$L := c \land \Box_a \neg p$
Being the penultimate cell to the right	$PR = c \land \neg R \land \Diamond_a R$
Being the penultimate cell to the right	$PL = c \land \neg L \land \Diamond_b L$
Being at least two steps away from the r/w-head and not being the left- or rightmost cell	$2_{AM} := h_{\geq 2} \land \neg R \land \neg L$

Table 1. Expressible properties used as preconditions. **Notes.** †: Recall that the extreme states of C are $c_{-(\lceil u \rceil + 5)}$ and $c_{\lceil u \rceil + 5}$ with $\lceil u \rceil$ even.

ii) Represent the content of a cell: Add worlds $S = \{s_j : \lambda_j = 1\}$ that represent the unique non-blank "symbol" 1 written on "cells" by a relational link. *iii*) Add a "read/write head" together with the appropriate state. To do so let $H = \{r/w\}$. Finally, we define the set of worlds as $W = C \cup S \cup H$.

Second, we add relations between the worlds. In the following let R^* denote the reflexive, symmetric, and transitive closure of the relation R on a given base set, here W. In particular $(w, w) \in R^*$ for all $w \in W$. We add relations also in three steps: *i*) We structure the cells c_i into a tape using relations R_a and R_b: $R_a = \{(c_j, c_{j+1}) : j \text{ is even}\}^*$, $R_b = \{(c_j, c_{j+1}) : j \text{ is odd}\}^*$. *ii*) We attach the non-blank symbols to the appropriate cells: Let $R_1 = \{(c_j, s_j) : s_j \in S\}^*$. *iii*) We mount the read/write head at the correct position and in the correct state, q: Let $R_q = \{(c_j, r/w) : j = i_{r/w}\}^*$. For the remaining states $q' \in Q \backslash \{q\}$, let $R_{q'} = \{\}^*$.

Third and finally, let $[\![p]\!] = \{c_j, s_j \in C \cup S : j \text{ is even}\} \cup H'$ where $H' = \{r/w\}$ iff $i_{r/w}$ is even and $H' = \emptyset$ else, and let the actual world be c_0.

We thus obtain a pointed Kripke model $\mathtt{K}(\mathsf{c}) = (W, \{R_i\}_{i \in Q'}, [\![\cdot]\!], c_0)$ for the finite configuration representation c of Turing machine M. Figure 3 illustrates this, depicting the model $\mathtt{K}(\mathsf{c})$ for configuration $\mathsf{c} = (t, 2, q_0)$. Given $\mathtt{K}(\mathsf{c})$, we may clearly invert the construction process and re-obtain an element of $[\mathsf{c}]$. Finally let $\mathcal{C} = \{\mathtt{K}(\mathsf{c}) : \mathsf{c} \in \mathsf{C}\}$.

Expressible Properties. The next step is to construct an epistemic program that simulates the Turing machine's transition function $\delta : \mathsf{C} \to \mathsf{C}$, i.e., satisfies Eq. (1). For this, we take advantage of the fact that various properties of configurations are modally expressible on pointed models in the class $\mathcal{C}$. Hence, we can use these as preconditions. The relevant properties and formulas are summarized in Table 1. For each of the formulas, it is straightforward to check that they express the desired property on $Ms \in \mathcal{C}$, given the hint †.

Epistemic Program. We construct $\Sigma\Gamma = (\Sigma, \{R_j\}_{j \in Q'}, pre, \Gamma)$, an epistemic program that simulates the Turing machine's transition function $\delta : \mathsf{C} \to \mathsf{C}$, cf. Eq. (1). We argue for the adequacy of the

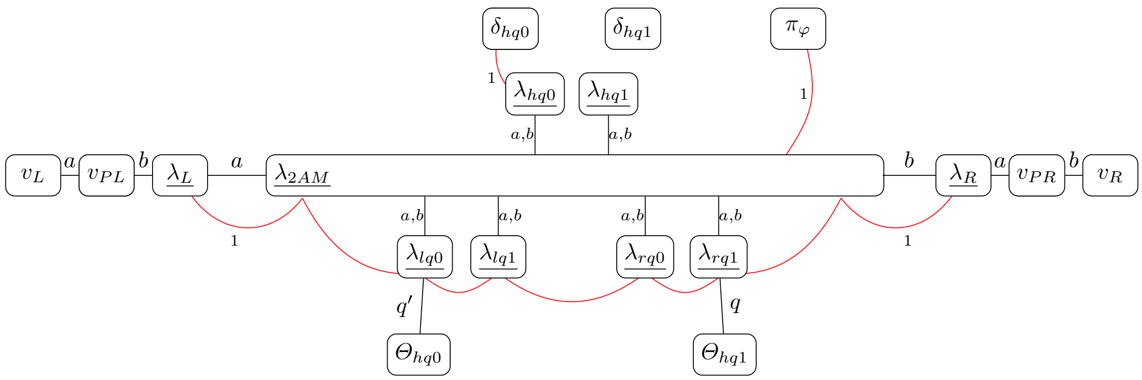

Fig. 4. The epistemic program $\Sigma\Gamma$ for a Turing machine with symbols $L = \{b, 1\}$, states $Q = \{q, q'\}$ and transition function $\delta(q, 0) = (q', 1, l)$, $\delta(q, 1) = (q, 0, r)$ and both $\delta(q', 0)$ and $\delta(q', 1)$ undefined. All rounded corner boxes are actions; the actual actions are underlined. Every action is labeled with it's precondition, viz Table 1. For relations, reflexive and transitive links are omitted. The curvy relation is R_1. That $\lambda_{lq0} R_{q'} \Theta_{hq0}$ ensures that on input $(q, 0)$ the $^r/_w$-head moves to the left and the machine changes to state q', while the relation $\lambda_{hq0} R_1 \delta_{hq0}$ ensures that the content of the current cell is set to 1. Similarly, that $\lambda_{rq1} R_q \Theta_{hq1}$ ensures that on input $(q, 1)$ the $^r/_w$-head moves to the right and the machine stays in state q. The absence of relation $\lambda_{hq1} R_1 \delta_{hq1}$ ensures that the content of the current cell is set to 0. See Figures 5 and 6 for $\Sigma\Gamma$ applied to the configuration of Figure 3.

epistemic program in parallel with its construction. In the following, the precondition of action σ_φ is the formula φ.

Before the formal construction, let us provide some intuitions about the epistemic program $\Sigma\Gamma$, illustrated using Figure 4. The epistemic program undertakes four tasks:

i) copying the (finite) tape and gradually enlarging it in order to guarantee that its end is never reached by the $^r/_w$-head,
ii) copying the labeling on the tape outside the current $^r/_w$-head's position, and
iii) emulating the $^r/_w$-head's action by
  a) symbol re-writing at the $^r/_w$-head's current position and
  b) moving and changing the state of the machine, and halting.

Task *i*), copying and enlarging the tape, is achieved by the λ- and ν-actions in Figure 4, respectively. Each tape-cell satisfies the precondition of exactly one λ-action, and is thus copied once. Moreover, the tape will be extended by two cells each to the left and right, by the the four ν-actions, two each on the left and right. Task *ii*), copying the current labeling outside the $^r/_w$-head's position is done by action π_φ on the top right. Task *iii*) subtasks *a*) and *b*) that jointly emulate the Turing machine's action, is performed by the δ- and Θ-actions on the top and bottom respectively. Figures 5 and 6 provide a detailed illustration of how this epistemic program is applied to a machine configuration.

Task *i*): Copying and Enlarging the Tape (see Fig. 5). The set of actual actions copies the tape cells of the previous tape. To successfully copy the tape, we use a number of actions with mutually exclusive preconditions distinguishing e.g. whether a cell is at the extreme end of the current tape, or at the current $^r/_w$-head's position. In the latter case, preconditions also take into account the current state of the machine as well as the content at the $^r/_w$-head's position. Formally, let the set of actual actions be given by $\Gamma = \{\lambda_\varphi \colon \varphi \in \Phi\}$ with $\Phi = \{R, L, 2_{AM}\} \cup \{h_{qi}, l_{qi}, r_{qi} \colon q \in Q, i \in \{0, 1\}\}$, cf. Table 1.

Then, for any $\mathtt{K}(\mathtt{c}) \in \mathcal{C}$, for every cell state $c_j \in C$ of $\mathtt{K}(\mathtt{c})$, c_j will satisfy exactly one of the formulas in Φ. As the actual world of $\mathtt{K}(\mathtt{c})$ is the cell state c_0, $\Sigma\Gamma$ is thus deterministic over $\mathcal{C}$, and the actual world of $\mathtt{K}(\mathtt{c}) \otimes \Sigma\Gamma$ is a cell state. Finally, formulas from Φ are only satisfied at cell states of $\mathtt{K}(\mathtt{c})$. Jointly, this implies that Γ "copies" the set of tape cells from $\mathtt{K}(\mathtt{c})$ to $\mathtt{K}(\mathtt{c}) \otimes \Sigma\Gamma$.

The copied over tape may not be long enough for future operations, so we include a set of actions to preemptively enlarge it.[23] To this end, let $\Upsilon = \{\upsilon_L, \upsilon_{PL}, \upsilon_R, \upsilon_{PR}\}$, corresponding to the two leftmost

[23] To save tape, this could be done in a more economical manner, only creating extra cells where actually needed.

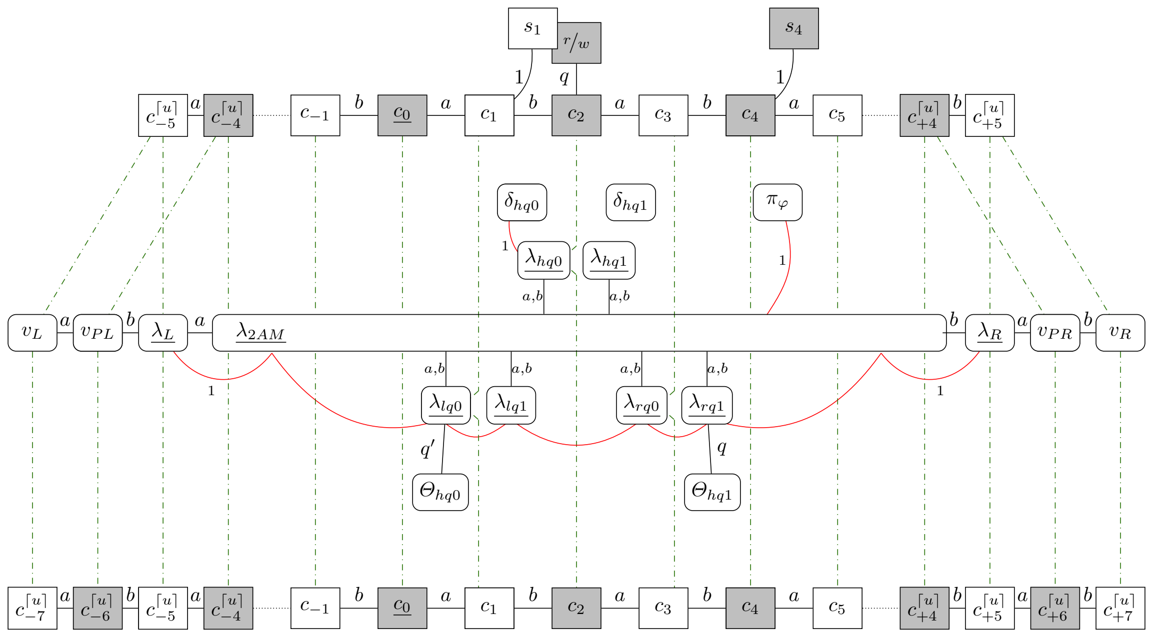

Fig. 5. The epistemic program $\Sigma\Gamma$ of Figure 4 copying and enlarging the tape configuration of Figure 3. The input configuration (top) is transformed into the output configuration (below). For simplicity, worlds in the product model are not labeled as world-action pairs. Instead, dotted lines show what world-action pairs the new worlds stem from: the lower c_{-1} is thus the pair (c_{-1}, λ_{2AM}) while c_1 is (c_1, λ_{lq0}). How $\Sigma\Gamma$ copies cell content, writes with and moves the ${}^r/_w$-head is shown in Figure 6.

and two rightmost actions in Figure 4. The precondition φ of each $\upsilon_\varphi \in \Upsilon$ is satisfied by exactly one state c_j of $\mathtt{K}(\mathtt{c})$ which is a cell state. These cell state will thus have two successors in $\mathtt{K}(\mathtt{c}) \otimes \Sigma\Gamma$: (c_j, λ_φ) defined before and (c_j, υ_φ). We thus gain four new cell states. Setting

$$R_a = \{(\lambda_\varphi, \lambda_\psi)\colon \varphi, \psi \in \Phi\backslash\{L\}\}^* \cup \{(\upsilon_{PR}, \upsilon_R), (\lambda_L, \upsilon_{PL})\}^*$$
$$R_b = \{(\lambda_\varphi, \lambda_\psi)\colon \varphi, \psi \in \Phi\backslash\{R\}\}^* \cup \{(\lambda_R, \upsilon_{PR}), (\upsilon_{PL}, \upsilon_L)\}^*$$

copies over the tape structure and suitably extends it to the new cell states, which are as the left most, penultimate left, penultimate right, and right most tape cells.

Task *ii*): Symbol Transfer (see Fig. 6). We copy all symbols from the old tape to the new, safe for the symbol at the current position of the ${}^r/_w$-head. To this end, add a single action π_φ with $\varphi = s \wedge \neg\Diamond_1(\bigvee_{\Diamond q \in Q} h_q)$. The formula φ is then satisfied in $\mathtt{K}(\mathtt{c})$ exactly at the symbols states $s_j \in S$ where the ${}^r/_w$-head currently is *not*. Let $\Gamma' = \{\lambda_\varphi\colon \varphi \in \Phi\}$ with $\Phi = \{R, L, 2_{AM}\} \cup \{l_{qi}, r_{qi}\colon q \in Q, i \in \{0,1\}\}$. Requiring that $(\Gamma' \times \{\pi_\varphi\})^* \subseteq R_1$ ensures that the symbol states copied over to $\mathtt{K}(\mathtt{c}) \otimes \Sigma\Gamma$ are connected to the correct cell world. We give the precise definition of R_1 below.

Task *iii, a*): Symbol Writing (see Fig. 6). We implement the symbol writing part of the transition function δ. Define a new set of actions by

$$\Delta = \{\delta_{h_{qi}}\colon q \in Q, i \in \{0,1\}\}.$$

At most one action from Δ will have its precondition satisfied at any $\mathtt{K}(\mathtt{c})$ and just in case $\delta(\mathtt{c})$ is defined. Moreover, an action $\delta_{h_{qi}}$ can only have it's precondition satisfied in $\mathtt{K}(\mathtt{c})$ if q is the machine's current state and the current ${}^r/_w$-head's cell has content i. The world satisfying this precondition is a cell world, c_j, which will have two successors in $\mathtt{K}(\mathtt{c}) \otimes \Sigma\Gamma$:[24] a cell world successor $(c_j, \lambda_{h_{qi}})$ defined

[24] Possibly three, see below.

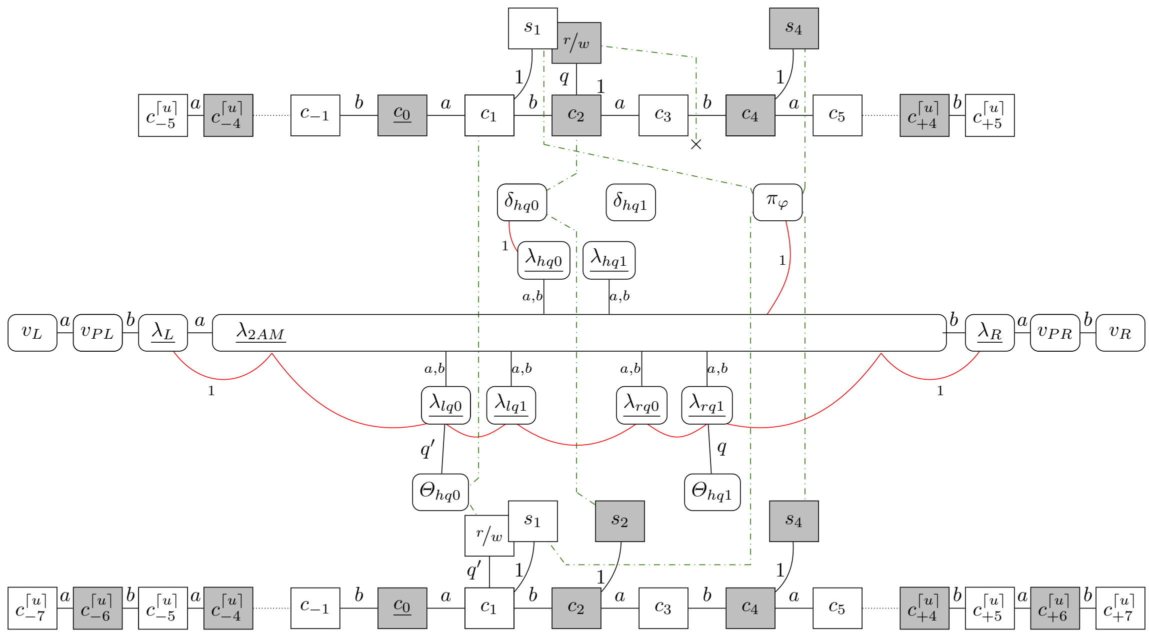

Fig. 6. The epistemic program $\Sigma\Gamma$ of Figure 4 copying cell content, writing at the r/w-head's current position and moving the r/w-head corresponding to the transition $\delta(q, 0) = (q', 1, l)$. The top r/w-head world is not copied down. Instead, then new r/w-head is a copy of c_1 by Θ_{hq0}. That the new r/w-head is related to c_1 by q' captures that the machine's state change. A new 1 symbol is written on cell 2 by copying c_2 with δ_{hq0}. The new symbol world s_2 is related to the lower c_2 by R_1; it is a duplicate of the upper c_2 by λ_{hq0}.

above and a symbol world successor $(c_j, \delta_{h_{qi}})$ defined here. We ensure that the emulation writes the correct symbol by connecting $(c_j, \delta_{h_{qi}})$ to $(c_j, \lambda_{h_{qi}})$ by R_1 or not: Let

$$R_{tmp} = \{\{(\delta_{h_{qi}}, \lambda_{h_{qi}})\colon \lambda \in L\} \mid \delta(i, q) \text{ is defined and } \delta(i, q) = (\cdot, 1, \cdot)\}$$

and let $R_1 = ((\Gamma' \times \{\pi_\varphi\}) \cup R_{tmp})^*$. This and the above ensures that the emulation produces a correctly labeled tape.

Task *iii, b*): State Change, Head Repositioning and Halting. We finally implement the state change and head repositioning encoded by δ. To this end, define a set of events

$$\Theta = \{\theta_{h_{qi}} : q \in Q, i \in \{0, 1\}\}.$$

Again, at most one action from Θ will have its precondition satisfied at any $\mathsf{K}(\mathsf{c})$ and just in case $\delta(\mathsf{c})$ is defined. Also again, $\Theta_{h_{qi}}$ can only have it's precondition satisfied in $\mathsf{K}(\mathsf{c})$ if q is the machine's current state and the current r/w-head's cell has content i. The world satisfying this precondition is a cell world, c_j, which will hence have two successors in $\mathsf{K}(\mathsf{c}) \otimes \Sigma\Gamma$:[25] a cell world successor (c_j, λ_φ) defined above and a r/w-head world successor $(c_j, \theta_{h_{qi}})$ defined here. We "mount" the r/w-head world at the correct position and in the correct state using the relations $\{R_{q'}\}_{q' \in Q}$: For all $q' \in Q$, let

$$R_{q'} = \{(\lambda_{xq'}, \theta_{h_{qi}})\colon\ \delta(i, q) \text{ is defined and } \delta(q, i) = (q', \cdot, x), i \in \{0, 1\}, q \in Q\}^*.$$

The definition of $\{R_q\}_{q \in Q}$ ensures that the r/w-head is moved and changes state appropriately, whenever $\delta(i, q)$ is defined. When $\delta(i, q)$ is not defined, the r/w-head world $(c_j, \theta_{h_q i})$ will be disconnected from the tape cell worlds. In that case, $\mathsf{K}(\mathsf{c}) \otimes \Sigma\Gamma$ will not be in $\mathcal{C}$, and the emulation is said to halt. This concludes the construction and proof. □

Remark 4. Having made use of the fact that the class of Turing machines with $L = \{0, 1\}, b = 0$, is Turing complete, cf. [46], our proof shows the following refinement of Theorem 3:

[25] Possibly three, cf. the above.

Theorem 4. *Any Turing machine can be emulated using a set X of finite pointed $S5$ Kripke models for 1 atom and finitely many modalities using a finite, static, multi-pointed action model deterministic over X that only employs preconditions of modal depth at most 3.*

Upper bounds on the number of required modalities may also be found. The proof's construction uses $|Q|+2$ many modalities to represent machine states, plus one for the non-blank symbol. However, the construction generalizes to Turing machines with symbol set $L=\{b,1,\ldots,k\}$ by replacing modality $\Box_1$ with modalities $\Box_1,\ldots,\Box_{\lceil log_2(|L|)\rceil}$ and encoding the symbol content of a cell in binary using the relations $R_1,\ldots R_{\lceil log_2(|L|)\rceil}$, such that cell i having symbol n is represented by cell world c_i being connected by R_j to the world s_i iff the binary encoding of n has 1 in the jth position. Formally, let $S:\{1,\ldots,k\}\to\{1,\ldots,\lceil log_2(|L|)\rceil\}$, be the map defined by $i=\sum_{j\in S(i)}2^j$ and replace the formula 1 of Table 1 with $i:=c\wedge\bigwedge_{j\in\ S(i)}\Diamond_i\neg c\wedge\bigwedge_{j\notin\ S(i)}\neg\Diamond_i\neg c$. Semantically, let $(c_i,s_i)\in R_j$ iff cell i has non-blank label k and $j\in S(k)$. A similar encoding using a map $T:Q\to\{0,1\}^{\lceil log_2(|Q|+1)\rceil}\setminus\emptyset$ reduces the number of modalities needed for representing the machine's state to $\lceil log_2(|Q|+1)\rceil$. The '+1' results as the relations for the machine state also serve to indicate the $^r/_w$'s current position, wherefore at least one such relation must be present. As [46] shows the classes of universal Turing machines with state no.–symbol no. pairs $(4,6)$ non-empty, we obtain the following, bearing in mind that two additional modalities are required to encode the tape.

Corollary 1. *Any Turing machine can be emulated using a set X of finite pointed $S5$ Kripke models for 1 atom, 7 agents using a finite, static, multi-pointed action model deterministic over X that only employs preconditions of modal depth at most 3.*

6.3 Turing Completeness with Three Agents

The above proof required $|Q|+3$ many $S5$ agents, with $|Q|$ the number of states of the machine formalized, while Theorem 4 shows 7 to be an upper bound for the number of modalities needed. However, three agents suffice for Turing completeness, albeit at the price of preconditions' modal depth going up from three to $2\max(|Q|,|L|)+8$. This section sketches a construction establishing the following:

Theorem 5. *Any Turing machine $\mathsf{M}=(Q,q_0,q_h,L,b,\delta)$ can be emulated using a set X of finite $S5$ pointed Kripke models for 1 atom and 3 agents using a finite, static, non-Boolean multi-pointed action model $\Sigma\Gamma$ deterministic over X whose preconditions have modal depth at most $2\max(|Q|,|L|)+8$.*

Proof (Sketch). Let M be a Turing machine with finite enumerated labels $L=\{l_1,l_2,...,l_{|L|}\}$ and states $Q=\{q_1,q_2,...,q_{|Q|}\}$. The same base construction is used for the tape, employing two agents and a single atom p true at every other state, cf. the underlying tape in Figure 7. Rather than encoding tape content, $^r/_w$-head position and machine state by attaching individual worlds using the appropriate relation, we now attach a tree to each tape node c_i, cf. Figure 7. All nodes in the tree share atomic valuation with c_i: p is true if i is even and false else. The tree attached is such that i) the left branch has k nodes (i.e., m^i_k as last element) iff the content of cell c_i is symbol l_k, and ii) the right branch has ℓ nodes (i.e., n^i_ℓ as last element) iff the read-right head is at c_i and the machine is in state q_ℓ. If the $^r/_w$-head is not at c_i, the left branch has length zero, i.e., there are no n^i_c, for any $c>0$.

The language $\mathcal{L}$ can express that a certain node c is a cell, the length of the left branch attached to c is s and the length of the left branch k for any $s,k\in\mathbb{N}$, cf. Table 2. For the needed modal depth, the deepest formula in Table 2 is $\mathsf{L}_{k,K}$ with depth $2\max(k,K)+8$. For a Turing machine with symbols L and states Q, the longest left and right branches have length $|L|$ and $|Q|$, respectively, making the deepest formula needed $\mathsf{L}_{|L|,|Q|}$. With this in hand, an action model similar to the one above can be constructed. We omit the details. $\Box$

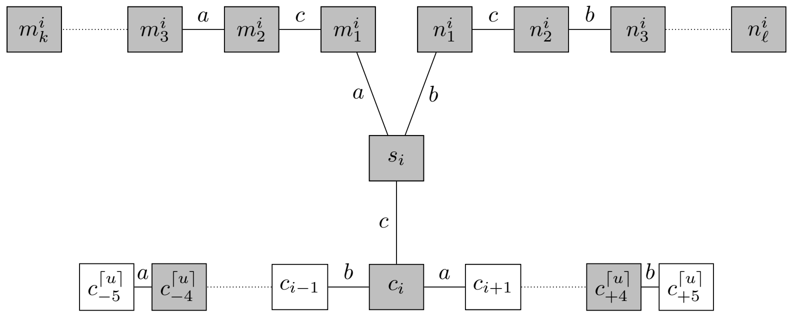

Fig. 7. Writing symbols and state on the tape using only three agents, but long paths. This tree model shows a configuration where the symbol on cell c_i is l_k (left branch) with the $^r/_w$-head also at c_i in state q_ℓ (right branch).

Being a branching node s_i:

$$m := (\neg c \wedge \Diamond_{\mathsf{c}} c)$$

Being a node in the left branch, at most k steps from the tape, at an even (p) cell:

$$\rho^l_k(p) := \neg c \wedge \neg m \wedge \left(\neg p \wedge \left(\underbrace{\Diamond_{\mathsf{a}} \Diamond_{\mathsf{c}} \Diamond_{\mathsf{a}} \ldots}_{k+3 \text{ modalities}} p \vee \underbrace{\Diamond_{\mathsf{c}} \Diamond_{\mathsf{a}} \Diamond_{\mathsf{c}} \ldots}_{k+3 \text{ modalities}} p \right) \right)$$

Being a node in the left branch, at most k steps from the tape, at an odd ($\neg p$) cell:

$$\rho^l_k(\neg p) := \rho^l_k(p)[p \mapsto \neg p, \neg p \mapsto p]$$

Being a node in the left branch, at most k_0 steps from the tape (i.e., being a m^i_k for $k \geq k_0$)

$$\rho^l_k := \rho^l_k(p) \vee \rho^l_k(\neg p)$$

Being a node in the right branch, at most k_0 steps from the tape (i.e., being a n^i_k for $k \geq k_0$)

$$\rho^r_{k_0} := \rho^l_{k_0}[\Diamond_{\mathsf{a}} \mapsto \Diamond_{\mathsf{b}}]$$

Being a left (right) node branch exactly k steps from the tape (i.e. being a m^i_k (n^i_k))

$$\tau^l_k := \rho^l_k \wedge \neg \rho^l_{k+1} \qquad \tau^r_k := \rho^r_k \wedge \neg \rho^r_{k+1}$$

Being a cell c_i with left (right) branch of length at least k (i.e. m^i_k (n^i_k) exists)

$$\sigma^l_k := c \wedge \underbrace{\Diamond_{\mathsf{c}} \Diamond_{\mathsf{a}} \Diamond_{\mathsf{c}} \ldots}_{k+1 \text{ modalities}} \tau^l_k \qquad \sigma^r_k := c \wedge \underbrace{\Diamond_{\mathsf{c}} \Diamond_{\mathsf{b}} \Diamond_{\mathsf{c}} \ldots}_{k+1 \text{ modalities}} \tau^r_k$$

Being a cell c_i with left (right) branch of length exactly k (i.e. m^i_k (n^i_k) is last element)

$$\mu^l_k := \sigma^l_k \wedge \neg \sigma^l_{k+1} \qquad \mu^r_k := \sigma^r_k \wedge \neg \sigma^r_{k+1}$$

Being a cell c_i immediately left (right) of a cell with left branch of length k and right branch of length K

$$\mathsf{L}_{k,K} := c \wedge \neg \mu^l_k \wedge \left(\left(p \wedge \Diamond_a \mu^l_k \wedge \mu^r_K \right) \vee \left(\neg p \wedge \Diamond_b \mu^l_k \wedge \mu^r_K \right) \right)$$

$$\mathsf{R}_{k,K} := L_{k,K}[p \mapsto \neg p, \neg p \mapsto p]$$

Table 2. Expressible properties used as preconditions. $[\cdot]$ marks general replacement: $\varphi[x \mapsto y]$ is the formula φ with all instances of the string x replaced by the string y. Properties defined in Table 1 are not repeated.

Again, given that there is a universal Turing machine that uses 5 states and 5 symbols [46], an immediate consequence of Proposition 5 is:

Corollary 2. *Any Turing machine can be emulated using a set X of finite $S5$ pointed Kripke models for 1 atom and 3 agents using a finite, static, non-Boolean multi-pointed action model $\Sigma\Gamma$ deterministic over X whose preconditions have modal depth at most* 18*.*

Remark 5. We do not know of a better bound for the modal depth of preconditions necessary for encoding a universal Turing machine into a finite, three agent static action model. We conjecture that improvements are possible, e.g. by replacing the linear segments on the top left and right of Figure 7with branching trees.

7 Discussion and Future Venues

We consider Theorem 2 our main conceptual contribution. With it, an interface between the discrete semantics of dynamic epistemic logic with dynamical systems have been provided; thus the former has been situated in the mathematical field of the latter. This paves the way for the application of results from dynamical systems theory and related fields to the information dynamics of dynamic epistemic logic.

The term *nontrivial recurrence* is adopted from Hasselblatt and Katok, [29]. They remark that "[nontrivial recurrence] is the first indication of complicated asymptotic behavior." Propositions 11 and 10 indicate that the dynamics of action models and product update may not be an easy landscape to map. Hasselblatt and Katok continue: "In certain low-dimensional situations [...] it is possible to give a comprehensive description of the nontrivial recurrence that can appear." [29, p. 24]. That the Stone topology is zero-dimensional fuels the hope that general topology and dynamical systems theory yet has perspectives to offer on dynamic epistemic logic. One possible direction is seeking a finer parametrization of clean maps restricting, for instance, to those representable with state machines or other limited models of computation. A further direction is to limit the space of admissible input models to, for instance, those where agents' information is ordered monotonously—a class heavily studied by [47]—combined with results specific to zero-dimensional spaces, as found, e.g., in the field of symbolic dynamics [38, 49]. But also other venues are possible: The introduction of [29] is counts an inspiration.

The approach presented furthermore applies to model transformations beyond multi-pointed action models and product update. Given the equivalence shown in [35] between single-pointed action model product update and *general arrow updates*, we see no reason to suspect that "clean maps" based on the latter should not be continuous on modal spaces. A further conjecture is that the *action-priority update* of [5] on plausibility models[26] yields "clean maps" continuous w.r.t. the suited Stone topology, and that this may be shown using a variant of our proof of the continuity of clean maps.[27] A more difficult case is the *PDL-transformations* of *General Dynamic Dynamic Logic* [25] given the signature change the operation involves.

There is a possible clinch between the suggested approach and epistemic logic with common knowledge. The state space of a dynamical system is compact. The Stone topology for languages including a common knowledge operator is non-compact. Hence, it cannot constitute the space of a dynamical system—but its *compactification* may. We are currently working on this clinch, the consequences of compactification, and relations to the problem of attaining common knowledge, cf. [28]. Concerning the latter, we show in [32] that there exist situations where common knowledge is attained in the limit if, and only if, the underlying language does *not* contain a common knowledge modality.

Questions also arise concerning the *dynamic logic* of dynamic epistemic logic. Propositions 10 and 11 indicate that there is more to the semantic dynamics of dynamic epistemic logic than is representable by finite compositional dynamic modalities—even when including a Kleene star. An open question still stands on how to reason about limit behavior. One interesting venue stems from van Benthem [10]. He notes[28] that the reduction axioms of dynamic epistemic logic could possibly be viewed on par with differential equations of quantitative dynamical systems. As modal spaces are zero-dimensional,

[26] Hence also the multi-agent belief revision policies *lexicographic upgrade* and *elite change*, also known as *radical* and *conservative upgrade*, introduced in [9], cf. [5].

[27] This is corroborated by a recent result by Rendsvig showing that any model transformer with reduction axioms is continuous in its Stone topology (see [45], Paper VI).

[28] In the omitted part of the quotation from the introduction.

they are imbeddable in $\mathbb{R}$ cf. [39, Thm 50.5], turning clean maps into functions from $\mathbb{R}$ to $\mathbb{R}$, possibly representable as discrete-time difference equations.

An alternative approach is possible given by consulting Theorem 2. With this theorem, a connection arises between dynamic epistemic logic and *dynamic topological logic* (see e.g. [23, 24, 36, 37]): Each system $(\boldsymbol{X}_d, \boldsymbol{f})$ may be considered a dynamic topological model with atom set $\mathcal{L}$ and the 'next' operator's semantics given by an application of $\boldsymbol{f}$, equivalent to a $\langle \boldsymbol{f} \rangle$ dynamic modality of DEL. The topological 'interior' operator has yet no DEL parallel. A 'henceforth' operator allows for a limited characterization of recurrence [37]. We are wondering about and wandering around the connections between a limit set operator with semantics $x \vDash [\omega_{\boldsymbol{f}}]\varphi$ iff $y \vDash \varphi$ for all $\boldsymbol{y} \in \omega_{\boldsymbol{f}}(\boldsymbol{x})$, dynamic topological logic and the study of oscillations suggested by van Benthem [11].

With the focal point on pointed Kripke models and action model transformations, we have only considered a special case of logical dynamics. It is our firm belief that much of the methodology here suggested is generalizable: With structures described logically using a countable language, the notion of logical convergence will coincide with topological convergence in the Stone topology on the quotient space *modulo* equivalence, and the metrics introduced will, *mutatis mutandis*, be applicable to said space [34]. The continuity of maps and compactness of course depends on what the specifics of the chosen model transformations and the compactness of the logic amount to.

Acknowledgements The Center for Information and Bubble Studies is funded by the Carlsberg Foundation. Both authors were partially supported by Deutsche Forschungsgemeinschaft (DFG) and Agence Nationale de la Recherche (ANR) as part of the joint project Collective Attitude Formation [RO 4548/8-1], DK also by DFG and Grantová Agentura České Republiky (GAČR) through the joint project From Shared Evidence to Group Attitudes [RO 4548/6-1], by DFG through the network grant Foundations, Applications and Theory of Inductive Logic [432308570], and by the National Science Foundation of China as part of the project Logics of Information Flow in Social Networks [17ZDA026].

References

1. G. Aucher. Generalizing AGM to a multi-agent setting. *Logic Journal of IGPL*, 18(4):530–558, 2010.
2. A. Baltag and L. S. Moss. Logics for Epistemic Programs. *Synthese*, 139(2):165–224, 2004.
3. A. Baltag, L. S. Moss, and S. Solecki. The Logic of Public Announcements, Common Knowledge, and Private Suspicions. In *TARK 1998*. M.Kaufmann, 1998.
4. A. Baltag and B. Renne. Dynamic epistemic logic. In *The Stanford Encyclopedia of Philosophy*. Fall 2016 edition, 2016.
5. A. Baltag and S. Smets. A Qualitative Theory of Dynamic Interactive Belief Revision. In *Proc. LOFT 7*. Amsterdam Univ. Press, 2008.
6. A. Baltag and S. Smets. Group Belief Dynamics Under Iterated Revision: Fixed Points and Cycles of Joint Upgrades. In *TARK'09*. ACM, 2009.
7. J. van Benthem. Games in Dynamic-Epistemic Logic. *Bulletin of Economic Research*, 53(1):219–249, 2001.
8. J. van Benthem. "One is a Lonely Number": Logic and Communication. In *Logic Colloquium '02*, Lecture Notes in Logic, 27. Association for Symbolic Logic, 2006.
9. J. van Benthem. Dynamic logic for belief revision. *Journal of Applied Non-Classical Logics*, 17(2):129–155, 2007.
10. J. van Benthem. *Logical Dynamics of Information and Interaction*. Cambridge University Press, 2011.
11. J. van Benthem. Oscillations, Logic, and Dynamical Systems. In *The Facts Matter*. College Publications, 2016.
12. J. van Benthem, J. van Eijck, and B. Kooi. Logics of communication and change. *Information and Computation*, 204(11):1620–1662, 2006.
13. J. van Benthem, J. Gerbrandy, T. Hoshi, and E. Pacuit. Merging Frameworks for Interaction. *Journal of Philosophical Logic*, 38(5):491–526, 2009.
14. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Camb. Univ. Press, 2001.
15. T. Bolander and M. Birkegaard. Epistemic planning for single- and multi-agent systems. *Journal of Applied Non-Classical Logics*, 21(1):9–34, 2011.
16. T. Bolander, M. Jensen, and F. Schwarzentruber. Complexity results in epistemic planning. In *Proc. IJCAI 2015*. AAAI Press, 2015.
17. T. Caridroit, S. Konieczny, T. de Lima, and P. Marquis. On Distances Between KD45n Kripke Models and Their Use for Belief Revision. In *ECAI 2016*. IOS Press, 2016.

18. C. Dégremont. *The Temporal Mind: Observations on the logic of belief change in interactive systems.* PhD thesis, University of Amsterdam, 2010.
19. H. van Ditmarsch and B. Kooi. Semantic Results for Ontic and Epistemic Change. In *Logic and the Foundations of Game and Decision Theory (LOFT 7)*, Texts in Logic and Games, Vol. 3. Amsterdam University Press, 2008.
20. H. van Ditmarsch, W. van der Hoek, and B. Kooi. *Dynamic Epistemic Logic.* Springer, 2008.
21. T. Eisner, B. Farkas, M. Haase, and R. Nagel. *Operator Theoretic Aspects of Ergodic Theory.* Springer International Publishing, 2015.
22. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge.* The MIT Press, 1995.
23. D. Fernández-Duque. A sound and complete axiomatization for Dynamic Topological Logic. *Journal of Symbolic Logic*, 77(3):1–26, 2012.
24. D. Fernández-Duque. Dynamic topological logic of metric spaces. *Journal of Symbolic Logic*, 77(1):308–328, 2012.
25. P. Girard, J. Seligman, and F. Liu. General Dynamic Dynamic Logic. In *Advances in modal logics, vol 9.* College Publications, 2012.
26. V. Goranko. Logical Topologies and Semantic Completeness. In *Logic Colloquium'99.* Lecture Notes in Logic 17, AK Peters, 2004.
27. V. Goranko and M. Otto. Model Theory of Modal Logic. In *Handbook of Modal Logic.* Elsevier, 2008.
28. Y. J. Halpern and Y. Moses. Knowledge and Common Knowledge in a Distributed Environment. *Journal of the ACM*, 37(3):549–587, 1990.
29. B. Hasselblatt and A. Katok. Principal structures. In *Handbook of Dynamical Systems, Vol. 1A*. Elsevier, 2002.
30. J. Hintikka. *Knowledge and Belief: An Introduction to the Logic of the Two Notions.* College Publications, 2nd, 2005 edition, 1962.
31. P. T. Johnstone. *Stone Spaces.* Cambridge studies in advanced mathematics. Cambridge University Press Cambridge, 1982.
32. D. Klein and R. K. Rendsvig. Converging on common knowledge. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 1741–1748. International Joint Conferences on Artificial Intelligence Organization.
33. D. Klein and R. K. Rendsvig. Convergence, Continuity and Recurrence in Dynamic Epistemic Logic. In A. Baltag and J. Seligman, editors, *Logic, Rationality, and Interaction (LORI 2017, Sapporo)*, volume 10455 of *LNCS*. Springer, 2017.
34. D. Klein and R. K. Rendsvig. Metrics for Formal Structures, with an Application to Dynamic Epistemic Logic. *arXiv:1704.00977*, 2017.
35. B. Kooi and B. Renne. Generalized Arrow Update Logic. In *TARK 2011*, New York, NY, USA, 2011. ACM.
36. P. Kremer and G. Mints. Dynamical Topological Logic. *Bulleting of Symbolic Logic*, 3:371–372, 1997.
37. P. Kremer and G. Mints. Dynamic Topological Logic. In *Handbook of Spatial Logics.* Springer, 2007.
38. D. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding.* Cambridge University Press, 1995.
39. J. R. Munkres. *Topology.* Prentice-Hall, 2nd edition, 2000.
40. J. A. Plaza. Logics of public communications. In *Proceedings of the 4th International Symposium on Methodologies for Intelligent Systems*, 1989.
41. R. K. Rendsvig. Towards a Theory of Semantic Competence. Master's thesis, Dept. of Philosophy and Science Studies and Dept. of Mathematics, Roskilde University, 2011.
42. R. K. Rendsvig. Diffusion, Influence and Best-Response Dynamics in Networks: An Action Model Approach. In *Proc. ESSLLI 2014 Stud.* Online, 2014.
43. R. K. Rendsvig. Pluralistic ignorance in the bystander effect: Informational dynamics of unresponsive witnesses in situations calling for intervention. *Synthese*, 191(11):2471–2498, 2014.
44. R. K. Rendsvig. Model Transformers for Dynamical Systems of Dynamic Epistemic Logic. In *Logic, Rationality, and Interaction*, LNCS. Springer, 2015.
45. R. K. Rendsvig. *Logical Dynamics and Dynamical Systems.* PhD thesis, Theoretical Philosophy, Lund University, Lund, Sweden, 2018. Lund University Publications.
46. Y. Rogozhin. Small universal turing machines. *Theoretical Computer Science*, 168(2):215–240, 1996.
47. T. Sadzik. Exploring the Iterated Update Universe. *ILLC PP-2006-26*, 2006.
48. M. H. Stone. The Theory of Representation for Boolean Algebras. *Transactions of the American Mathematical Society*, 40(1):37–111, 1936.
49. J. de Vries. *Topological Dynamical Systems.* de Gruyter, 2014.