Learning in a changing world, an algebraic approach

Prakash Panangaden* and Mehrnoosh Sadrzadeh**

[1] School of Computer Science, McGill University, Montréal, Canada
prakash@cs.mcgill.ca
[2] Oxford University Computing Laboratory, Oxford, UK
mehrs@comlab.ox.ac.uk

Abstract. We develop an algebraic modal logic that combines epistemic and dynamic modalities with a view to modelling information acquisition (learning) by automated agents in a changing world. Unlike most treatments of dynamic epistemic logic, we have transitions that "change the state" of the underlying system and not just the state of knowledge of the agents. The key novel feature that emerges is the need to have a way of "inverting transitions" and distinguishing between transitions that "really happen" and transitions that are possible.

Our approach is algebraic, rather than being based on a Kripke-style semantics. The semantics are given in terms of quantales. We study a class of quantales with the appropriate inverse operations and prove properties of the setting. We illustrate the ideas with toy robot-navigation problems. These illustrate how an agent learns information by taking actions.

1 Introduction

Epistemic logic has proved very important in the analysis of protocols in distributed systems (see, for example, [FHMV95]) and, more generally in any situation where there is some notion of cooperation or "agreement" between agents. The original work in distributed systems, by Halpern and Moses [HM84,HM90] and several others modelled the knowledge of agents using Kripke-style [Kri63] models. In these models there are a set of states (often called "possible worlds") in which the agent could be and, for each agent, an equivalence relation on the states. If two states are equivalent to an agent then that agent cannot "tell them apart". An agent "knows" a fact ϕ in the state s if, in all states t that the agent "thinks" is equivalent to s, the fact ϕ holds. The quoted words in the preceding sentences are, of course, unnecessary anthropomorphisms that are intended to give an intuition for the definitions.

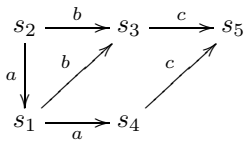
A vital part of any analysis is how processes "learn" as they participate in the protocol. The bulk of papers in the distributed systems community treat this as a change in the Kripke equivalence relations and argue about these changes *only in the semantics*. The logic itself does not have the "dynamic" modalities that refer to updating of the state of knowledge. On the other hand, dynamic epistemic logic has indeed

* Supported by NSERC and the Office of Naval Research.
** Corresponding author; supported by EPSRC grant EP/F042728/1.

been studied; see, for example the recent book [vDvdHK08]. In the second author's doctoral dissertation an algebraic approach to dynamic epistemic logic was studied in depth [BCS07,Sad06].

The advantage of working in the algebraic setting is that it abstracts over the details of the Kripke structures and showcases the high level structure of the actions and their updates. As a result, one can relate the structure of epistemic actions and their updates to the other areas of computer science, e.g. reasoning about correctness of programs and observational logics, e.g. of Abramsky and Vickers [AV93]. In particular, it turns out that the epistemic update is the action of the quantale of programs/actions on the module of propositions (factual and epistemic), hence it is the left adjoint to the dynamic modality which encodes the weakest precondition of Hoare Logic. Secondly (and in a novel attempt), epistemic modalities (too) are encoded as an adjoint pair: the belief modality is the right adjoint of the appearance map, which is the lifting (to subsets) of the accessibility relation of the Kripke structure. Apart from the conceptual novelty and the charm of this adjoint-based approach, it offers a very simple method of reasoning about knowledge acquisition after an action, i.e. by uniform unfolding of epistemic and dynamic adjunctions. This method simplifies, to a great extent, the proofs of complex protocols and puzzles, such as the muddy children, even the versions with dishonest children, for details see [BCS07,Sad06].

The bulk of the work in this area (algebraic and relational), concerns situations where the *state of knowledge* is changed by broadcasts but not situations where the *state of the system* is changed. An illuminating and concrete example of such situations arise in, but are not limited to, robot navigation in AI. The general features of these protocols is that an agent is given the description of a place, but cannot determine where it is; however, it can move and as a result may acquire information that allows it to infer where it is. Consider a robot is given the map of a small computing laboratory with 5 rooms accessible via 3 actions, as follows:



Since the robot can do the same actions in the pairs s_1, s_2 and s_3, s_4, it cannot tell them apart. Once in s_1 (similarly for s_2), it thinks that it could be in s_1 or s_2, and once in s_3 (similarly for s_4), it thinks that it could be in s_3 or s_4. But if once in s_1 it performs an a action, then it reaches s_4 and learns where it is and where it was before moving.

A deeper investigation of such situations reveals that it is not a question of "patching up" the theory. There are some interesting fundamental changes that need to be made. First of all, one has to distinguish between transitions that exist in the agent's "mental model" of the system and actions that *actually occur*. Second, one has to introduce a converse dynamic modality in order to correctly formulate the axioms for updating knowledge. To see why, let us reason as we think the robot should: when it reaches s_4, it checks with its map and reasons that the only way it could have reached s_4 would

be that it was originally in s_1. It rules out s_3 from its uncertainty set about s_4, because, according to the map, it could not have reached s_3 via an a action. We have two types of data here, the locations and actions described on the map versus the ones in reality. The data on the map are hard-coded in the robot and there is no uncertainty about it, the map fully describes the system. But the real locations and actions are only partially known. The robot is uncertain about locations and the actions it takes change its uncertainties. The other issue is that to be able to encode what actions *could have* led the robot to where it is, it needs to look back, so we need a converse operation to reason about the past. Now by moving from s_1 to s_4, the robot has changed its uncertainty, acquired information, and learned where it is located. This is exactly the manner in which our new *uncertainty reduction* axiom formalizes the elimination of past uncertainties: after performing a certain move in the real world, the robot consults its description, considers its possibilities and eliminates the ones that could not have been reached as a result of the action it just performed. Furthermore with this converse operation, we can also derive information about past, that the robot was in s_1 before doing action a.

This paper presents an algebraic theory with these features. The algebra of previous work, e.g. [BCS07] fails for such situations. The reason is that its reduction axiom responsible for changing the uncertainty after an action, is only geared towards epistemic actions and is not powerful enough for fact-changing actions. It requires that the uncertainty about (possible states of) a location after an action to be included in the result of applying the action to the uncertainty about the location beforehand, a property similar to *perfect recall* in protocol models of [HM84,HM90]. This fails here, since after performing an a at s_1 one ends up in s_4, hence uncertainty about s_1 after an a is the same as uncertainty about s_4, consisting of s_3 and s_4. But performing a on the set of uncertainties about s_1, consisting of s_1 and s_2, results in both s_4 and s_1. However, $\{s_4, s_1\}$ is not included in $\{s_3, s_4\}$. Moreover, after the robot moved to s_4, it can conclude that it *was* in s_1 before moving; the language of [BCS07] simply cannot express these *past tense* properties.

Finally, regarding related work, dynamic epistemic logic has been extended with *assignments* and *post-conditions*, e.g. see [vanDit05], to be able to reason about learning after fact-changing actions. Although the protocols we are interested in can be modeled in the relational models of [vanDit05] (these being transition systems with uncertainty as well as action transitions), the reduction axiom thereof cannot derive the knowledge properties we are interested in. This may be because their approach has different kinds of fact-changing actions in mind, e.g. the ones that change the status of a child in the muddy children puzzle from dirty to clean via washing (and not our location-changing actions). Nevertheless, they do not discuss or specify what kind of actions their reduction axiom targets. So there is indeed a gap in modeling and reasoning about the protocols we deal with here. Also, since we use converse actions, there might be connections to a DEL with converse actions, e.g. see [Auch07]. However, a preliminary study seems to indicate that our reduction axiom is still very different from the one developed there. A further exploration of these connections constitutes future work.

We develop an algebraic setting to formalize information acquisition from such navigation protocols. We study special cases of the past and future deterministic action and

converse action operations of the algebra and prove some of their axiomatic properties. We use these to establish connections with temporal algebras of von Karger [vK98], applied to model program evolution. We apply our algebra to model a grid and a map-based navigation protocol and use the axioms to prove that the agent learns where he is and was after moving about. Further applications of our setting are to AI, mobile communication, security, and control theory. We show that our algebraic structure generalizes that of previous work [BCS07], by proving that the latter faithfully embeds in ours. Hence our setting is also strong enough to reason about learning as a result of communication actions.

2 The Algebra of di-Systems

We need to model "actions" and "formulas". The actions are modelled by a quantale while the propositions are a module over the quantale; i.e. actions modify propositions.

Definition 1. *A **quantale** $(Q, \bigvee, \bullet, 1)$ is a sup-lattice equipped with a unital monoid structure satisfying $q \bullet \bigvee_i q_i = \bigvee_i (q \bullet q_i)$ and $\bigvee_i q_i \bullet q = \bigvee_i (q_i \bullet q)$. Instead of an arbitrary sup-lattice we take it to be a* completely distributive prime-algebraic lattice.

Recall that a prime element, or simply "prime", p in lattice has the property that for any x, y in the lattice, $p \leq x \vee y$ implies that $p \leq x$ or $p \leq y$; "prime algebraic" means that every element is the supremum of the primes below it. The restriction to prime algebraic lattices is not a serious restriction for the logical applications that we are considering; it would be a restriction for extensions to probabilistic systems; we will address such issues in future work. The use of algebraicity is to be able to use simple set-theoretic arguments via the representation theorem for such lattices [Win09]. For finite distributive lattices it is not a restriction at all because of Birkhoff's classical representation theorem. Henceforth, we will not explicitly state that we are working with (completely) distributive prime-algebraic lattices.

Definition 2. *A **right-module** over Q is a sup-lattice M with an action of Q on M, $- \cdot -\colon M \times Q \longrightarrow M$ satisfying*

$$(m{\cdot}q){\cdot}q' = m{\cdot}(q{\bullet}q') \qquad m{\cdot}\bigvee_i q_i = \bigvee_i (m{\cdot}q_i) \qquad \bigvee_i m_i{\cdot}q = \bigvee_i (m_i{\cdot}q) \qquad m{\cdot}1 = m$$

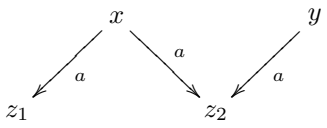
We call the collection of actions and propositions a *system*.

Definition 3. *A **system** is a pair consisting of a quantale Q and a right-module M over Q. We write $(M, Q, \cdot)$ for a system.*

This is closely related to the definition of Abramsky and Vickers who have also argued for the application to Computer Science of quantales of actions, see [AV93]. Like it

is usually done, we interpret elements of the module as *propositions* and the order as entailment, thus $m \vee m'$ is the logical disjunction and $\bot$ is the falsum. The elements of the quantale are interpreted as actions and the order is the order of non-determinism, thus $q \vee q'$ is the non-deterministic choice and $\bot$ is crash, monoid multiplication $q \bullet q'$ is sequential composition, and its unit 1 is the action that does nothing.

Example 1. Consider the following transition system



We model it as a system $(\mathcal{L}(S), \mathcal{M}(A^*), \cdot)$, where A^* is the free monoid generated from the set $A = \{a\}$ with the multiplication being juxtaposition and its unit the empty string. $\mathcal{M}(A^*)$ is the quantale generated on that monoid and $\mathcal{L}(S)$ is the sup-lattice generated from the set $S = \{x, y, z_1, z_2\}$. The most concrete examples of $\mathcal{L}(S)$ and $\mathcal{M}(A^*)$ are $\mathcal{P}(S)$ and $\mathcal{P}(A^*)$. The action on atoms is given by $x \cdot a = z_1 \vee z_2$ and $y \cdot a = z_2$, whereas $z_1 \cdot a = z_2 \cdot a = \bot$. This is extended to juxtaposition and choice (subsets of actions), as well as subsets of states pointwisely.

Example 2. The powerset $\mathcal{P}(S)$ of a set S is the right module of the quantale of all the relations thereon $\mathcal{P}(S \times S)$. Relational composition is the monoid multiplication, the diagonal relation is its unit, and the join is set union. The action is the pointwise image of the relation, i.e. for $W \subseteq S$ and $R \subseteq S \times S$

$$W \cdot R = \bigcup_{w \in W} R[w] = \{z \in W \mid \exists w \in W, (w, z) \in R\}$$

Since the action preserves all the joins of its module, the map $- \cdot q \colon M \longrightarrow M$, obtained by fixing the quantale argument, has a Galois right adjoint that preserves all the meets. This is denoted by $- \cdot q \dashv [q]-$ and defined in the canonical way, as follows:

$$[q]m := \bigvee \{m' \mid m' \cdot q \leq m\}$$

The right adjoints stand for the "dynamic modality" of Hoare logic, encoding the "weakest preconditions" of programs. Each of $[q]m$ is read as "after doing action q or running program q, proposition m holds". This is, in effect, all the propositions that should be true at the input of q such that at its output m holds. One gets very nice logical properties, relating the action and its adjoint to each other and to the $\vee$ and $\wedge$ operators of the lattice and their units $\bot$ and $\top$. Some examples are as follows:

Proposition 1. *The following inequalities hold in any system* $(M, Q, \cdot)$*:*

(1) $([q]m) \cdot q \leq m$ *(2)* $m \leq [q](m \cdot q)$
(3) $(m \wedge m') \cdot q \leq m \cdot q \wedge m' \cdot q$ *(4)* $m \cdot (q \wedge q') \leq m \cdot q \wedge m \cdot q'$
(5) $[q](m \vee m') \geq [q]m \vee [q]m'$
(6) $q \leq q' \implies [q']m \leq [q]m$ *(7)* $[\bot]m = \top$
(8) $[q \vee q']m = [q]m \wedge [q']m$ *(9)* $[q \vee q']m \leq [q]m \vee [q']m$
(10) $[q \wedge q']m \geq [q]m \vee [q']m$ *(11)* $[q \wedge q']m \geq [q]m \wedge [q']m$
(12) $[\bigvee_i q_i]m = \bigwedge_i [q_i]m$

Proof. The proofs are easy but pedagogical, for reasons of space we do not give them here and refer the reader to the full version of the paper [PS10].

Definition 4. *A sup lattice M is a right di-module of the quantale Q whenever there are two right actions $- \cdot -\colon M \times Q \longrightarrow M$ and $- \times -\colon M \times Q \longrightarrow M$. We call the pair of a quantale and its di-module $(M, Q, \cdot, \times)$ a **di-System**.*

Definition 5. *Whenever the two actions – written $\cdot and \cdot^c$ for the purposes of this definition – of a di-system are related by the following three axioms*

$$\begin{array}{lll} (i) & m \cdot q \leq m' \implies m \leq m' \cdot^c q & \text{whenever } m \cdot q \neq \bot \\ (ii) & m \cdot^c q \leq m' \implies m \leq m' \cdot q & \text{whenever } m \cdot^c q \neq \bot \\ (iii) & m \cdot^c (q \bullet q') = (m \cdot^c q') \cdot^c q & \end{array}$$

*then we refer to the di-system as a **converse di-System** and denote it by $(M, Q, \cdot, \cdot^c)$.*

Proposition 2. *A converse di-System satisfies*

$$\begin{array}{lll} m \leq (m \cdot q) \cdot^c q & \text{whenever} & m \cdot q \neq \bot \\ m \leq (m \cdot^c q) \cdot q & \text{whenever} & m \cdot^c q \neq \bot \end{array}$$

Definition 6. *A converse di-System is past-deterministic iff $m \leq m' \cdot q \implies m \cdot^c q \leq m'$, for $m' \cdot q \neq \bot$. It is future-deterministic iff $m \leq m' \cdot^c q \implies m \cdot q \leq m'$, for $m' \cdot^c q \neq \bot$.*

Proposition 3. *In a past-deterministic converse di-System we have $m \leq m' \cdot q \iff m \cdot^c q \leq m'$ for $m' \cdot q, m \cdot^c q \neq \bot$. In a future-deterministic converse di-System we have $m \leq m' \cdot^c q \iff m \cdot q \leq m'$ for $m' \cdot^c q, m \cdot q \neq \bot$.*

Example 3. Consider the transition system of example 1, this is moreover an example of a converse di-System $(\mathcal{L}(S), \mathcal{M}(A^*), \cdot, \cdot^c)$, where the converse action is given by $z_1 \cdot^c a = x$, and $z_2 \cdot^c a = x \vee y$. It is easy to check that these satisfy the inequalities of definition 5, but not their converses: the transition system is neither past-deterministic nor future-deterministic. A counterexample for the converse of part (i) is $x \leq z_2 \cdot^c a$ but $x \cdot a \not\leq z_2$. If we eliminate the leftmost edge, then the system becomes future-deterministic and the converse of (i) holds. A counterexample for the converse of part (ii) is $z_2 \leq y \cdot a$ but $z_2 \cdot^c a \not\leq y$. If we eliminate the rightmost edge, then the system becomes past-deterministic and the converse of (i) holds.

Example 4. The transition system of the introduction is a future-deterministic converse di-System, in the same way as the above example, where $S = \{s_1, \ldots, s_5\}$ and $A = \{a, b, c\}$. It is not past-deterministic, since $s_3 \cdot^c b = s_1 \vee s_2$, also $s_5 \cdot^c c = s_3 \vee s_4$.

Example 5. Consider the setting of example 2, this is also an example of a converse di-System, where the converse action is the point wise image of the converse relation, i.e. for $W \subseteq S$ and $R^c \subseteq S \times S$ converse of R, we have:

$$W \cdot^c R = \bigcup_{w \in W} R^c[w] = \{z \in W \mid \exists w \in W, (w, z) \in R^c\}$$

It is easy to see that $W \cdot^c R = W \cdot R^c$. If R^c is a singleton then this di-system becomes a past-deterministic one, if R is a singleton, it becomes future-deterministic.

The converse action preserves all the joins of the module, thus similar to the action, it has a Galois right adjoint denoted by $- \cdot^c q \dashv [q]^c -$, defined in the canonical way. Similar to $[q]m$, we read $[q]^c m$ as "before doing action q, proposition m held".

We end this section by proving some logical properties that relate the action and its converse to their adjoints. These are of particular interest, since it turns out that in the presence of a Boolean negation on the module, the de Morgan dual of the right adjoint to the action is the converse action, and the de Morgan dual of the right adjoint to the converse action is the action. In other words $- \cdot q$ and $[q]^c -$ are de Morgan duals and so are $- \cdot^c q$ and $[q]-$. Our modules need not necessary be Boolean, nevertheless, these connections can be expressed using the following properties, which axiomatize de Morgan duality in the absence of negation.

Proposition 4. *In any converse di-System we have* $[q](l \vee l') \leq [q]l \ \vee \ l' \cdot^c q$ *and* $[q]^c(l \vee l') \leq [q]^c l \ \vee \ l' \cdot q$. *If it is future-deterministic, we also have* $l \cdot^c q \ \wedge \ [q]l' \leq (l \wedge l') \cdot^c q$. *If its past-deterministic, we also have* $l \cdot q \ \wedge \ [q]^c l' \leq (l \wedge l') \cdot q$.

Proposition 5. *If the module of a past and future deterministic converse di-System is a Boolean algebra with negation operator* $\neg -\colon M \longrightarrow M$, *we have* $m \cdot q = \neg[q]^c \neg m$ *and* $m \cdot^c q = \neg[q]\neg m$.

For details of this, we refer the reader to [PS10]. We have also defined a Kleene star for iteration and shown that it preserves the adjunctions. In the Boolean setting of von Karger [vK98], these iteration operators model modalities of temporal logic.

3 Navigation di-Systems

To distinguish the "potential" actions that happen in the mind of the agent, e.g. actions described by a map, from the "real" actions that take place in the real world, we go higher order. We make real actions act on the di-system that describes potential actions.

Definition 7. *A **second order converse di-System** $((M, Q, \cdot, \cdot^c), Q, \odot, \odot^c)$ is a converse di-System whose module is itself a converse di-System, given by*

$$\begin{array}{lll} - \odot - : & (M, Q, \cdot, \cdot^c) \times Q \longrightarrow (M, Q, \cdot, \cdot^c), & \textit{for} \quad - \odot - \colon M \times Q \longrightarrow M \\ - \odot^c - : & (M, Q, \cdot, \cdot^c) \times Q \longrightarrow (M, Q, \cdot, \cdot^c), & \textit{for} \quad - \odot^c - \colon M \times Q \longrightarrow M \end{array}$$

Potential and real actions have the same labels and both live in the quantale Q. Potential actions change the state of the map via the actions $\cdot$ and $\cdot^c$, real actions change the state of the world via the actions $\odot$ and $\odot^c$. The reason potential and real actions are distinguished from one another is that their targets have different uncertainties. For example, consider the scenario of the introduction, modeled as a converse di-system in example 4. There, the uncertainty of $s_1 \cdot a$ is $s_3 \vee s_4$, whereas the uncertainty of $s_1 \odot a$ is only s_4. So the real actions have an extra significance: they also change the uncertainty of the states. Since real actions cannot be reversed, their converse actions $\odot^c$ is taken to be the same as the converse of the potential action $\cdot^c$. The former, i.e. $\odot^c$, is introduced for reasons of symmetry with the real action, so that we can uniformly use the right adjoints to the second order actions to express the logical properties of "after" and "before"; as we shall see in the sequel section.

To encode the uncertainties, we use *lax* endomorphisms of the system. The reason these are called *lax* is that we require them to satisfy axiomatic inequalities (rather than equalities). These axioms encode the change of uncertainty; the reason they are inequalities has been motivated in [Sad06]. In a nutshell, they are so to be able to encode the process of learning as a decrease in the uncertainty (hence an increase in information).

Definition 8. *A **lax endomorphism** u of a second order converse di-System consists of a pair of endomorphisms $u = (u^M \colon M \longrightarrow M, u^Q \colon Q \longrightarrow Q)$, where u^M preserves joins of M and u^Q preserves joins of Q, moreover we have*

$$\begin{array}{rll} u^M(m \odot q) & \leq \bigvee \{ m' \in M \mid m' \leq u^M(m \cdot q), \quad m' \cdot^c u^Q(q) \neq \bot \} & \textit{(1)} \\ u^Q(q \bullet q') & \leq u^Q(q) \bullet u^Q(q') & \textit{(2)} \\ 1 & \leq u^Q(1) & \textit{(3)} \end{array}$$

We read $u^M(m)$ as the uncertainty about proposition m, the join of all propositions that are possibly true when in reality m is true. For example $u^M(m) = m \vee m'$, says that in reality m is true, but agent considers it possible that either m or m' might be true. Similarly, we read $u^Q(q)$ is the uncertainty about action q, the join of all actions that are possibly happening when in reality action q is happening. E.g. $u^Q(q) = q \vee q'$ says that in reality action q is happening but the agent considers it possible that either q or q' is.

Putting it all together, we define:

Definition 9. *A **Navigation di-System (Nav-diSys)** is a second order converse di-System $((M, Q, \cdot, \cdot^c), \odot, \odot^c, u)$ endowed with a di-System lax endomorphism $u = (u^M, u^Q)$.*

The real action $- \odot q$ changes the uncertainty of a proposition m via inequalities of definition 8. We refer to it as the *uncertainty reduction* axiom. The intuition behind it is as follows: when one does actions in reality, they change our uncertainty. In navigation systems this change is as follows: the uncertainty after performing an action in reality $u^M(m \odot q)$ is the uncertainty of performing a potential action according to the description of the system, i.e. $u^M(m \cdot q)$ minus the choices to which one could not have reached via a q action (according to the description). For example, $u^M(m \cdot q)$ can be a choice of $m' \vee m''$ and it is not possible to reach m' via a q action, i.e. $m' \cdot^c q = \bot$. Hence m' is removed from the choices in $u^M(m \odot q)$, hence $u^M(m \odot q) = m''$. The other two inequalities are for coherence of uncertainty with regard to composition, the motivations for these are as in [BCS07].

Example 6. The transition system of the introduction, i.e. example 4, can be modeled in the following Nav-diSys

$$((\mathcal{P}(\Sigma), \mathcal{P}(A^*), \cdot, \cdot^c), \mathcal{P}(A^*), \odot, \odot^c, u)$$

Here, Σ is obtained by closing the set of states S under product with A, i.e. $\Sigma := S \cup (S \times A) \cup (S \times A \times A) \cup \ldots$. So it contains states $s \in S$, pairs of states and actions $(s, a) \in S \times A$, pairs of pairs of states and actions $((s, a), b) \in (S \times A) \times A)$ and so on. The first order action on states $s \cdot a$ is given by the transitions. This is extended to pairs by consecutive application of the action, i.e. $(s, a) \cdot b$ is given by $(s \cdot a) \cdot b$ and so on. The states action pairs encode the second order actions, i.e. we define the second order action by $s \odot a := (s, a), (s \odot a) \odot b := ((s, a), b), \ldots$ for the atoms and extend it to all the other elements pointwisely, e.g. $s \odot (a \vee b) := (s \odot a) \vee (s \odot b)$ and $s \odot (a \bullet b) := ((s, a), b)$. As for $\odot^c$, for all actions a and states s, we have that $s \odot^c a = s \cdot^c a$.

The lax di-System endomorphism on the module u^M are determined by indistinguishability of states as follows: s, s' are indistinguishable iff the same action a can be performed on them. In formal terms

$$u^M(s) := \{s' \in M \mid \forall a \in A, \quad s \cdot a \neq \bot \quad \text{iff} \quad s' \cdot a \neq \bot\}$$

The u^M of the states updated by potential actions is the u^M of the image, i.e. for the transition system of the introduction, we have $u^M(s_1 \cdot a) = u^M(s_4) = s_4 \vee s_4$. The u^M of states updated by the real action is determined by inequality (1) of definition 8, e.g. $u^M(s_1 \odot a) = u^M(s_1, a) \leq s_4$. The uncertainties of actions, i.e. u^Q can be set similar to that of states: by indistinguishability under application to states. Since for our navigation applications these do not play a crucial role, we assume them to be the identity, i.e. $u^Q(q) = q$ for all $q \in \mathcal{P}(A^*)$. We refer to Nav-diSys' described in this example as *concrete Nav-diSys'* and use them to model scenarios of navigation in the sequel section.

Finally, recall that since each projection of u is join preserving, it has a Galois right adjoint, we focus on the right adjoint of u^M, which we denote by the epistemic modality $\Box$. This is canonically defined as follows

$$\Box m := \bigvee \{m' \in M \mid u^M(m') \leq m\}$$

We read $\Box m$ as 'according to the information available m holds in reality'. Alternatively, one can use the belief modality of doxastic logic and read it as 'it is believed that, or the (sole) agent believes that, m holds in reality'. Putting these modalities together with the dynamic ones, we can express properties such as $[q]\Box m$, read as "after action q the agent believes that m holds", and such as $[q]\Box[q]^c m$, read as "after action q the agent believes that before action q proposition m held", and so on.

4 Applications to Navigation

The scenarios of this section are modeled using the concrete Nav-diSys of example 6.

4.1 Map-based Navigation

For these navigation scenarios, we quotient the concrete Nav-diSys' of map-based navigation protocols over the following property, referred to as Θ:

$$\forall l \in S, \quad \text{if} \quad \exists a \in Ac, \quad \text{s.t.} \quad R_a[l] \neq \emptyset \qquad \text{then} \qquad u^M(l) \supset \{l\}\,.$$

This is to rule out the scenarios which are based on the maps that do not have this property. That is, we assume that all our maps have the property that, all locations in which some action can be done have a non-singleton uncertainty. In other words, if the agent can do some action at a location, then it cannot know that it is actually at that location. The intuition behind this property is that, in the protocols we are interested in agents move to be able to find out where they are, if they already know where they are, then there would be no point in moving and following a protocol.

Consider the navigation protocol of introduction, we encode it in a concrete Nav-diSys with the set of locations $S = \{s_1, s_2, s_3, s_4, s_5\}$, the set of actions $Ac = \{a, b, c\}$, and applicability of actions and uncertainty of states as described there. After quotienting this over Θ, we show that after doing an a action on s_1, the robot knows where it is and where it was before moving.

Proposition 6. *The following hold in a concrete $\mathcal{N}/\Theta$ based on the above data.*

$$s_1 \leq [a]\Box s_4 \qquad\qquad s_1 \leq [a]\Box[a]^c s_1$$

Proof. Consider the first one: by the adjunction $-\odot a \dashv [a]-$, it is equivalent to $s_1 \odot a \leq \Box s_4$. By the adjunction $u^M \dashv \Box$, this is equivalent to $u^M(s_1 \odot a) \leq s_4$. Now by the uncertainty reduction inequality, it is enough to show that

$$\bigvee \{s_i \in S \mid s_i \leq u^M(s_1 \cdot a),\ s_i \cdot^c a \neq \bot\} \ \leq\ s_4$$

Since $s_1 \cdot a = s_4$, and $u^M(s_4) = s_3 \vee s_4$, but $s_3 \cdot^c a = \bot$ where as $s_4 \cdot^c a \neq \bot$, hence the lhs of the above is equal to s_4, which is $\leq s_4$. Consider the second inequality, it becomes equivalent to $u^M(s_1 \odot a) \odot^c a \leq s_1$, by a series of 3 unfoldings of adjunctions. We have shown that $u^M(s_1 \odot a) \leq s_4$, so it suffices to show $s_4 \odot^c a \leq s_1$, which is true since $s_4 \odot^c a = s_4 \cdot^c a = s_1 \leq s_1$.

For an example of a protocol based on the partial map of a city, see [PS10]

4.2 Staircase Navigation

Navigating on the staircase is one of the simplest cases of robot navigation: if the robot is anywhere except for the first and last floor, it does not know where it is. But if it moves to any of these location, it learns where it is and was before moving. We model the n-floor stair case as $n \in \mathbb{N}$ locations $S = \{f_n \mid n \in \mathbb{N}\}$. The atomic actions available to the robot are $Ac = \{up, down\}$.

$$f_1 \underset{\text{down}}{\overset{\text{up}}{\rightleftarrows}} f_2 \underset{\text{down}}{\overset{\text{up}}{\rightleftarrows}} \cdots \underset{\text{down}}{\overset{\text{up}}{\rightleftarrows}} f_{n-1} \underset{\text{down}}{\overset{\text{up}}{\rightleftarrows}} f_n$$

The floors f_2 to f_{n-1} are indistinguishable from one another, i.e. for $1 < i < n$, we have $u^M(f_i) = \bigvee_{1<i<n} f_i$, the first and last floor and the actions have no uncertainty.

Proposition 7. *The following hold in a concrete $\mathcal{N}$ based on the above data*

$$\begin{array}{ll} f_k \leq [up^{n-k}]\Box f_n & f_k \leq [up^{n-k}]\Box[up^{n-k}]^c f_k \\ f_k \leq [dn^{k-1}]\Box f_1 & f_k \leq [dn^{k-1}]\Box[dn^{k-1}]^c f_k \\ f_k \leq [up^{n-k}]\Box[dn^{n-k}] f_k & f_k \leq [dn^{k-1}]\Box[up^{k-1}] f_k \end{array}$$

for $1<n<k$, $up^{n-k} = \underbrace{up \bullet \cdots \bullet up}_{n-k}$, $up^{k-1} = \underbrace{down \bullet \cdots \bullet down}_{k-1}$, *and similarly for* dn^{k-n} *and* dn^{k-1}.

For the proof see [PS10].

4.3 Grid Navigation

A more complex robot navigation protocol happens on the grid: a robot is in a grid with n rows and m columns, it can go up, down, left, and right and is supposed to move about and find out where it is. The grid cells look alike to it as long as it can do the same movements in them, hence it knows where it is iff it ends up in one of the four corner cells. We model this protocol in a concrete Nav-diSys and show that no matter where the robot is, there is always some sequence of movements that it can do to get it to one of the corners. After doing either of these it learns where it is and where it was beforehand.

Each grid cell is modeled by a state s_{ij} in the i'th row and j'th column. Uncertainty of corner states $s_{11}, s_{1m}, s_{n1}, s_{nm}$ is identity, i.e.

$$u^M(s_{11}) = s_{11} \quad u^M(s_{1m}) = s_{1m} \quad u^M(s_{n1}) = s_{n1} \quad u^M(s_{nm}) = s_{nm}$$

For the non-corner cells of the first row and first column, we have

$$u^M(s_{1j}) = \bigvee_{1<y<m} s_{1y} \qquad u^M(s_{i1}) = \bigvee_{1<x<n} s_{x1}$$

For the non-corner cells of last row n and last column m , we have

$$u^M(s_{nj}) = \bigvee_{1<y<m} s_{ny} \qquad u^M(s_{im}) = \bigvee_{1<x<n} s_{xm}$$

For the rest of the cells we have $u^M(s_{ij}) = \bigvee_{\substack{1<x<n \\ 1<y<m}} s_{xy}$. The set of actions is $Ac = \{u, d, l, r\}$, their non-applicability is as follows

$$s_{1j} \cdot u = s_{1j} \cdot^c d = s_{i1} \cdot l = s_{i1} \cdot^c r = s_{nj} \cdot d = s_{nj} \cdot^c u = s_{im} \cdot r = s_{im} \cdot^c l = \bot$$

All the other actions are applicable in all the other states.

Proposition 8. *The following hold in a concrete $\mathcal{N}$ based on the above data.*

$$s_{ij} \leq [\alpha]\Box(s_{11} \vee s_{1m} \vee s_{n1} \vee s_{nm}) \qquad s_{ij} \leq [\alpha]\Box[\alpha]^c s_{ij}$$

for $1 < i < n, 1 < j < m$ and α the following choices of sequences of movements

$$(u^{i-1} \vee d^{n-i}) \bullet (l^{j-1} \vee r^{m-j}) \vee (l^{j-1} \vee r^{m-j}) \bullet (u^{i-1} \vee d^{n-i})$$

For the proof see [PS10].

5 Embedding Epistemic Systems

An algebraic semantics for information learning from communication has been presented in previous work [BCS07], referred to as *Epistemic Systems*. In this section we make the connection between Epistemic Systems and Nav-diSys formal.

Definition 10. *A (mono-modal)* **Epistemic System** $(M, Q, -\otimes-, f)$ *as defined in [BCS07] is a quantale Q acting on its right module M via the action $- \otimes - \colon M \times Q \longrightarrow M$, where $f = (f^M \colon M \longrightarrow M, f^Q \colon Q \longrightarrow Q)$ is a lax system endomorsphism of the setting satisfying the following three inequalities*

$$f^M(m \otimes q) \leq f^M(m) \otimes f^Q(q) \tag{1}$$
$$f^Q(q \bullet q') \leq f^Q(q) \bullet f^Q(q') \tag{2}$$
$$1 \leq f^Q(1) \tag{3}$$

Moreover every element of the quantale $q \in Q$ has a kernel, $ker(q) = \bigvee\{m \in M \mid m \otimes q = \bot\}$ and the module has a special subset $Fact \subseteq M$, defined as $\Phi = \{p \in M \mid \forall q \in Q, p \otimes q \leq p\}$. The module and quantale have a set of atoms $At(M)$ and $At(Q)$ and we have that $At(M) \subseteq \Phi$.

Inequality number (1) is referred to as the *appearance-update* inequality. The kernel of each action encodes the propositions to which the action cannot apply, i.e. if you update those propositions with this action, you will get the $\bot$. Kernels are the opposite of the *preconditions* of actions, as used in the DEL literature, as propositions to which the action can be applied. The facts represent states, and the reason they are stable under updates here is that epistemic actions do not change the state of the world, but only the state of information of agents.

Definition 11. *An* **atomic Nav-diSys**, *similarly* **atomic Epistemic System**, *is one that has an atomic module with set of atoms $At(M)$ and an atomic quantale with a set of atoms $At(Q)$.*

Definition 12. *A* **weak reflexive Nav-diSys** *is an atomic one in which for $s \in At(M), \pi \in At(Q)$ we have $s \leq u^M(s)$ and $\pi \leq u^Q(\pi)$*[3]*.*

Theorem 1. *Given a weak reflexive atomic Nav-diSys $\mathcal{N}$, the structure*

$$\mathcal{N}^\sigma \quad = \quad (M^\sigma, Q^\sigma, - \otimes -, f)_\Phi$$

obtained by setting M^σ to M, Q^σ to Q, f to u, Φ to $At(M)$, and $m \otimes q$ to $m \odot q$, is an atomic Epistemic System.

Proof. We need to show that $\mathcal{N}^\sigma$ satisfies the *appearance-update* axiom. We do so by deriving it from the *uncertainty reduction* axiom of $\mathcal{N}$. In an atomic setting the *uncertainty reduction* axiom becomes equivalent to the following

$$(I) \quad u^M(m \odot q) \leq \bigvee \{s_i \in At(M) \mid s_i \leq u^M(m \cdot q), s_i \cdot^c u^Q(q) \neq \bot\}$$

In the atomic Epistemic System $\mathcal{N}^\sigma$, the *appearance-update* axiom becomes equivalent to the following

$$(II) \quad u^M(m \odot q) \leq \bigvee \{t_j \in At(M) \mid t_j \leq u^M(m), t_j \odot u^Q(q) \neq \bot\}$$

This is a result of atoms becoming facts, that is since $t_j \in \Phi$ we obtain $t_j \otimes u^Q(q) \leq t_j$. We show $(I) \leq (II)$. Take $s_i \leq (I)$, that is $s_i \leq u^M(m \cdot q)$ where $s_i \cdot^c u^Q(q) \neq \bot$. We analyze $u^M(m \cdot q)$ by analyzing $m \cdot q$, which is the same as $m \odot q$ in $\mathcal{N}^\sigma$, and is thus equivalent to

$$m \odot q = \bigvee \{w_k \in At(M) \mid w_k \leq m, w_k \odot q \neq \bot\}$$

From this by monotonicity of u^M, we obtain

$$u^M(m \odot q) = \bigvee \{u^M(w_k) \in At(M) \mid w_k \leq m, w_k \odot q \neq \bot\}$$

From the above and $s_i \leq u^M(m \cdot q) = u^M(m \odot q)$ in $\mathcal{N}^\sigma$ we obtain that $s_i \leq u^M(w_k)$ where $w_k \odot q \neq \bot$. Since $w_k \leq m$ then $u^M(w_k) \leq u^M(m)$, thus $s_i \leq u^M(m)$. Since $w_k \odot q \neq \bot$ and by weak reflexivity from $w_k \leq u^M(w_k)$ and $q \leq u^Q(w_k)$, we have $w_k \odot q \leq u^M(w_k) \odot u^Q(q)$, we obtain that $u^M(w_k) \odot u^Q(q) \neq \bot$, hence $s_i \leq (II)$.

[3] Concrete systems that arise from applications have this property.

Weak reflexive and transitive Nav-diSys's and Epistemic Systems form a pair of categories with morphisms of each being its corresponding lax endomorphisms. In this setting, the above construction becomes a forgetful functor from the latter to the former, most likely having a right adjoint.

6 Conclusions and future work

We have developed an algebraic framework for dynamic epistemic logic in which the dynamic and epistemic modalities appear as right adjoints. The key new feature in the present work relative to previous work [Sad06,BCS07] is the presence of converse actions and the algebraic laws that govern uncertainty reduction. Robot navigation protocols, as well as the three-player game in Phillips's thesis [Phi09], give examples in which the old learning inequality was violated, showing that there were new subtleties that arise when there are actions that really change the state of the world.

A number of directions for future work naturally suggest themselves. On the purely theoretical side, we would like to relate boolean converse di-Systems to Kleene algebras with test and converse. To develop a logic for Nav-diSys, we need to first develop a logic for the algebra of di-Systems. The latter must be similar to the positive fragment of Propositional Dynamic Logic with converse [Par78]. It seems routine to add epistemic modalities to this, the challenge would be to come up with a logical form of the uncertainty reduction axiom. Establishing closer connections to other DEL logics [Auch07,vanDit05] are also worth investigating. We are also particularly interested in extending this work to apply to examples that involve security protocols where "knowledge" and "learning" play evident roles. A fundamental extension, and one in which we have begun preliminary investigations, is the extension to the probabilistic case. Here knowledge and information theory may well merge in an interesting and not obvious way.

Acknowledgements

We have benefited greatly from discussions with Caitlin Phillips and Doina Precup. The latter invented the three-player game and the former discovered the violation of the update inequality. This research was supported by EPSRC (UK) and NSERC (Canada) and the Office of Naval Research.

References

AV93. S. Abramsky and S. Vickers. Quantales observational logic and process semantics. *Mathematical Structures in Computer Science*, 3:161–227, 1993.

Auch07. G. Aucher and A. Herzig. From DEL to EDL : Exploring the Power of Converse Events. *LNCS*, 4724: 199–209, 2007.

BCS07. A. Baltag, B. Coecke, and M. Sadrzadeh. Epistemic actions as resources. *Journal of Logic and Computation*, 17:555–585, 2007. arXiv:math/0608166.

DMS06. Jules Desharnais, B. Müller, and G. Struth. Kleene algebra with domain. *ACM Trans. Comput. Log.*, 7:798–833, 2006.

Dun05. Michael Dunn. Positive modal logic. *Studia Logica*, 55:301–317, 2005.

FHMV95. R. Fagin, J. Y. Halpern, Y. Moses, and M. Y. Vardi. *Reasoning About Knowledge*. MIT Press, 1995.

vanDit05. H.P. van Ditmarsch, W. van der Hoek, B.P. Kooi. Dynamic Epistemic Logic with Assignment. *Proceedings of AAMAS*, 141–148, 2005.

GNV05. M. Gehrke, H. Nagahashi, and Y. Venema. A Sahlqvist theorem for distributive modal logic. *Annals of Pure and Applied Logic*, 131:65–102, 2005.

HKT00. D. Harel, D. Kozen, and J. Tiuryn. *Propositional Dynamic Logic*. MIT Press, 2000.

HM84. J. Y. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. In *Proceedings of the Third A.C.M. Symposium on Principles of Distributed Computing*, pages 50–61, 1984. A revised version appears as IBM Research Report RJ 4421, Aug., 1987.

HM90. J. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *JACM*, 37:549–587, 1990.

Kri63. S. Kripke. Semantical analysis of modal logic. *Zeitschrift fur Mathematische Logik und Grundlagen der Mathematik*, 9:67–96, 1963.

PS10. P. Panangaden and M. Sadrzadeh. Learning in a changing world via algebraic modal logic. http://www.comlab.ox.ac.uk/files/2815/mehrnoosh_prakash.pdf and http://www.cs.mcgill.ca/ prakash/Pubs/mehrnoosh_prakash.pdf

Par78. R. Parikh. The Completeness of Propositional Dynamic Logic. *LNCS* 64:403–415, 1978.

Phi09. C. Phillips. An algebraic approach to dynamic epistemic logic. Master's thesis, School of Computer Sciecne; McGill University, 2009.

Sad06. M. Sadrzadeh. *Actions and Resources in Epistemic Logic*. PhD thesis, Université du Québec à Montréal, 2006.

SD09. M. Sadrzadeh and R. Dyckhoff. Positive logic with adjoint modalities: Proof theory, semantics and reasoning about information. *ENTCS* 23: 211–225, 2009.

vDvdHK08. Hans van Ditmarsch, Wiebe van der Hoek, and Barteld Kooi. *Dynamic Epistemic Logic*. Number 337 in Synthese Library. Springer-Verlag, 2008.

vK98. B. von Karger. Temporal algebras. *Mathematical Structures In Computer Science*, 8:277–320, 1998.

Win09. G. Winskel. Prime algebraicity. *Theoretical Computer Science*, 410:4160–4168, 2009.