Axioms for Knowledge and Time in Distributed Systems with Perfect Recall

Ron van der Meyden

Information Sciences Laboratory,
NTT Basic Research laboratories,
3-9-11 Midori-cho, Musashino-shi,
Tokyo 180, Japan.

email: meyden@ntt-20.ntt.jp

June 25, 1993

Abstract

A distributed system, possibly asynchronous, is said to have perfect recall if at all times each processor's state includes a record of all its previous states. The completeness of a propositional modal logic of knowledge and time with respect to such systems is established. The logic includes modal operators for knowledge, and the temporal operators "next" and "until."

1 Introduction

The recognition that concepts of knowledge are useful in the analysis of distributed systems [1, 2, 7, 8, 18] has motivated the development of a variety of logics for reasoning about knowledge and time [5, 16, 19, 20]. Such logics have turned out to be very sensitive to the underlying assumptions about the distributed system.

Among the most important distinctions are whether the system is synchronous or asynchronous, and whether or not the processors have perfect recall. A processor is said to have perfect recall if its state incorporates a record of all its previous states. An extensive analysis of these properties, and others, has been carried out by Halpern and Vardi [13, 14, 15], who study the complexity of close to 100 different combinations of logics and assumptions regarding the distributed system. In addition to the complexity analysis, they also provide complete axiomatizations in a number of cases, including synchronous systems with perfect recall [12, 14].

In this paper we establish completeness of a logic of knowledge and time with respect to the class of all systems with perfect recall. This class includes asynchronous as well as synchronous systems. It is essentially the class of systems underlying TLP, the Tree Logic of Protocols introduced by Ladner and Reif [16], the logic of Parikh and Ramanujam [19], and Halpern and Fagin's [9] model of knowledge and action in distributed systems. We follow the formulation of Halpern and Vardi, and deal with a language with operators for the knowledge of each processor, in addition to the linear time temporal operators "next" and "until." The language does not include operators for common knowledge.

The completeness proof we present involves an approach different from that employed by Halpern and Vardi to prove completeness for synchronous systems with perfect recall. Our methods may be used to obtain a new and arguably simpler proof of their result. We also obtain as part of this work a new abstract characterization of systems with perfect recall. A prior characterization of Halpern and Vardi suggests an axiom which turns out to be too weak to yield completeness for the present language, although it may suffice in more expressive languages.

Section 2 introduces the language and its semantics, and discusses the perfect recall assumption. Section 3 describes the axiomatization and establishes its soundness. Section 4 is concerned with the proof of completeness. In Section 5 we discuss a different axiomatization which had been conjectured by Halpern and Vardi to yield completeness for systems with perfect recall. Section 6 contains concluding remarks.

2 Syntax and Semantics

In this section we describe the syntax and semantics of the logic of knowledge and time in distributed systems, and introduce three characterizations of systems with perfect recall. We begin by recalling the formalization of knowledge in distributed systems, following the presentation of Halpern and Moses [10].

A *system for m processors* is comprised of a set $\mathcal{R}$ of *runs*. Each run $r \in \mathcal{R}$ is a mapping from the natural numbers $\mathbf{N}$ to $L^m \times E$, where L is a set of local states of the processors, and E is a set of states of the environment. A pair (r, n) consisting of a run r and a natural number n is called a *point*. If $i = 1, \ldots, m$ is a processor number and r is a run then we write $r_i(n)$ for the i-th component of $r(n)$, which corresponds to the local state of processor i at time n in the run r. We write $r_e(n)$ for the $m+1$-st component of $r(n)$, which represents the state of the environment at the point (r, n).

If P is a set of propositional constants then a valuation on P is a function π mapping each constant $p \in P$ to a truth value in $\{true, false\}$. An *interpreted system* $\mathcal{I} = (\mathcal{R}, \pi)$ for m processors consists of a set of runs $\mathcal{R}$ together with an interpretation function π mapping points to valuations on P. Thus, $\pi(r, n)(p)$ denotes a truth value for every point (r, n) and every proposition p.

The local state assignment may be used to define for each processor i an equivalence relation $\sim_i$ on points, by $(r, n) \sim_i (r', n')$ just when $r_i(n) = r'_i(n')$. That is, two points are equivalent according to the relation $\sim_i$ when processor i has the same local state at these points. Intuitively, this means that the processor is unable to distinguish these points on the basis of its knowledge.

The language for describing interpreted systems based on a set of propositional constants P will be the following. Each propositional constant $p \in P$ is a formula. The language is closed under the usual propositional connectives, so if φ_1 and φ_2 are formulae then so are $\neg\varphi_1$ and $\varphi_1 \wedge \varphi_2$. For expressing knowledge of processors, we have for each processor i a monadic modal operator K_i. If φ is a formula and i is a processor then $K_i\varphi$ is a formula, which intuitively states that processor i knows that the formula φ is true. Finally we have two temporal modal operators: if φ_1 and φ_2 are formulae then $\bigcirc\varphi_1$ is a formula expressing that φ_1 will be true at the next point in time, and $\varphi_1 \mathcal{U} \varphi_2$ is a formula which expresses that φ_2 is eventually true, and φ_1 is true until that time.

Formally, the semantics of this language is given by a binary relation $\models$ between points in an interpreted system and formulae: if $\mathcal{I}, (r, n) \models \varphi$ holds we say the formula φ is true at point (r, n) in the interpreted system $\mathcal{I} = (\mathcal{R}, \pi)$. This relation is defined by means of the recursion:

1. If p is a propositional constant then $\mathcal{I}, (r, n) \models p$ if $\pi(r, n)(p) = \mathit{true}$.
2. $\mathcal{I}, (r, n) \models \neg\varphi$ if not $\mathcal{I}, (r, n) \models \varphi$.
3. $\mathcal{I}, (r, n) \models \varphi_1 \wedge \varphi_2$ if $\mathcal{I}, (r, n) \models \varphi_1$ and $\mathcal{I}, (r, n) \models \varphi_2$.
4. $\mathcal{I}, (r, n) \models K_i\varphi$ if $\mathcal{I}, (r', n') \models \varphi$ for all points (r', n') such that $(r, n) \sim_i (r', n')$.
5. $\mathcal{I}, (r, n) \models \bigcirc\varphi$ if $\mathcal{I}, (r, n+1) \models \varphi$.
6. $\mathcal{I}, (r, n) \models \varphi_1 \mathcal{U} \varphi_2$ if there exists $m \geq n$ such that $\mathcal{I}, (r, m) \models \varphi_2$ and $\mathcal{I}, (r, k) \models \varphi_1$ for all k with $n \leq k < m$.

A formula φ is *valid* with respect to a class of interpreted systems if it is true at all points in all interpreted systems that class.

If S is a set, and S^* is the set of all finite sequences over S, we define the absorptive concatenation function $\#$ from $S^* \times S$ to S^* as follows. Given a sequence σ in S^* and an element x of S,

1. if the final element of σ is x then $\sigma\#x = \sigma$, and
2. if the final element of σ is not equal to x then $\sigma\#x$ is σx, i.e. the result of concatenating x to σ.

We will have two distinct uses for this function, applying it to sequences of instantaneous states of processes as well as to sequences of processors.

We say that a system $\mathcal{I}$ is a system with *perfect recall* if the local states L are finite sequences over some set of instantaneous states Q and

1. the state $r_i(0)$ is a sequence of length one for all runs r and processors i, and
2. for every run r, processor i, and natural number n, we have that $r_i(n+1) = r_i(n)\#q$ for some $q \in Q$.

That is, at each point the local state of a processor is the sequence of its distinct instantaneous states up to that point. This is a concrete characterization of perfect recall in terms of the local states, but a more abstract formulation in terms of the knowledge equivalence relations $\sim_i$ is also possible. Say that two interpreted systems are *isomorphic* if there exists a one-to-one correspondence of their runs which preserves the interpretation function π and the relations $\sim_i$.

Proposition 2.1: Let $\mathcal{I}$ be an interpreted system. Then the following are equivalent:

(a) For all processors i, for all runs r, s and for all natural numbers n, m, if $(r, n) \sim_i (s, m)$ then for all $k \leq n$ there exists $l \leq m$ such that $(r, k) \sim_i (s, l)$.

(b) For all processors i, for all runs r, s and for all numbers n, m, if $(r, n+1) \sim_i (s, m)$ then either $(r, n) \sim_i (s, m)$ or there exists a number $l < m$ such that $(r, n) \sim_i (s, l)$ and for all k with $l < k \leq m$ we have $(r, n+1) \sim_i (s, k)$.

(c) $\mathcal{I}$ is isomorphic to a system with perfect recall.

Proof: The implication from (c) to (b) is immediate from the definition of perfect recall. The implication from (b) to (a) is by a straightforward backwards induction. We show that (a) implies (c).

We suppose the system $\mathcal{I} = (\mathcal{R}, \pi)$ satisfies (a), and construct an isomorphic system with perfect recall. Let L be the set of local states of runs in $\mathcal{I}$, and take L^* to be the set of finite sequences over L. Given a run r over states L, define the run r^* with local states in L^* as follows. For processors i, we put $r_i^*(1) = r_i(1)$ and $r_i^*(n+1) = r_i^*(n)\#r_i(n+1)$. The environment component of r is left unchanged: $r_e^*(n) = r_e(n)$. Define the interpretation function π^* by $\pi^*(r^*, n) = \pi(r, n)$ and let $\mathcal{I}^* = (\{r^* | r \in \mathcal{R}\}, \pi^*)$. Evidently $\mathcal{I}^*$ is a system with perfect recall.

We show that the bijection mapping r to r^* makes $\mathcal{I}$ is isomorphic to $\mathcal{I}^*$. It is clear that the interpretation function π is preserved. For the preservation of the knowledge equivalence relations, note that for any run r of $\mathcal{I}$ the final element of the sequence $r_i^*(n)$ is equal to $r_i(n)$. It is therefore immediate that $(r^*, n) \sim_i (s^*, m)$ implies $(r, n) \sim_i (s, m)$. For the converse, we proceed by induction on $m+n$. For the case $m = n = 1$, note that if $(r, 1) \sim_i (s, 1)$ then $r_i^*(1) = r_i(1) = s_i(1) = s^*(1)$, which means that $(r^*, 1) \sim_i (s^*, 1)$. For the inductive part, suppose that $(r, n) \sim_i (s, m)$ and assume that the result has been established for all m', n' such that $m' + n' < m + n$. We will show that $(r^*, n) \sim_i (s^*, m)$.

By condition (a), there exists $k \leq n$ such that $(r, k) \sim_i (s, m-1)$. By the induction hypothesis it follows that $(r^*, k) \sim_i (s^*, m-1)$. Now if either $k = n-1$ or $k = n$

then we have

$$\begin{aligned} r_i^*(n) &= r_i^*(k)\#r_i(n) \\ &= s_i^*(m-1)\#s_i(m) \\ &= s_i^*(m). \end{aligned}$$

Thus we may assume that $k < n-1$. By symmetry, we may also assume that there exists $l < m-1$ such that $(r, n-1) \sim_i (s, l)$ and, by the induction hypothesis, that $(r^*, n-1) \sim_i (s^*, l)$. But then it follows from $l < n-1$ that $r_i^*(n-1) = s_i^*(l)$ is a prefix of $s_i^*(m-1)$. But $s_i^*(m-1) = r_i^*(k)$ is a prefix of $r_i^*(n-1)$, because $k < n-1$. This implies that $r_i^*(n-1)$ is equal to $s_i^*(m-1)$, so it follows using $r_i(n) = s_i(m)$ that $(r^*, n) \sim_i (s^*, m)$. $\Box$

We will work primarily with the property (b). The equivalence of (a) and (c) has previously been stated by Halpern and Vardi [13]. We have included a proof of the implication from (a) to (c) for the sake of later discussion in Section 5.

3 Axiomatization

We now describe the axiomatization we will show to be complete for reasoning about asynchronous systems with perfect recall. It turns out that it suffices to add a single axiom to the axioms and rules of inference already known to be complete for reasoning about knowledge and time individually. For reasoning about knowledge alone, the following axiom schemata

A1. All tautologies of propositional logic.

K1. $K_i\varphi \wedge K_i(\varphi \Rightarrow \psi) \Rightarrow K_i\psi$

K2. $K_i\varphi \Rightarrow \varphi$

K3. $K_i\varphi \Rightarrow K_iK_i\varphi$

K4. $\neg K_i\varphi \Rightarrow K_i\neg K_i\varphi$

are known to be complete [11] when added to the rules of inference

RK. If $\vdash \varphi$ then $\vdash K_i\varphi$

RA. If $\vdash \varphi$ and $\vdash \varphi \Rightarrow \psi$ then $\vdash \psi$.

When there are n processors the resulting logic is known as $S5_n$. For reasoning about the temporal operators individually, the axiom schemata

T1. $\bigcirc(\varphi) \wedge \bigcirc(\varphi \Rightarrow \psi) \Rightarrow \bigcirc\psi$

T2. $\bigcirc(\neg\varphi) \Rightarrow \neg\bigcirc\varphi$

T3. $\varphi\,\mathcal{U}\,\psi \Leftrightarrow \psi \vee (\varphi \wedge \bigcirc(\varphi\,\mathcal{U}\,\psi))$

together with the rules of inference

RT1. If $\vdash \varphi$ then $\vdash \bigcirc\varphi$

RT2. If $\vdash \varphi' \Rightarrow \neg\psi \wedge \bigcirc\varphi'$ then $\vdash \varphi' \Rightarrow \neg(\varphi\,\mathcal{U}\,\psi)$

are known to be complete [3, 6]. The system containing the above axioms and inference rules for both knowledge and time is called $S5_n^U$. It is known that $S5_n^U$ is complete for the class of all systems, as well as for the class of all synchronous systems [12, 14]. To these axiom schemata and inference rules we add the single schema

KT. $K_i\varphi_1 \wedge \bigcirc(K_i\varphi_2 \wedge \neg K_i\varphi_3) \Rightarrow \neg K_i\neg\{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$.

We call the resulting deductive system $S5_n^U + \mathrm{KT}$. We write $\vdash \varphi$ when the formula φ is a theorem of this system. The main result of this paper is the following.

Theorem 3.1: The system $S5_n^U + \mathrm{KT}$ is sound and complete for systems with perfect recall.

That is, for all formulae φ, we have $\vdash \varphi$ if and only if φ is valid with respect to systems with perfect recall. Let us first consider soundness. All axioms and rules of inference other than KT are known to be sound in *all* systems, so their soundness in systems with perfect recall is trivial. The next result establishes soundness of KT.

Lemma 3.2: If (r, n) is a point of a system $\mathcal{I}$ with perfect recall then $\mathcal{I}, (r, n) \models \varphi$ for all instances φ of the schema KT.

Proof: First note that if ψ is any formula and $(r, n) \sim_i (s, m)$ are any two i-equivalent points then $\mathcal{I}, (r, n) \models K_i\psi$ if and only if $\mathcal{I}, (s, m) \models K_i\psi$. To show that KT is sound, we assume that $\mathcal{I}, (r, n) \models K_i\varphi_1 \wedge \bigcirc(K_i\varphi_2 \wedge \neg K_i\varphi_3)$. We will show that $\mathcal{I}, (r, n) \models \neg K_i\neg\{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$. Now it follows from the assumption that $\mathcal{I}, (r, n+1) \models \neg K_i\varphi_3$, so there exists a point (s, m) such that $(r, n+1) \sim_i (s, m)$ and $\mathcal{I}, (s, m) \models \neg\varphi_3$. Since the system has perfect recall, by condition (b) of Proposition 2.1 either (i) $(r, n) \sim_i (s, m)$ or (ii) there exists a number $l < m$ such that $(r, n) \sim_i (s, l)$ and $(r, n+1) \sim_i (s, k)$ for all k with $l < k \leq m$. We claim that in either case $\mathcal{I}, (r, n) \models \neg K_i\neg\{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$. In case (i), it follows form the fact that $\mathcal{I}, (s, m) \models \neg\varphi_3$ that $\mathcal{I}, (s, m) \models \{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$. The desired conclusion is then immediate from the fact that $(r, n) \sim_i (s, m)$. In case (ii), since $\mathcal{I}, (r, n) \models K_i\varphi_1$, we obtain by the observation above that $\mathcal{I}, (s, l) \models K_i\varphi_1$ also. Similarly, because $\mathcal{I}, (r, n+1) \models K_i\varphi_2$, we obtain that $\mathcal{I}, (s, k) \models K_i\varphi_2$ for all numbers k with $l < k \leq m$. Together with $\mathcal{I}, (s, m) \models \neg\varphi_3$, this implies that $\mathcal{I}, (s, l) \models \{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$. Again, since $(r, n) \sim_i (s, l)$, we obtain that $\mathcal{I}, (r, n) \models \neg K_i\neg\{(K_i\varphi_1)\,\mathcal{U}\,[(K_i\varphi_2)\,\mathcal{U}\,\neg\varphi_3]\}$. □

Before considering the completeness proof, let us note that the axiom KT is stronger than the following axiom

KT'. $K_i \Box\varphi \Rightarrow \Box K_i\varphi$

discussed by Ladner and Reif [16]. Here the formula $\Box\varphi$ is an abbreviation of $\neg(\mathit{true}\ \mathcal{U}\ \neg\varphi)$ and asserts that φ is now true and will be true at all future states. Informally, this axiom states that if a proposition is known to be always true, then it will always be known to be true. It is not hard to show using condition (a) of Proposition 2.1 that the axiom KT' holds in systems with perfect recall.

To see that KT' is derivable in $S5_n^U + KT$, note that by purely temporal reasoning, we may establish $\vdash \Box\varphi \Leftrightarrow \Box\Box\varphi$. Using RK and K1 this implies that $\vdash K_i\Box\varphi \Leftrightarrow K_i\Box\Box\varphi$. Now if $\varphi_1 = \varphi_2 = \mathit{true}$ then KT simplifies to $\bigcirc\neg K_i\varphi_3 \Rightarrow \neg K_i\Box\varphi_3$. In particular, taking the contrapositive, substituting $\varphi_3 = \Box\varphi$, and using T2, we obtain $\vdash K_i\Box\Box\varphi \Rightarrow \bigcirc K_i\Box\varphi$, which yields $\vdash K_i\Box\varphi \Rightarrow \bigcirc K_i\Box\varphi$ by the equivalence noted above. It is also straightforward to show $\vdash K_i\Box\varphi \Rightarrow K_i\varphi$. The axiom KT' now follows using the induction axiom RT2.

It had been been conjectured in a draft of the book [4] that the system $S5_n^U + KT'$, would be complete for the class of systems with perfect recall. We will show in Section 5 that this conjecture is false, and that KT is strictly stronger than KT'.

4 Completeness

In this section we establish the completeness result. The proof will proceed by constructing for any consistent formula ψ, a point in a system with perfect recall at which the formula is true.

A formula φ is said to be *consistent* if it is not the case that $\vdash \neg\varphi$. The alternation depth $ad(\varphi)$ of a formula φ is the number of alternations of distinct operators K_i in φ. For the remainder of this section, we fix the consistent formula ψ and let the number d be the alternation depth of ψ.

A finite sequence of processors $\sigma = i_1 i_2 \ldots i_k$, possibly equal to the null sequence ϵ, will be called an *index* if $k \leq d$ and $i_l \neq i_{l+1}$ for all $l < k$. We write $|\sigma|$ for the length k of such a sequence; the null sequence has length equal to 0. Note that if σ is an index and i is a processor then $\sigma\#i$ is also an index provided its length is not greater than d.

For each $k \leq d$ we define the k-closure $cl_k(\psi)$ and for each $k < d$ and processor i we define the k,i-closure $cl_{k,i}(\psi)$. The definition of these sets will be by mutual recursion as follows. First, we let the *basic closure* $cl_0(\psi)$ be the set of subformulae of ψ, together with all formulae $\neg\varphi$, where φ is a subformula of ψ not of the form $\neg\varphi_1$. If i is a processor and $k < d$ we take $cl_{k,i}(\psi)$ to be the union of $cl_k(\psi)$ with the set of formulae of the form $K_i(\varphi_1 \vee \ldots \vee \varphi_n)$ or $\neg K_i(\varphi_1 \vee \ldots \vee \varphi_n)$, where the φ_l are distinct formulae in $cl_k(\psi)$. If $0 \leq k < d$ then $cl_{k+1}(\psi)$ is defined to be the union of the sets $cl_{k,i}(\psi)$ for i a processor.

If U is a finite set of formulae we write φ_U for the conjunction of the formulae in U. A finite set U of formulae is said to be consistent if φ_U is consistent. If U is a finite set of formulae and φ is a formula we write $U \Vdash \varphi$ when $\vdash \varphi_U \Rightarrow \varphi$. Clearly if $U \Vdash \varphi_1$ and $\vdash \varphi_1 \Rightarrow \varphi_2$ then $U \Vdash \varphi_2$.

Suppose C is a finite set of formulae with the property that for all $\varphi \in C$ either $\neg\varphi \in U$ or φ is of the form $\neg\varphi'$ and $\varphi' \in U$. (Note that the sets $cl_k(\psi)$ and $cl_{k,i}(\psi)$ all have this property.) We define an *atom* of C to be a maximal consistent subset of C. Evidently, if U is an atom of C and $\varphi \in C$ then either $U \Vdash \varphi$ or $U \Vdash \neg\varphi$. Thus, we have that $\vdash \bigvee\{\varphi_U \mid U \text{ an atom of } C\}$.

We begin the construction of the model of ψ by first constructing a *pre-model,* which will be a structure $\langle S, \rightarrow, \approx_1, \ldots, \approx_n \rangle$ consisting of a set of states S, a binary relation $\rightarrow$ on S, and for each processor i a binary relation $\approx_i$ on S. The states S will be all the pairs (σ, U) such that σ is an index and

1. if $\sigma = \epsilon$ then U is an atom of $cl_d(\psi)$, and

2. if $\sigma = \tau i$ then U is an atom of $cl_{k,i}(\psi)$, where $k = d - |\sigma|$.

The relation $\rightarrow$ is defined by $(\sigma, U) \rightarrow (\tau, V)$ just when $\tau = \sigma$ and the formula $\varphi_U \wedge \bigcirc \varphi_V$ is consistent. If U is an atom we write U/K_i for the set of formulae φ such that $K_i\varphi \in U$. The relation $\approx_i$ is defined by $(\sigma, U) \approx_i (\tau, V)$ when $\tau = \sigma\#i$ and $U/K_i = V/K_i$.

We will use the term *'σ-states'* to refer to states of the form (σ, U). Thus (σ, U) is the unique σ-state with atom U. If $s = (\sigma, U)$ is a state we define φ_s to be the formula φ_U, and write $s \Vdash \varphi$ for $\vdash \varphi_s \Rightarrow \varphi$. We say that the state s *directly decides* a formula φ if either (a) $\varphi \in U$ or (b) $\neg\varphi \in U$ or (c) $\varphi = \neg\varphi'$ and $\varphi' \in U$. Note that this implies that either $s \Vdash \varphi$ or $s \Vdash \neg\varphi$. In case this latter condition holds we say simply that s *decides* φ. Note that if $\sigma = \tau i$ then each σ-state directly decides every formula in $cl_{d-|\sigma|,i}(\psi)$, and every ϵ-state directly decides every formula in $cl_d(\sigma)$.

When σ is not of the form τi, we have that if s is a σ-state then $s \not\approx_i s$, and furthermore if $s \approx_i t$ then $t \not\approx_i s$. Thus the relations $\approx_i$ are not equivalence relations. However, they are transitive, and also *euclidean,* i.e. if s, t and u are states with $s \approx_i t$ and $s \approx_i u$ then $t \approx_i u$. We will say that states (σ, U) and (τ, V) are *i-adjacent* if $\sigma\#i = \tau\#i$. Clearly, the relation of i-adjacency is an equivalence relation.

Lemma 4.1: Precisely the same formulae of the form $K_i\varphi$ are directly decided by all states i-adjacent to a given σ-state. If $\sigma = \tau j$ with $j \neq i$ and $|\sigma| = d$ then these formulae are those in $cl_0(\psi)$, otherwise they are those in $cl_{k,i}(\psi)$, where $k = d - |\sigma\#i|$.

Proof: Observe first that if $\sigma = \tau j$ with $j \neq i$ and $|\sigma| = d$ then all states i-adjacent to a given σ-state are σ-states, and the K_i-formulae directly decided by these are precisely those in $cl_{0,j}(\psi)$, which consists of $cl_0(\psi)$ and formulae not of the form $K_i\varphi$. On the other hand, if $\sigma = \tau i$ then $\sigma\#i = \sigma$ and it is immediate that the K_i-formulae directly decided by all σ-states are precisely those in $cl_{k,i}(\psi)$, where $k = d - |\sigma\#i|$. It therefore remains to show that if σ is not of the form τi and $|\sigma| < d$ then the formulae of the form $K_i\varphi$ decided by σ-states are precisely those in $cl_{k,i}(\psi)$. Note that in this case $\sigma\#i = \sigma i$, and therefore $k = d - |\sigma\#i| = d - |\sigma| - 1$. Now if σ is of the form τj with $j \neq i$ then the atoms of σ-states are subsets of $cl_{k+1,j}(\psi)$, which consists of $cl_{k+1}(\psi)$ and a set of formulae not of the form $K_i\varphi$. Thus, whether σ is of the form τj with $j \neq i$ or $\sigma = \epsilon$, the K_i-formulae directly decided by σ-states are in $cl_{k+1}(\psi)$. But $cl_{k+1}(\psi)$ is the union of the sets $cl_{k,j}(\psi)$. Thus we see that the K_i-formulae directly decided by σ-states are either in $cl_{k,i}(\psi)$ or in $cl_{k,j}(\psi)$ for some $j \neq i$. In the former case there is nothing to prove. In the latter we obtain by repeating the argument above that any K_i-formula directly decided by a σ-state is in $cl_k(\psi)$. But this set is contained in $cl_{k,i}(\psi)$, so we are done. $\Box$

If s is a state we write $\Phi_{s,i}$ for the disjunction of the formulae φ_t, where t ranges over the states satisfying $s \approx_i t$. Observe that because $\approx_i$ is transitive and euclidean we have that if $s \approx_i t$ then $\Phi_{s,i} = \Phi_{t,i}$. The following result lists a number of knowledge formulae decided by states.

Lemma 4.2:

(1) For all σ-states s and $(\sigma\#i)$-states t, if $s \not\approx_i t$ then $s \Vdash K_i\neg\varphi_t$.

(2) For all states s we have $s \Vdash K_i\Phi_{s,i}$.

(3) For all states s, t with $s \approx_i t$ we have $s \Vdash \neg K_i\neg\varphi_t$.

(4) For all σ-states s and $(\sigma\#i)$-states t, if $s \not\approx_i t$ then $t \Vdash \neg K_i\Phi_{s,i}$.

Proof: For (1), suppose that $s \not\approx_i t$, where $s = (\sigma, U)$ and $t = (\sigma\#i, V)$. Then $U/K_i \neq V/K_i$ so either there exists a formula $K_i\varphi \in U$ such that $K_i\varphi \notin V$ or there exists a formula $K_i\varphi \in V$ such that $K_i\varphi \notin U$. The states s and t are i-adjacent, so by Lemma 4.1, in either case the formula $K_i\varphi$ is directly decided by both the states s and t. In the first case, we have that $\vdash \varphi_t \Rightarrow \neg K_i\varphi$, hence, using RK, that $\vdash K_i(K_i\varphi \Rightarrow \neg\varphi_t)$. By K3 we obtain from $K_i\varphi \in U$ that $s \Vdash K_iK_i\varphi$. It now follows using K1 that $s \Vdash K_i\neg\varphi_t$. In the second case, we have that $s \Vdash \neg K_i\varphi$ and $\vdash \neg K_i\varphi \Rightarrow \neg\varphi_t$. Thus, using K4 we obtain that $s \Vdash K_i\neg K_i\varphi$ and, using RK, that $\vdash K_i(\neg K_i\varphi \Rightarrow \neg\varphi_t)$. It then follows by K1 that $s \Vdash K_i\neg\varphi_t$.

For (2), recall that $\vdash \bigvee\{\varphi_U \mid U \text{ an atom of } cl_{k,i}(\psi)\}$. Hence by RK we obtain that $\vdash K_i \bigvee\{\varphi_t \mid t \text{ a } (\sigma\#i)\text{-state}\}$. It follows from this using (1) and K1 that $s \Vdash K_i\Phi_{s,i}$.

For (3), suppose that $s = (\sigma, U) \approx_i (\sigma\#i, V) = t$ and $k = d - |\sigma\#i|$. We claim first that if $W = V \cap cl_k(\psi)$ then $s \Vdash K_i\neg\varphi_t \Leftrightarrow K_i\neg\varphi_W$. This is because the fact that V is a subset of $cl_{k,i}(\psi)$ implies that all formulae φ in $V \setminus W$ are of the form $K_i\varphi'$ or $\neg K_i\varphi'$, hence $\varphi \in U$ if and only if $\varphi \in V$. Also, by K3 and K4 we have that $s \Vdash K_iK_i\varphi'$ when $K_i\varphi' \in U$ and $s \Vdash K_i\neg K_i\varphi'$ when $K_i\varphi' \notin U$. It follows using K1 that $s \Vdash K_i\varphi_{V\setminus W}$. Since φ_t is equivalent to $\varphi_W \wedge \varphi_{V\setminus W}$, we obtain using K1 that $s \Vdash K_i\neg\varphi_t \Leftrightarrow K_i\neg\varphi_W$.

Now by K2 we have $\vdash \varphi_t \Rightarrow \neg K_i\neg\varphi_t$. Further, the argument of the previous paragraph also shows $t \Vdash K_i\neg\varphi_t \Leftrightarrow K_i\neg\varphi_W$, so we obtain that $t \Vdash \neg K_i\neg\varphi_W$. But $\neg\varphi_W$ is equivalent to the disjunction of a set of formulae $\{\varphi_1, \ldots, \varphi_n\}$ of $cl_k(\psi)$. Let α be the formula $K_i(\varphi_1 \vee \ldots \vee \varphi_n)$, which is equivalent to $K_i\neg\varphi_W$. It follows from the definition of $cl_{k,i}(\psi)$ that α is in $cl_{k,i}(\psi)$, hence directly decided by both t and s. Consequently, α is not in V, since $t \Vdash \neg\alpha$. Because $U/K_i = V/K_i$, the formula α is not in U either, so $s \Vdash \neg\alpha$. Applying the fact that α is equivalent to $K_i\neg\varphi_W$, we see that $s \Vdash \neg K_i\neg\varphi_W$. The equivalence of the previous paragraph now yields that $s \Vdash \neg K_i\neg\varphi_t$.

For (4) note that if t and v are distinct $(\sigma\#i)$-states then $t \Vdash \neg\varphi_v$. Thus, if s is a σ-state such that $s \not\approx_i t$ then $t \Vdash \neg\Phi_{s,i}$, which implies, using K2, that $t \Vdash \neg K_i\Phi_{s,i}$. □

If T is a set of states then we write φ_T for the disjunction of the formulae φ_t for t in T. Using RT1, T1 and T2, the following is direct from the fact that $s \not\to t$ implies $\vdash \varphi_s \Rightarrow \neg\bigcirc\varphi_t$, together with the fact that $\vdash \bigvee\{\varphi_s \mid s \text{ a } \sigma\text{-state}\}$.

Lemma 4.3: Let s be a state and let T be the set of states t such that $s \to t$. Then $s \Vdash \bigcirc\varphi_T$

The next result provides a useful way to derive formulae containing the operator 'until'.

Lemma 4.4: For all formulae α, β and γ, if $\vdash \alpha \Rightarrow \neg\gamma$ and $\vdash \alpha \Rightarrow \bigcirc(\alpha \vee (\neg\beta \wedge \neg\gamma))$ then $\vdash \alpha \Rightarrow \neg(\beta\,\mathcal{U}\,\gamma)$.

Proof: Suppose that $\vdash \alpha \Rightarrow \neg\gamma \wedge \bigcirc(\alpha \vee (\neg\beta \wedge \neg\gamma))$. By T3, we obtain that $\vdash \alpha \wedge (\beta\,\mathcal{U}\,\gamma) \Rightarrow \neg\gamma \wedge \bigcirc(\beta\,\mathcal{U}\,\gamma) \wedge \bigcirc(\alpha \vee (\neg\beta \wedge \neg\gamma))$. Since, by T3 again, $\vdash \beta\,\mathcal{U}\,\gamma \Rightarrow \neg(\neg\beta \wedge \neg\gamma)$, it follows using T1 and RT1 that $\vdash \alpha \wedge (\beta\,\mathcal{U}\,\gamma) \Rightarrow \neg\gamma \wedge \bigcirc(\alpha \wedge (\beta\,\mathcal{U}\,\gamma))$. Now using RT2 we obtain $\vdash \alpha \wedge (\beta\,\mathcal{U}\,\gamma) \Rightarrow \neg(\beta\,\mathcal{U}\,\gamma)$, which implies that $\vdash \alpha \Rightarrow \neg(\beta\,\mathcal{U}\,\gamma)$. □

The following shows that the the pre-model almost satisfies the truth definitions for formulae in the basic closure. Note that every state directly decides all formulae in the basic closure.

Lemma 4.5: For all formulae $\varphi \in cl_0(\psi)$, and all states s,

(1) if φ is of the form $\bigcirc\varphi_1$ then for all states t such that $s \to t$, $s \Vdash \varphi$ iff $t \Vdash \varphi_1$,

(2) if φ is of the form $K_i\varphi_1$ then $s \Vdash \varphi$ iff $t \Vdash \varphi_1$ for all states t such that $s \approx_i t$, and

(3) if φ is of the form $\varphi_1\,\mathcal{U}\,\varphi_2$ then $s \Vdash \varphi$ iff there exists a sequence of states $s = s_0 \to s_1 \to \ldots \to s_n$, where $n \geq 0$, such that $s_n \Vdash \varphi_2$, and $s_m \Vdash \varphi_1$ for all $m < n$.

Proof: For part (1), suppose first that $s \Vdash \bigcirc\varphi_1$ and $s \to t$. Since $\varphi_1 \in cl_0(\psi)$, it follows that $t \Vdash \varphi_1$ or $t \Vdash \neg\varphi_1$. But by T2 the latter would contradict the assumption that $\varphi_s \wedge \bigcirc\varphi_t$ is consistent. Hence we have $t \Vdash \varphi_1$. Conversely, suppose that $t \Vdash \varphi_1$. Since $\varphi \in cl_0(\psi)$ we have either $s \Vdash \varphi$ or $s \Vdash \neg\varphi$. Using T2 again, the latter would contradict $s \to t$, so we obtain $s \Vdash \varphi$.

For part (2), note first that the fact that $K_i\varphi_1$ is in $cl_0(\psi)$ implies that if $s \approx_i t$ and $s \Vdash K_i\varphi_1$ then $t \Vdash K_i\varphi_1$, hence $t \Vdash \varphi_1$ by K2. Conversely, suppose that $t \Vdash \varphi_1$ for all t with $s \approx_i t$. Then $\vdash \Phi_{s,i} \Rightarrow \varphi_1$, hence $\vdash K_i\Phi_{s,i} \Rightarrow K_i\varphi_1$, using K1 and RK. By Lemma 4.2(2) we have $s \Vdash K_i\Phi_{s,i}$. It follows immediately that $s \Vdash K_i\varphi_1$.

For part (3), note that if $\varphi_1 \,\mathcal{U}\, \varphi_2$ is in $cl_0(\psi)$ then every state directly decides each of the formulae φ_1, φ_2 and $\varphi_1 \,\mathcal{U}\, \varphi_2$. We first show that if there exists a sequence of states $s = s_0 \to s_1 \to \ldots \to s_n$ such that $s_n \Vdash \varphi_2$ and $s_m \Vdash \varphi_1$ for all $m < n$ then $s \Vdash \varphi_1 \,\mathcal{U}\, \varphi_2$. For, suppose to the contrary that $s \Vdash \neg(\varphi_1 \,\mathcal{U}\, \varphi_2)$. By T3, we obtain $s \Vdash \neg\varphi_2$ and $s \Vdash \neg\varphi_1 \vee \neg \bigcirc (\varphi_1 \,\mathcal{U}\, \varphi_2)$. In particular, the former yields that $n \neq 0$. Thus $s = s_0 \Vdash \varphi_1$, and the latter implies that $s_0 \Vdash \neg \bigcirc (\varphi_1 \,\mathcal{U}\, \varphi_2)$. Because $s_0 \to s_1$, this means that $\varphi_1 \,\mathcal{U}\, \varphi_2$ is not in the atom of s_1. Consequently, $s_1 \Vdash \neg(\varphi_1 \,\mathcal{U}\, \varphi_2)$. Continuing this argument eventually yields $s_n \Vdash \neg(\varphi_1 \,\mathcal{U}\, \varphi_2)$, which contradicts $s_n \Vdash \varphi_2$.

We now show that if $s \Vdash \varphi_1 \,\mathcal{U}\, \varphi_2$ then there exists a sequence of states $s = s_0 \to s_1 \to \ldots \to s_n$ such that $s_n \Vdash \varphi_2$ and $s_m \Vdash \varphi_1$ for all $m < n$. For, suppose that no such sequence exists, and let T be the smallest set S of states such that (i) $s \in S$, and (ii) if $t \in S$ and $t \to u$ and $u \Vdash \varphi_1$ then $u \in S$. Then we have that $t \Vdash \neg\varphi_2$ for all $t \in T$, so $\vdash \varphi_T \Rightarrow \neg\varphi_2$. In addition, for each $t \in T$ and state u such that $t \to u$, we have either $u \in T$ or $u \Vdash \neg\varphi_1 \wedge \neg\varphi_2$. Thus, using Lemma 4.3, we obtain $\vdash \varphi_T \Rightarrow \bigcirc(\varphi_T \vee (\neg\varphi_1 \wedge \neg\varphi_2))$. It now follows using Lemma 4.4 that $\vdash \varphi_T \Rightarrow \neg(\varphi_1 \,\mathcal{U}\, \varphi_2)$. In particular, since $s \in T$, we have $s \Vdash \neg(\varphi_1 \,\mathcal{U}\, \varphi_2)$, which contradicts $s \Vdash \varphi_1 \,\mathcal{U}\, \varphi_2$. □

We now establish a lemma which concerns the interaction of knowledge and time in the pre-model. This result will enable us to satisfy the perfect recall requirement in using the pre-model to construct an interpreted system. It is convenient to introduce the notation $[s]_i$, where s is a state, for the set of states t such that $s \approx_i t$. The reader is encouraged to compare the following result with condition (b) of Proposition 2.1.

Lemma 4.6: For all σ-states s, t and for all $(\sigma\#i)$-states t', if $s \to t$ and $t \approx_i t'$ then either $s \approx_i t'$ or there exists a $(\sigma\#i)$-state s' such that $s \approx_i s'$ and there exists a sequence of $(\sigma\#i)$-states $u_0 \to u_1 \to \ldots \to u_n = t'$, where $n \geq 0$, such that $s' \to u_0$ and $u_l \approx_i u_{l+1}$ for all $l = 0 \ldots n-1$.

Proof: We derive a contradiction from the assumption that $s \to t$ and $t \approx_i t'$, but $s \not\approx_i t'$ and for all $(\sigma\#i)$-states s' such that $s \approx_i s'$ and all sequences of $(\sigma\#i)$-states $u_0 \to u_1 \to \ldots \to u_n$ such that $s' \to u_0$ and $u_i \approx_i u_{i+1}$ for $i = 0 \ldots n-1$, we have $u_n \neq t'$. Define T to be the smallest set of $(\sigma\#i)$-states such that

1. if $v \in [s]_i$ and $v \to v'$ and $v' \in [t]_i$ then $v' \in T$, and

2. if $v \in T$ and $v \to v'$ and $v' \in [t]_i$ then $v' \in T$.

Because $s \not\approx_i t'$, it follows using the fact that the relation $\approx_i$ is transitive and euclidean that the intersection $[s]_i \cap [t]_i$ is empty. Additionally, t' is not in T, for otherwise we could find a sequence of the sort presumed not to exist.

For all $v \in T$ we have $v \neq t'$, which means that $\vdash \varphi_v \Rightarrow \neg\varphi_{t'}$. This implies that $\vdash \varphi_T \Rightarrow \neg\varphi_{t'}$. Next, let T' be the set of $(\sigma\#i)$-states v' such that $v \rightarrow v'$ for some $v \in T$. For $v' \in T'$ there are two possibilities. First, if $v' \in T$ then clearly we have $v' \Vdash \varphi_T$. Alternately, suppose that $v' \notin T$. By the second condition of the definition of T this means that v' is not in $[t]_i$. It follows using Lemma 4.2(4) that $v' \Vdash \neg K_i\Phi_{t,i}$. Further, $t \not\approx_i v'$ implies that $v' \neq t'$, so $v' \Vdash \neg\varphi_{t'}$. Thus, in either case, we have $v' \Vdash \varphi_T \vee (\neg K_i\Phi_{t,i} \wedge \neg\varphi_{t'})$. This shows that $\vdash \varphi_{T'} \Rightarrow \varphi_T \vee (\neg K_i\Phi_{t,i} \wedge \neg\varphi_{t'})$. Now by Lemma 4.3, we have $\vdash \varphi_T \Rightarrow \bigcirc\varphi_{T'}$, so using T1 and RT1 we obtain that $\vdash \varphi_T \Rightarrow \bigcirc(\varphi_T \vee (\neg K_i\Phi_{t,i} \wedge \neg\varphi_{t'}))$. Combining this with $\vdash \varphi_T \Rightarrow \neg\varphi_{t'}$ and using Lemma 4.4 establishes that $\vdash \varphi_T \Rightarrow \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'})$. In particular, we obtain $v \Vdash \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'})$ for all states v in T.

We now repeat this argument to obtain a similar conclusion for the elements of $[s]_i$. Since t' is not in $[s]_i$ we have that $v \in [s]_i$ implies $v \Vdash \neg\varphi_{t'}$. Further, since $[s]_i \cap [t]_i$ is empty we also have by Lemma 4.2(4) that $v \in [s]_i$ implies $v \Vdash \neg K_i\Phi_{t,i}$. Using T3 this yields that $\vdash \Phi_{s,i} \Rightarrow \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'})$.

Let R be the set of $(\sigma\#i)$-states v' such that $v \rightarrow v'$ for some $v \in [s]_i$. There are two possibilities for $v' \in R$. If $v' \in [s]_i$ then clearly $v' \Vdash \Phi_{s,i}$. Alternately, suppose v' is not in $[s]_i$. By Lemma 4.2(4) we see that $v' \Vdash \neg K_i\Phi_{s,i}$. We now consider two subcases: (a) $v' \in T$ and (b) $v' \notin T$. If $v' \in T$ then, as was shown above, we have $v' \Vdash \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'})$. If v' is not in T, then by the definition of T it follows that $t \not\approx_i v'$. By Lemma 4.2(4) this implies that $v' \Vdash \neg K_i\Phi_{t,i}$. Further, since $t \approx_i t'$ we also obtain that $v' \neq t'$, so $v' \Vdash \neg\varphi_{t'}$. Using T3, this yields that $v' \Vdash \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'})$. Thus, in every case we have that $v' \in R$ implies that $v' \Vdash \Phi_{s,i} \vee (\neg K_i\Phi_{s,i} \wedge \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'}))]$. Using Lemma 4.3 we obtain that $\vdash \Phi_{s,i} \Rightarrow \bigcirc[\Phi_{s,i} \vee (\neg K_i\Phi_{s,i} \wedge \neg(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'}))]$. Applying Lemma 4.4 to this and the result of the previous paragraph establishes that $\vdash \Phi_{s,i} \Rightarrow \neg(K_i\Phi_{s,i}\,\mathcal{U}\,(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'}))$.

It follows using Lemma 4.2(2), RK and K1 that $s \Vdash K_i\neg(K_i\Phi_{s,i}\,\mathcal{U}\,(K_i\Phi_{t,i}\,\mathcal{U}\,\varphi_{t'}))$. By KT, we obtain $s \Vdash \neg(K_i\Phi_{s,i} \wedge \bigcirc(K_i\Phi_{t,i} \wedge \neg K_i\neg\varphi_{t'}))$. Since by Lemma 4.2(2), $s \Vdash K_i\Phi_{s,i}$, we obtain using T2 that $s \Vdash \bigcirc\neg(K_i\Phi_{t,i} \wedge \neg K_i\neg\varphi_{t'}))$. Because $s \rightarrow t$ we have that $\varphi_s \wedge \bigcirc\varphi_t$ is consistent, so it follows that $\varphi_t \wedge \neg(K_i\Phi_{t,i} \wedge \neg K_i\neg\varphi_{t'})$ is consistent. But by Lemma 4.2, $t \Vdash K_i\Phi_{t,i} \wedge \neg K_i\neg\varphi_{t'}$, so this is a contradiction. $\Box$

Say that an infinite sequence of states $A = (s_0, s_1, \ldots)$, where $s_n = (\sigma, U_n)$ for all n, is *acceptable* if

1. $s_n \rightarrow s_{n+1}$ for all $n \geq 0$, and
2. for all $n \geq 0$, if $\varphi_1\,\mathcal{U}\,\varphi_2 \in U_n$ then there exists an $m \geq n$ such that $s_m \Vdash \varphi_2$ and $s_k \Vdash \varphi_1$ for all k with $n \leq k < m$.

Every finite sequence of states $s_0 \rightarrow s_1 \rightarrow \ldots \rightarrow s_n$, may be extended to an infinite acceptable sequence. To see this, notice first that for every σ-state s there exists a state t with $s \rightarrow t$. For, otherwise $s \Vdash \neg\bigcirc\varphi_t$ for all σ-states t, which contradicts $\vdash \bigcirc\bigvee\{\varphi_t \mid t \text{ a } \sigma\text{-state}\}$. Thus every finite sequence of states may be extended to an infinite sequence, and it remains to show that the obligations arising from the second condition can be satisfied.

Now, for any formula $\varphi_1 \mathcal{U} \varphi_2 \in U_0$, it follows using T3 and the fact that the s_i directly decide each of the formulae φ_1, φ_2 and $\varphi_1 \mathcal{U} \varphi_2$ that either the obligation imposed by $\varphi_1 \mathcal{U} \varphi_2$ at s_0 with is already satisfied in the sequence $(s_0, \ldots, s_n)$, or else $s_n \Vdash \varphi_1 \mathcal{U} \varphi_2$ and $s_k \Vdash \varphi_1$ for $0 \leq k \leq n$. In the later case, by Lemma 4.5(3) there exists a sequence $s_n \rightarrow s_{n+1} \rightarrow \ldots \rightarrow s_m$ such that $s_m \Vdash \varphi_2$ and $s_k \Vdash \varphi_1$ for $n \leq k < m$. This gives a finite extension of the original sequence which satisfies the obligation imposed by $\varphi_1 \mathcal{U} \varphi_2$ at s_0. Applying this argument to the remaining obligations at s_0, we eventually obtain a finite sequence which satisfies all the obligations at s_0. We may then move on to s_1 and apply the same procedure. It is clear that in the limit we obtain an acceptable sequence extending the original sequence.

We now construct an interpreted system $\mathcal{I} = (\mathcal{R}, \pi)$ as follows. The set of runs $\mathcal{R}$ will correspond to the set of acceptable sequences. For each primitive proposition P and run $r = (s_0, s_1, \ldots)$ we define $\pi(r,n)(P) = true$ if and only if $s_n \Vdash P$. When $s = (\sigma, U)$ is a state and i is a processor, define $O_i(s)$ to be the pair $(\sigma\#i, U/K_i)$. Observe that if $s \approx_i t$ then $O_i(s) = O_i(t)$. We define the local state of processor i at a point (r,n) in $\mathcal{I}$ inductively by $r_i(0) = O_i(s_0)$ and $r_i(n+1) = r_i(n)\#O_i(s_{n+1})$. The state of the environment may be defined arbitrarily. By construction, $\mathcal{I}$ is a system with perfect recall.

Lemma 4.7: Suppose s and t are states with $s \approx_i t$. Then for all runs $r = (s_0, s_1, \ldots)$ and numbers n such that $s_n = s$ there exists a run $r' = (t_0, t_1, \ldots)$ and a number m such that $(r,n) \sim_i (r',m)$ and $t_m = t$.

Proof: By induction on n. The result for $n = 0$ is immediate, since we may take r' to be any run with $t_0 = t$. Such a run will exist by the discussion above. Suppose that the result holds for some n, and let $s_{n+1} = s$. Because $s_n \rightarrow s_{n+1}$ and $s_{n+1} \approx_i t$, it follows by Lemma 4.6 that either (i) $s_n \approx_i t$ or (ii) there exists a state s' such that $s_n \approx_i s'$ and there exists a sequence of states $u_0 \rightarrow u_1 \rightarrow \ldots \rightarrow u_k = t$ such that $s' \rightarrow u_0$ and $u_l \approx_i u_{l+1}$ for all $l = 0 \ldots k-1$. We will show that in either case a point (r', m) of the required sort exists.

In case (i), by the induction hypothesis there exists a run $r' = (t_0, t_1, \ldots)$ and a number m such that $(r,n) \sim_i (r',m)$ and $t_m = t$. Note that $s_n \approx_i t$ and $s_{n+1} \approx_i t$ implies $O_i(s_n) = O_i(t) = O_i(s_{n+1})$, so $r_i(n+1) = r_i(n) = r'_i(m)$. This means that (r', m) satisfies the condition required.

In case (ii), by the induction hypothesis there exists a run $r'' = (v_0, v_1, \ldots)$ and a number m such that $(r,n) \sim_i (r'',m)$ and $v_m = s'$. Define r' to be any run with initial segment $(v_0, \ldots, v_m, u_0, \ldots, u_k)$. Because $s_{n+1} \approx_i t = u_k$ and $u_l \approx_i u_{l+1}$ for all $l = 0 \ldots k-1$ we have $O_i(u_l) = O_i(s_{n+1})$ for all $l = 0 \ldots k$. Thus $r_i(n+1) = r_i(n)\#O_i(s_{n+1}) = r''_i(m)\#O_i(t) = r'_i(m+k+2)$. Since the $m+k+2$-nd state of r'' is $u_k = t$ the point $(r'', m+k+2)$ satisfies the condition we require. □

If σ is an index, define a formula φ to be *σ-compatible* when either

(a) $ad(\varphi) \leq d - |\sigma|$, or

(b) σ is of the form τi and $ad(\varphi) = d - |\sigma| + 1$ and any subformula $K_j\varphi'$ of φ with $j \neq i$ has alternation depth strictly less than $d - |\sigma| + 1$.

Clearly any subformula of a σ-compatible formula is also σ-compatible.

Lemma 4.8: If σ and τ are indices satisfying $\sigma\#i = \tau\#i$ then a formula of the form $K_i\varphi$ is σ-compatible if and only if it is τ-compatible.

Proof: If $\sigma\#i = \tau\#i$ then either $\sigma = \tau$ or $\sigma = \tau i$ or $\tau = \sigma i$. In the first case there is nothing to prove. We establish the result when $\sigma = \tau i$. The case $\tau = \sigma i$ follows by symmetry. Assume $\sigma = \tau i$. Note first that this implies that every formula which is σ-compatible is τ-compatible. For, if φ' is σ-compatible then $ad(\varphi') \leq d - |\sigma| + 1 = d - |\tau|$. Thus φ' is τ-compatible by condition (a). Therefore, it remains to prove that if $K_i\varphi$ is τ-compatible then $K_i\varphi$ is σ-compatible. Suppose $K_i\varphi$ is τ-compatible. Because $\sigma = \tau i$, the index τ is not of the form $\tau' i$, so we must have $ad(K_i\varphi) \leq d - |\tau| = d - |\sigma| + 1$. If $ad(K_i\varphi) \leq d - |\sigma|$ then $K_i\varphi$ is σ-compatible by condition (a). This leaves the case where $ad(K_i\varphi) = d - |\sigma| + 1$. Now any subformula of $K_i\varphi$ of the form $K_j\varphi'$ with $j \neq i$ must have alternation depth strictly less than $d - |\sigma| + 1$, for otherwise the alternation depth of $K_i\varphi$ would be greater than $d - |\sigma| + 1$. It follows that $K_i\varphi$ is σ-compatible by condition (b). $\square$

The following lemma gives a sufficient condition for a formula in the basic closure to hold at a point in the system $\mathcal{I}$.

Lemma 4.9: Let φ be a σ-compatible formula in the basic closure $cl_0(\psi)$, and let $r = (s_0, s_1, \ldots)$ be a run consisting of σ-states. Then for all $n \geq 0$, we have $\mathcal{I}, (r, n) \models \varphi$ if and only if $s_n \Vdash \varphi$.

Proof: By induction on the complexity of φ. If φ is a propositional constant then the result is immediate from the definition of $\mathcal{I}$. The cases where φ is of the form $\neg\varphi_1$ or $\varphi_1 \wedge \varphi_2$ are similarly trivial. This leaves three cases:

Case 1: Suppose that φ is of the form $\bigcirc\varphi_1$. Then $\mathcal{I}, (r, n) \models \varphi$ if and only if $\mathcal{I}, (r, n+1) \models \varphi_1$. Since φ_1 is a subformula of φ it is σ-compatible, so it follows by the induction hypothesis that $\mathcal{I}, (r, n+1) \models \varphi_1$ holds precisely when $s_{n+1} \Vdash \varphi_1$. Now since $s_n \rightarrow s_{n+1}$, we obtain from Lemma 4.5(1) that $s_{n+1} \Vdash \varphi_1$ if and only if $s_n \Vdash \bigcirc\varphi_1$. Putting the pieces together, we get $\mathcal{I}, (r, n) \models \varphi$ if and only if $s_n \Vdash \varphi$.

Case 2: Suppose that φ is a σ-compatible formula of the form $\varphi_1 \,\mathcal{U}\, \varphi_2$. Then the subformulae φ_1 and φ_2 are also σ-compatible. Since r is an admissible sequence, if $s_n \Vdash \varphi_1 \,\mathcal{U}\, \varphi_2$ then there exists some $m \geq n$ such that $s_m \Vdash \varphi_2$ and $s_k \Vdash \varphi_1$ for $n \leq k < m$. By the induction hypothesis, this implies that $\mathcal{I}, (r, m) \models \varphi_2$ and $\mathcal{I}, (r, k) \models \varphi_1$ for $n \leq k < m$. In other words, we have $\mathcal{I}, (r, n) \models \varphi_1 \,\mathcal{U}\, \varphi_2$. Conversely, if $\mathcal{I}, (r, n) \models \varphi_1 \,\mathcal{U}\, \varphi_2$ then by the induction hypothesis and the semantics of $\mathcal{U}$ we have that there exists some $m \geq n$ such that $s_m \Vdash \varphi_2$ and $s_k \Vdash \varphi_1$ for $n \leq k < m$. Since $s_n \rightarrow s_{n+1} \rightarrow \ldots \rightarrow s_m$, it follows using Lemma 4.5(3) that $s_n \Vdash \varphi_1 \,\mathcal{U}\, \varphi_2$.

Case 3: Suppose that φ is of the form $K_i\varphi_1$. We first show that $s_n \Vdash K_i\varphi_1$ implies $\mathcal{I}, (r,n) \models K_i\varphi_1$. Assume $s_n \Vdash K_i\varphi_1$ and suppose $r' = (t_0, t_1, \ldots)$ and m are such that $(r,n) \sim_i (r',m)$. Then $O_i(s_n) = O_i(t_m)$, so since $K_i\varphi \in cl_0(\psi)$ we obtain $t_m \Vdash K_i\varphi_1$. By K2 this implies $t_m \Vdash \varphi_1$. Further, we see that r' is a sequence of τ-states for some τ with $\sigma\#i = \tau\#i$. It follows by Lemma 4.8 that φ_1 is τ-compatible. By the induction hypothesis, we obtain that $\mathcal{I}, (r',m) \models \varphi_1$. This shows that $\mathcal{I}, (r',m) \models \varphi_1$ for all points $(r',m) \sim_i (r,n)$. That is, we have $\mathcal{I}, (r,n) \models K_i\varphi_1$.

Conversely, we suppose $s_n \Vdash \neg K_i\varphi_1$ and show $\mathcal{I}, (r,n) \models \neg K_i\varphi_1$. By Lemma 4.5(2) there exists a $(\sigma\#i)$-state t such that $s_n \approx_i t$ and $t \Vdash \neg\varphi_1$. By Lemma 4.7, there exists a run $r' = (t_0, t_1, \ldots)$ and a number m such that $(r,n) \sim_i (r',m)$ and $t_m = t$. By Lemma 4.8, the formula φ_1 is $(\sigma\#i)$-compatible. Using the induction hypothesis we obtain that $\mathcal{I}, (r',m) \models \neg\varphi_1$. It follows that $\mathcal{I}, (r,n) \models \neg K_i\varphi_1$. $\square$

The completeness result now follows directly. Since ψ is consistent there exists a state $s = (\epsilon, U)$ with $\psi \in U$, hence $s \Vdash \psi$. Let r be any run $(s_0, s_1, \ldots)$ with $s_0 = s$. Since $ad(\psi) = d = d - |\epsilon|$ the formula ψ is ϵ-compatible. By definition of the basic closure, ψ is in $cl_0(\psi)$. Using Lemma 4.9 we obtain that $\mathcal{I}, (r, 0) \models \psi$.

5 Incompleteness of KT'

We now show that the conjecture that $S5_n^U + KT'$ is complete for the class of systems with perfect recall is false. Before doing so, we note that this conjecture was reasonable. Consider repeating the argument of our completeness proof for the logic $S5_n^U + KT'$. It is possible to use axiom KT' to establish a result corresponding to Lemma 4.6, stating that

> for all states s, t and u such that $s \rightarrow t$ and $t \approx_i u$, there exists a sequence of states $v_0 \rightarrow v_1 \rightarrow \ldots \rightarrow v_n = u$ such that $s \approx_i v_0$. (*)

This resembles condition (a) of Proposition 2.1 in the same way that Lemma 4.6 resembles condition (b) of Proposition 2.1. In the light of Proposition 2.1 this might lead one to expect that the remainder of the proof can be carried through. However, condition (*) turns out to be too weak to support the proof of any result analogous to Lemma 4.7, the problem being that we might have both $v_1 \not\approx_i v_0$ and $v_1 \not\approx_i v_n$ in the above sequence.

Indeed, KT is not a theorem of $S5_n^U + KT'$, even in the single processor case. To show this, let us consider modifying the definition of interpreted system by taking runs to be functions from the integers $\mathbf{Z}$ to $L \times E$, instead of functions from the natural numbers $\mathbf{N}$ to $L \times E$. In this generalized class of systems, all the axioms of $S5_n^U$ are still valid. Further, it is not hard to show that the axiom KT' is also valid if the condition (a) of Proposition 2.1 holds, where the quantifiers are generalized to range over the integers. Consider the generalized system $\mathcal{I}$ consisting of two runs r and s over the integers, in which the propositions p and q are true at all points, with the exceptions $\mathcal{I},(s,1) \models q \wedge \neg p$ and $\mathcal{I},(s,2) \models p \wedge \neg q$. We define the knowledge access relation $\sim_1$ by stating its equivalence classes of points, which are singletons except for the following:

$$\begin{array}{l}\{(r,2),(s,2)\}\\ \{(s,n) \mid n<2 \text{ is odd}\} \cup \{(r,n) \mid n<2 \text{ is even}\}\\ \{(r,n) \mid n<2 \text{ is odd}\} \cup \{(s,n) \mid n<2 \text{ is even}\}\end{array}$$

This structure satisfies condition (a) of Proposition 2.1, so axiom KT' holds at all points. However, $\mathcal{I},(r,1) \models K_1 p \wedge \bigcirc [K_1 p \wedge \neg K_1 q] \wedge K_1 \neg (K_1 p \,\mathcal{U}\, \neg q)$, showing that KT is false at the point $(r,1)$, with $\varphi_1 = \varphi_2 = p$ and $\varphi_3 = q$.

This means that KT is not a theorem of $\mathrm{S5}_n^U + \mathrm{KT}'$, so this logic is not complete for systems with perfect recall. Note that the proof of the inference from (a) to (c) in Proposition 2.1 breaks down when we consider runs over the integers, since the induction no longer has a base case. Indeed, as the example above shows, the inference from (a) to (b) is also false under this generalization. Condition (b) still seems to give a reasonable characterization of perfect recall over backwards infinite runs, although it is far from clear how any such system could ever be implemented.

Nevertheless, the fact conditions (a), (b) and (c) are equivalent for runs over the natural numbers suggests that the axiom KT' may be sufficient to give completeness if the language is extended in such a way as to be able to express the existence of an initial point in every run, e.g. by adding past temporal operators. It would be interesting to study such an extension.

6 Conclusion

Halpern and Vardi [13] have already shown the complexity of the logic of this paper to be complete for non-elementary time. Furthermore, they have shown that the addition of operators for common knowledge makes validity Π_1^1-complete, which implies the impossibility of a recursive axiomatization of this extension. Since the relation $\rightarrow$ is defined proof theoretically, our completeness proof does not directly entail Halpern and Vardi's decidability result. However, it appears that our methods can be modified to give an alternative to their decidability proof, by showing satisfiability of a formula to be equivalent to the existence of a finite structure resembling the pre-model, which is required to satisfy the conditions of Lemma 4.5 and Lemma 4.6.

The methods we have developed in this paper may also be used to give a new and arguably simpler proof of Halpern and Vardi's [14] completeness result for the logic of knowledge and time in synchronous systems with perfect recall. Their axiomatization consists of $\mathrm{S5}_n^U$ plus the schema

$$K_i \bigcirc \varphi \Rightarrow \bigcirc K_i \varphi$$

introduced by Lehman [17]. This axiom may be used to establish a result corresponding to Lemma 4.6, stating that for all states s, t and u such that $s \rightarrow t$ and

$t \approx_i u$, there exists a state v such that $v \to u$ and $s \approx_i v$. This leads to an analogue of Lemma 4.7, and the remainder of the proof goes through as before, provided we define the local state of a processor by standard concatenation rather than absorptive concatenation of the instantaneous states $O_i(s)$.

Finally, we remark that the completeness result we have established refers to a class of systems which includes both synchronous and asynchronous systems. That is, processors are permitted to share parts of their local states in the class of systems we consider. By including a proposition which alternates at successive points, and whose present value is always known to all processors, it is possible to force this class of systems to simulate synchronous systems. It is not immediately apparent how to obtain completeness results for smaller classes of systems, such as asynchronous message passing systems, in which sharing of local state is prohibited. Our results do not apply to this case. Indeed, we can show that additional axioms are needed for reasoning about knowledge and time in asynchronous message passing systems.

Acknowledgement: Thanks to Hirofumi Katsuno for his comments on an earlier draft of this paper.

References

[1] K.M. Chandy and J. Misra. How processes learn. *Distributed Computing*, 1(1):40–52, 1986.

[2] C. Dwork and Y. Moses. Knowledge and common knowledge in a byzantine environment. *Information and Computation*, 88(2):156–186, 1990.

[3] A. E. Emerson and J. Halpern. Decision procedures and expressiveness in the modal logic of branching time. *Journal of Computer and System Sciences*, 30(1):1–24, 1985.

[4] R. Fagin, J. Halpern, Y. Moses, and M. Vardi. *Reasoning about knowledge*. draft, 1993.

[5] R. Fagin, J. Halpern, and M. Y. Vardi. What can machines know? On the epistemic properties of machines. *Journal of the ACM*, 39(2):382–428, 1992.

[6] D. Gabbay, A. Pnueli, S. Shelah, and J. Stavi. On the temporal analysis of fairness. In *Proceedings of the 7th ACM Symposium on Principles of Programming Languages*, pages 163–173, 1980.

[7] V. Hadzilacos. A knowledge theoretic analysis of atomic commitment protocols. In *Proceedings of the 6th ACM SIGACT-SIGMOD-SIGART Conference on Principles of Database Systems*, pages 129–134, New York, 1987. ACM.

[8] J. Halpern. Using reasoning about knowledge to analyze distributed systems. In J.F. Traub, B. J. Grosz, B. W. Lampson, and N. J. Nilsson, editors, *Annual Review of Computer Science*. Annual Reviews Inc., Palo Alto, CA, 1987.

[9] J. Halpern and R. Fagin. Modelling knowledge and action in distributed systems. *Distributed Computing*, 3(4):159–179, 1989.

[10] J. Halpern and Y. Moses. Knowledge and common knowledge in a distributed environment. *Journal of the ACM*, 37(3):549–587, 1990.

[11] J. Halpern and Y. Moses. A guide to completeness and complexity for modal logics of knowledge and belief. *Artificial Intelligence*, 54:319–379, 1992.

[12] J. Halpern and M. Vardi. The complexity of reasoning about knowledge and time: extended abstract. In *Proceedings of the 18th Annual ACM Symposium on Theory of Computing*, pages 304–315, 1986.

[13] J. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time in asynchronous systems. In *Proceedings of the 20th ACM Symposium on Theory of Computing*, pages 53–65, 1988.

[14] J. Halpern and M. Y. Vardi. The complexity of reasoning about knowledge and time: Synchronous systems. Technical Report RJ 6097, IBM Almaden Research Center, 1988.

[15] J. Halpern and M.Y. Vardi. The complexity of reasoning about knowledge and time, I: Lower bounds. *Journal of Computer and System Sciences*, 38(1):195–237, 1989.

[16] R. Ladner and J. Reif. The logic of distributed protocols. In J.Y. Halpern, editor, *Theoretical Aspects of Reasoning about Knowledge: Proceedings of the 1986 Conference*, pages 207–222, San Mateo, California, 1986. Morgan Kaufmann.

[17] D. J. Lehman. Knowledge, common knowledge and related puzzles. In *Proceedings of the 3rd ACM Symposium on Principles of Distributed Computing*, pages 62–67, 1984.

[18] M. Mazer. A knowledge theoretic account of recovery in distributed systems: The case of negotiated commitment. In M.Y. Vardi, editor, *Proceedings of the 2nd Conference on Theoretical Aspects of Reasoning about Knowledge*, pages 207–222, San Mateo, California, 1986. Morgan Kaufmann.

[19] R. Parikh and R. Ramanujam. Distributed processing and the logic of knowledge. In *Proceedings, Workshop on Logics of Programs*, pages 256–268. Springer LNCS No. 193, 1985.

[20] M. Sato. A study of Kripke-style methods for some modal logics by Gentzen's sequential method. *Publications of the Research Institute for Mathematical Sciences, Kyoto University*, 13(2), 1977.