Vassar College
Digital Window @ Vassar

Faculty Research and Reports

Winter 2-2-2018

Armstrong's Axioms and Navigation Strategies

Kaya Deuser
kdeuser@vassar.edu

Pavel Naumov
pnaumov@vassar.edu

Follow this and additional works at: https://digitalwindow.vassar.edu/faculty_research_reports

Part of the Artificial Intelligence and Robotics Commons, Logic and Foundations Commons, and the Theory and Algorithms Commons

Citation Information

Armstrong's Axioms and Navigation Strategies

Kaya Deuser and **Pavel Naumov**

Vassar College
124 Raymond Avenue
Poughkeepsie, NY 12604
{kdeuser, pnaumov}@vassar.edu

Abstract

The paper investigates navigability with imperfect information. It shows that the properties of navigability with perfect recall are exactly those captured by Armstrong's axioms from database theory. If the assumption of perfect recall is omitted, then Armstrong's transitivity axiom is not valid, but it can be replaced by a weaker principle. The main technical results are soundness and completeness theorems for the logical systems describing properties of navigability with and without perfect recall.

Introduction

Navigation is a commonly encountered task by autonomous agents that need to reach a destination or, more generally, to find a solution to a problem, where the solution is a sequence of instructions that transition a system from one state to another. This task is often performed when the agent does not have precise information about her current location. Examples of such agents are self-navigating missiles, self-driving cars, and robotic vacuum cleaners.

Figure 1 depicts an example T_0 of a transition system. This system consists of eight states $a, \ldots, h$ represented by the vertices of the graph. The agent cannot distinguish state a from state b and state c from state d, which is denoted by dashed lines connecting the indistinguishable states. The directed edges of the graph represent transitions that the system can make and the labels on these edges represent the instructions that the agent must give to do this. For example, if in state a the agent executes instruction 0, then the system transitions into state g, if, instead, the agent executes instruction 1, the system transitions into state e. Although in this paper we consider non-deterministic transition systems where the execution of the same instruction can transition the system into one of the several states, for the sake of simplicity the transition system T_0 is deterministic.

Note that, in system T_0 the agent can navigate from state a to state c by using instruction 1 in state a and the same instruction 1 again in state e. However, a different sequence of instructions is required to reach state c from state b. As the agent cannot distinguish state a from state b, in state a she does not know which instructions to use to accomplish her
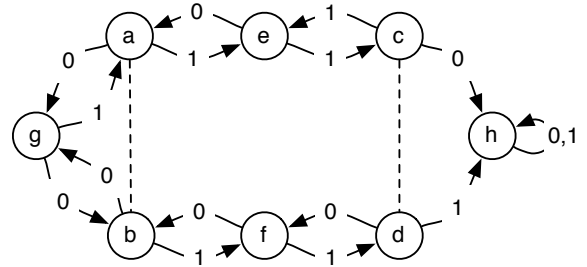
Figure 1: Transition System T_0.

goal. Moreover, if the agent does reach state c, she cannot verify that the task is completed, because she cannot distinguish state c from state d. For this reason, in this paper instead of navigation between states, we consider navigation between equivalence classes of states with respect to the indistinguishability relation. For example, the agent can navigate from class $[a] = \{a, b\}$ to class $[c] = \{c, d\}$ by using instruction 1 in each state she passes.

Perfect Recall In order to achieve a goal, the agent would need to follow a certain strategy that must be stored in her memory. We assume that the strategy is permanently stored ("hardwired") in the memory and cannot be changed during the navigation. For example, a robotic vacuum cleaner might be programmed to change direction when it encounters a wall, to make a circle when the dirt sensor is triggered, and to follow a straight path otherwise. A crucial question for us is if the vacuum cleaner can remember the walls and the dirty spots it has previously encountered. In other words, we distinguish an agent that can keep track of the classes of states she visited and the instructions she used from an agent who only knows her current state. We say that in the former case the agent has *perfect recall* and in the later she does not.

A strategy of an agent without perfect recall can only use information available to her about *the current state* to decide which instruction to use. In other words, a strategy of such an agent is a function that maps classes of indistinguishable *states* into instructions. A strategy of an agent with perfect recall can use information about *the history* of previous tran-

sitions to decide which instruction to use. In other words, a strategy of such an agent is a function that maps classes of indistinguishable *histories* into instructions. We call the former *memoryless* strategies and the later *recall* strategies.

In theory, a robotic vacuum cleaner without perfect recall is only equipped with read-only memory to store the strategy. A theoretical robotic vacuum with perfect recall in addition to read-only memory that contains the strategy also has an unlimited read-write memory that contains logs of all previous transitions. In practice, the most popular brand of robotic vacuum cleaners, Roomba, is only using read-write memory to store information, such as a cleaning schedule, that is not used for navigation. This means that Roomba is using a memoryless strategy. The other popular robotic vacuum cleaner, Neato, is scanning the room before cleaning to use this information for navigation purposes. Although, of course, Neato only has a read-write memory of a limited size, the navigation strategy used by Neato is an example of a recall strategy.

Examples Transition system T_0 has a recall strategy to navigate from class $[a]$ to class $[e]$. This might be surprising, because the same strategy must work to navigate from both state a and state b of class $[a]$ to the only state e of class $[e]$. Yet, one would expect to use instruction 1 in state a and instruction 0 in state b to get to e. The recall strategy to navigate from class $[a]$ to class $[e]$ consists in first using instruction 0 once no matter what the starting state is, and then using instruction 1 until state e is reached. When this strategy is used starting from a state in class $[a]$, the system first transitions into state g, then into state a, and then finally into state e. This is a recall strategy because it uses a different instruction in state a depending if the system has already visited a state in class $[g]$ or not.

To show that there is no memoryless strategy to navigate from class $[a]$ to class $[e]$ note that any such strategy will have to use the same instruction i_0 at every visit to states a and b. If $i_0 = 0$, then when the navigation starts in state b, the system stays "locked" in set of states $\{a, g, b\}$ and never reaches state e. Similarly, if $i_0 = 1$, then when the navigation starts in state b, the system stays "locked" in set of states $\{b, f, d, h\}$ and never reaches state e. Thus, there is no memoryless strategy to navigate from class $[a]$ to class $[e]$.

In some situation, even when there is a path between appropriate states, there might be no memoryless strategy and no recall strategy. For example, transition system T_0 has no recall strategy to navigate from class $[c]$ to class $[g]$. Indeed, if such a recall strategy exists, it would have to use the same instruction i_0 when the system *starts* in either state c or state d. Suppose $i_0 = 0$. Thus, if the system *starts* in state c it is "locked" in state h. Assume now that $i_0 = 1$. Hence, if the system *starts* in state d it is "locked" in state h. Therefore, there is no recall strategy from class $[c]$ to class $[g]$.

Table 1 shows when memoryless strategies and recall strategies between classes of states of transition system T_0 exist. In this table, letter "m" at the intersection of row x and column y denotes the existence of a memoryless strategy from class x to class y. Letter "r" denotes the existence of a recall strategy, but no memoryless strategy, and dash "-"

	$\{a,b\}$	$\{c,d\}$	$\{e\}$	$\{f\}$	$\{g\}$	$\{h\}$
$\{a,b\}$	m	m	r	r	m	r†
$\{c,d\}$	-	m	-	-	-	r†
$\{e\}$	m	m	m	r†	m	m
$\{f\}$	m	m	r†	m	m	m
$\{g\}$	m	m	m	m	m	m†
$\{h\}$	-	-	-	-	-	m

Table 1: Navigability between classes in system T_0.

denotes that neither memoryless strategy nor recall strategy exist. Symbol † marks the cases that we have found to be interesting to think about.

Navigability between Sets As we have seen, there is a recall strategy, but no memoryless strategy, to navigate from class $[a]$ to class $[e]$. One can similarly show that there is no memoryless strategy to navigate from class $[a]$ to class $[f]$. However, if the goal is to navigate from class $[a]$ to either class $[e]$ or to class $[f]$, then there is a memoryless strategy to do this. Indeed, consider a memoryless strategy that uses instruction 1 in every state. This strategy can transition the system from a state of class $[a]$ to a state of a class in set $\{[e], [f]\}$.

Thus, navigability between *sets of classes* can not be reduced to navigability between *classes*. For this reason, in this paper we study properties of navigability between sets of classes. If there is a strategy to navigate from a set of classes A to a set of classes B, then we write $A \rhd B$. It will be clear from the context if we refer to the existence of a memoryless strategy or a recall strategy.

Universal Properties of Navigability In the examples above we talked about properties of navigability for the transition system T_0. In the rest of this paper we study universal properties of navigability between sets of classes that are true in all transition systems. An example of such a property is *reflexivity*: $A \rhd B$, where $A \subseteq B$. This property is true for both memoryless and recall strategies because absolutely *any* strategy can be used to navigate from a subset to the whole set. In fact, in this case the goal is achieved before the navigation even starts.

Another example of a property of navigation which is universally true for both memoryless and recall strategies is *augmentation*: $A \rhd B \rightarrow (A \cup C) \rhd (B \cup C)$. It says that if there is a strategy to navigate from set A to set B, then there is a strategy to navigate from set $A \cup C$ to set $B \cup C$.

An example of a property which is universally true for recall strategies, but is not universally true for memoryless strategies is *transitivity*: $A \rhd B \rightarrow (B \rhd C \rightarrow A \rhd C)$. It states that if there is a strategy to navigate from set A to set B and a strategy to navigate from set B to set C, then there is a strategy to navigate from set A to set C. To see that this property is not universally true for memoryless strategies, note that, in transition system T_0, memoryless strategy that always uses instruction 0 can be used to navigate from set $\{[a]\}$ to set $\{[g]\}$ and memoryless strategy that always uses instruction 1 can be used to navigate from set $\{[g]\}$ to set $\{[e]\}$. At the same time, as we have shown earlier, there is no

memoryless strategy to navigate from set $\{[a]\}$ to set $\{[e]\}$.

In this paper we show that reflexivity, augmentation, and transitivity principles form a sound and complete logical system that describes all universal properties of navigability by recall strategies. These are the three principles known in database theory as Armstrong's axioms (Garcia-Molina, Ullman, and Widom 2009, p. 81), where they give a sound and complete axiomatization of functional dependency (Armstrong 1974). We also give a sound and complete axiomatization of universal properties of navigability by memoryless strategies. It consists of the reflexivity and augmentation principles mentioned above as well as the *monotonicity* principle $(A \cup C) \triangleright B \rightarrow A \triangleright B$. The latter principle is true for the recall strategies as well, but it is provable from Armstrong's axioms.

Literature Review   Most of the existing literature on logical systems for reasoning about strategies is focused on modal logics. Logics of coalition power were developed by (Pauly 2001; 2002), who also proved the completeness of the basic logic of coalition power. Pauly's approach has been widely studied in literature (Goranko 2001; van der Hoek and Wooldridge 2005; Borgo 2007; Sauro et al. 2006; Ågotnes et al. 2010; Ågotnes, van der Hoek, and Wooldridge 2009; Belardinelli 2014). Alternative logical system were proposed by (More and Naumov 2012), (Wang 2015; 2016), and (Li and Wang 2017). (Alur, Henzinger, and Kupferman 2002) introduced Alternating-Time Temporal Logic (ATL) that combines temporal and coalition modalities. (van der Hoek and Wooldridge 2003) proposed to combine ATL with epistemic modality to form Alternating-Time Temporal Epistemic Logic. A completeness theorem for a logical system that combines coalition power and epistemic modalities was proven by (Ågotnes and Alechina 2012).

The notion of a strategy that we consider in this paper is much more restrictive than the notion of strategy in the works mentioned above. Namely, we assume that the strategy must be based only on the information available to the agent. This is captured in our setting by requiring the strategy to be the same in all indistinguishable states or histories. This restriction on strategies has been studied before under different names. (Jamroga and Ågotnes 2007) talk about "knowledge to identify and execute a strategy", (Jamroga and van der Hoek 2004) discuss "difference between an agent knowing that he has a suitable strategy and knowing the strategy itself". (van Benthem 2001) calls such strategies "uniform". (Naumov and Tao 2017a) use the term "executable strategy". (Naumov and Tao 2017b) proposed a complete trimodal logical system describing an interplay between distributed knowledge, uniform strategic power modality, and standard strategic power modality for achieving a goal by a coalition in one step. (Fervari et al. 2017) developed a complete logical system in a single-agent setting for uniform strategies to achieve a goal in multiple steps. (Naumov and Tao 2017a) developed a similar system for maintaining a goal in multi-agent setting. Our contribution is different from all of the above papers by being the first to propose complete logical systems for recall strategies and memoryless strategies.

Paper Outline   In the next section we define transition systems and the syntax of our logical systems. This section applies equally to recall and memoryless strategies. The rest of the paper is split into two independent sections. The first of them proves the soundness and the completeness of Armstrong's axioms for navigability under recall strategies and the second gives an axiomatization for memoryless strategies.

Syntax and Semantics

In this section we formally define the language of our logical system, the notion of a transition system, and the related terminology. In the introduction, relation $\triangleright$ was viewed as a relation between equivalence classes of a given transition system. Thus, our language depends on these classes and changes from transition system to transition system. In order to have a single language for all transition systems we introduce a fixed finite set of "views" V, whose elements act as names of the equivalence classes in any given transition system.

Definition 1 Φ *is the minimal set of formulae such that*

1. $A \triangleright B \in \Phi$ *for all nonempty[1] sets $A, B \subseteq V$,*
2. $\neg \varphi, \varphi \rightarrow \psi \in \Phi$ *for all formulae $\varphi, \psi \in \Phi$.*

Each transition system specifies a mapping $*$ of views into equivalence classes of states. Transitions between states under an instruction i are captured by a transition function Δ_i.

Definition 2 $(S, \sim, *, I, \{\Delta_i\}_{i \in I})$ *is a transition system, if*

1. S *is a set of states,*
2. $\sim$ *is an equivalence (indistinguishability) relation on S,*
3. $*$ *is a function from V to $S/\sim$,*
4. I *is an arbitrary nonempty set of "instructions",*
5. Δ_i *maps set S into nonempty subsets of S for each $i \in I$.*

We write $a*$ instead of $*(a)$, where $a \in V$. An example of a transition system is system T_0 depicted in Figure 1.

Definition 3 *A finite sequence* $w_0, i_1, w_1, \ldots, i_n, w_n$, *where $n \geqslant 0$, is called a history if*

1. $w_k \in S$ *for each k such that $0 \leqslant k \leqslant n$,*
2. $i_k \in I$, *for each k such that $1 \leqslant k \leqslant n$,*
3. $w_k \in \Delta_{i_k}(w_{k-1})$, *for each k such that $1 \leqslant k \leqslant n$.*

For example, sequence $g, 1, a, 1, e, 1, c, 0, h$ is a history for system T_0. The set of all histories is denoted by H.

Definition 4 *History $h = w_0, i_1, w_1, \ldots, i_n, w_n$ is indistinguishable from history $h' = w'_0, i_1, w'_1, \ldots, i_n, w'_n$ if $w_k \sim w'_k$ for each k such that $0 \leqslant k \leqslant n$.*

For example, histories $a, 0, g$ and $b, 0, g$ are indistinguishable in transition system T_0. Indistinguishability of histories of h and h' is denoted by $h \approx h'$. The equivalence class of history h with respect to this equivalence relation is denoted by $[\![h]\!]$. Equivalence class of a state w with respect to equivalence relation $\sim$ is denoted by $[w]$.

---

[1]If one allows sets A and B to be empty, most of the proofs in this paper will remain unchanged, but both logical systems will need an additional axiom $\neg(A \triangleright \varnothing)$ for each nonempty set A.

Lemma 1 *If $w_0, i_1, w_1, \ldots, w_n \approx w'_0, i_1, w'_1, \ldots, w'_n$, then $[w_k] = [w'_k]$ for each $k \leqslant n$.* $\boxtimes$

Definition 5 *A memoryless strategy is a function from set $S/\sim$ to set I. A recall strategy maps set $H/\approx$ to set I.*

We write $s[w]$ and $s[\![h]\!]$ instead of $s([w])$ and $s([\![h]\!])$.

Definition 6 *An infinite sequence $w_0, i_1, w_1, i_2, w_2 \ldots$ is called a path under a memoryless strategy s if for each $k \geqslant 1$*

1. $w_0, i_1, w_1, i_2, w_2 \ldots, w_{k-1} \in H$,
2. $i_k = s[w_{k-1}]$.

Lemma 2 *For any history $w_0, i_1, w_1, \ldots, i_n, w_n$ and any memoryless strategy s, if $i_k = s[w_{k-1}]$ for each k such that $1 \leqslant k \leqslant n$, then there are states $w_{n+1}, w_{n+2}, \cdots$ and instructions $i_{n+1}, i_{n+2}, \ldots$ such that sequence $w_0, i_1, w_1, \ldots, i_n, w_n, i_{n+1}, w_{n+1}, i_{n+2}, w_{n+2}, \ldots$ is a path under strategy s.*

Proof. Elements $i_{n+1}, w_{n+1}, i_{n+2}, w_{n+2}, \ldots$ can be constructed recursively because (a) there is a state $w_{k+1} \in \Delta_{i_{k+1}}(w_k)$ for any state w_k and any $i_{k+1} \in I$ by item 5 of Definition 2; (b) $I \neq \varnothing$ by item 4 of Definition 2. $\boxtimes$

Definition 7 *An infinite sequence $w_0, i_1, w_1, i_2, w_2 \ldots$ is called a path under a recall strategy s if for each $k \geqslant 1$*

1. $w_0, i_1, w_1, i_2, w_2 \ldots, w_{k-1} \in H$,
2. $i_k = s[\![w_0, i_1, w_1, i_2, w_2 \ldots, w_{k-1}]\!]$.

Definition 8 $A^* = \{a^* \mid a \in A\}$, *for all sets $A \subseteq V$.*

Definition 9 *For a given memoryless strategy or recall strategy s, let $Path_s(A)$ be the set of all paths $w_0, i_1, w_1, i_2, w_2 \ldots$ under s such that $[w_0] \in A^*$.*

Definition 10 *Let set $Visit_s(B)$ be the set of all paths $w_0, i_1, w_1, i_2, w_2 \ldots$ under s such that $[w_k] \in B^*$ for some $k \geqslant 0$.*

We write $Visit(B)$ instead of $Visit_s(B)$ when value of s is clear from the context.

Navigation with Recall Strategies

In this section we show that Armstrong's axioms give a complete axiomatization of navigability between sets of classes with recall strategies. We start with a formal semantics of navigability relation $\rhd$ under recall strategies.

Definition 11 $T \vDash A \rhd B$ *if $Path_s(A) \subseteq Visit(B)$ for some recall strategy s of transition system T.*

Axioms The axioms of the logical system that we consider in this section are the tautologies in language Φ and the following additional principles known as Armstrong's axioms (Garcia-Molina, Ullman, and Widom 2009, p. 81):

1. Reflexivity: $A \rhd B$, where $A \subseteq B$,
2. Augmentation: $A \rhd B \to (A \cup C) \rhd (B \cup C)$,
3. Transitivity: $A \rhd B \to (B \rhd C \to A \rhd C)$.

We write $\vdash \varphi$ if formula φ is provable from these axioms using the Modus Ponens inference rule. We write $X \vdash \varphi$ if φ is provable using a set of additional axioms X.

Soundness In this section we prove soundness of the above axioms with respect to perfect recall semantics.

Theorem 1 *If $\vdash \varphi$, then $T \vDash \varphi$ for every system T.*

The soundness of Armstrong's axioms with respect to perfect recall semantics is established in Lemma 10, Lemma 11, and Lemma 12 below. We start with technical notions that we need to prove Lemma 12.

Definition 12 *For any history $w_0, i_1, \ldots, i_n, w_n$ and any set $B \subseteq V$, let truncation $(w_0, i_1, \ldots, i_n, w_n)|_B$ be*

1. $w_0, i_1, \ldots, i_n, w_n$, *if $n = 0$ or $[w_0] \in B^*$,*
2. $(w_1, i_2, \ldots, i_n, w_n)|_B$, *otherwise.*

For transition system T_0, if $B = \{b\}$ and $b^* = [e]$, then truncation $(g, 1, a, 1, e, 1, c, 0, h)|_B$ is history $(e, 1, c, 0, h)$.

Lemma 3 *If $h_1 \approx h_2$, then $h_1|_B \approx h_2|_B$.* $\boxtimes$

Definition 13 *For any recall strategies s_1 and s_2, any set $B \subseteq V$, and any history $h = w_0, i_1, \ldots, w_n$, let*

$$(s_1 \circ_B s_2)[\![h]\!] = \begin{cases} s_1[\![h]\!], & \text{if } \forall k \leqslant n \, ([w_k] \notin B^*), \\ s_2[\![h|_B]\!], & \text{otherwise.} \end{cases}$$

In other words, "composition" strategy $s_1 \circ_B s_2$ follows strategy s_1 until a state w is reached such that $[w] \in B^*$. Once this happens, the memory of all prior states and instructions is erased and strategy s_2 is executed as if navigation started from state w.

Lemma 4 *Operation $s_1 \circ_B s_2$ is well-defined.*

Proof. See Definition 12, Lemma 1, and Lemma 3. $\boxtimes$

Definition 14 *For any path $w_0, i_1, w_1, i_2, \ldots$ and any set $B \subseteq V$, let truncation $(w_0, i_1, w_1, i_2, \ldots)|_B$ be*

1. *path $w_0, i_1, w_1, i_2, \ldots$, if $[w_0] \in B^*$,*
2. *path $(w_1, i_2, w_2, i_3 \ldots)|_B$, otherwise.*

Lemma 5 *For any recall strategies s_1, s_2 and any sets $A, B \subseteq V$, if path $\pi \in Path_{s_1 \circ_B s_2}(A) \cap Visit(B)$, then $\pi|_B \in Path_{s_2}(B)$.* $\boxtimes$

Lemma 6 $Path_{s_1 \circ_B s_2}(A) \subseteq Visit(B)$ *if $Path_{s_1}(A) \subseteq Visit(B)$, for any recall strategies s_1, s_2 and $A, B \subseteq V$.*

Proof. Assume that there is a path $w_0, i_1, w_1, \ldots$ in $Path_{s_1 \circ_B s_2}(A)$ such that $w_0, i_1, w_1, \cdots \notin Visit(B)$. Thus, $i_k = (s_1 \circ_B s_2)[\![w_0, i_1, w_1, \ldots, w_{k-1}]\!]$ for each $k \geqslant 1$ and $[w_0] \in A^*$ by Definition 9 and Definition 7. Also, $[w_k] \notin B^*$ for each $k \geqslant 0$ by Definition 10. Hence, by Definition 13, we have $i_k = s_1[\![w_0, i_1, w_1, \ldots, w_{k-1}]\!]$ for each $k \geqslant 1$. Thus, $w_0, i_1, w_1, \cdots \in Path_{s_1}(A)$ by Definition 7 and Definition 7. Therefore, $w_0, i_1, w_1, \cdots \in Visit(B)$ by the assumption $Path_{s_1}(A) \subseteq Visit(B)$ of the lemma. $\boxtimes$

Lemma 7 *For any recall strategies s_1, s_2 and $A, B \subseteq V$, if $Path_{s_1 \circ_B s_2}(A) \subseteq Visit(B)$, then $(Path_{s_1 \circ_B s_2}(A))|_B \subseteq Path_{s_2}(B)$.*

Proof. Consider any path $\pi \in Path_{s_1 \circ_B s_2}(A)$. Then, $\pi \in Visit(B)$ because $Path_{s_1 \circ_B s_2}(A) \subseteq Visit(B)$. Hence, $\pi|_B \in Path_{s_2}(B)$ by Lemma 5. $\boxtimes$

Lemma 8 $Path_s(A) \subseteq Visit(C)$, if $(Path_s(A))|_B \subseteq Visit(C)$, for any s and any $A, B, C \subseteq V$. ⊠

Lemma 9 *For any recall strategy s and any $A, B, C \subseteq V$,*

1. $Path_s(A) \subseteq Path_s(B)$ if $A \subseteq B$,
2. $Path_s(A) \subseteq Visit(A)$,
3. $Path_s(A \cup C) = Path_s(A) \cup Path_s(C)$,
4. $Visit(B) \cup Visit(C) = Visit(B \cup C)$.

Proof. See Definition 7 and Definition 10. ⊠

The next three lemmas prove soundness of Armstrong's axioms with respect to the perfect recall semantics.

Lemma 10 $T \vDash A \rhd B$, where $A \subseteq B$.

Proof. By Definition 2, set I is nonempty, and thus it contains at least one instruction i_0. Let s be a recall strategy such that $s[\![h]\!] = i_0$ for each history h. By item 1 and item 2 of Lemma 9, $Path_s(A) \subseteq Path_s(B) \subseteq Visit(B)$. Therefore, $T \vDash A \rhd B$ by Definition 11. ⊠

Lemma 11 *If $T \vDash A \rhd B$, then $T \vDash (A \cup C) \rhd (B \cup C)$.*

Proof. By Definition 11, the assumption $T \vDash A \rhd B$ implies that there exists a recall strategy s such that $Path_s(A) \subseteq Visit(B)$. Thus, by item 3, item 2, and item 4 of Lemma 9,

$$
\begin{aligned}
Path_s(A \cup C) &= Path_s(A) \cup Path_s(C) \\
&\subseteq Visit(B) \cup Path_s(C) \\
&\subseteq Visit(B) \cup Visit(C) \\
&\subseteq Visit(B \cup C).
\end{aligned}
$$

Therefore, $T \vDash (A \cup C) \rhd (B \cup C)$ by Definition 11. ⊠

Lemma 12 *If $T \vDash A \rhd B$ and $T \vDash B \rhd C$, then $T \vDash A \rhd C$.*

Proof. By the assumption $T \vDash A \rhd B$ and Definition 11, we have $Path_{s_1}(A) \subseteq Visit(B)$ for some recall strategy s_1. Similarly, the assumption $T \vDash B \rhd C$ implies that there exists a recall strategy s_2 such that $Path_{s_2}(B) \subseteq Visit(C)$.

By Lemma 6, statement $Path_{s_1}(A) \subseteq Visit(B)$ implies $Path_{s_1 \circ_B s_2}(A) \subseteq Visit(B)$. Thus, by Lemma 7,

$$
(Path_{s_1 \circ_B s_2}(A))|_B \subseteq Path_{s_2}(B).
$$

Hence, we have $(Path_{s_1 \circ_B s_2}(A))|_B \subseteq Visit(C)$ because of $Path_{s_2}(B) \subseteq Visit(C)$. Thus, by Lemma 8, $Path_{s_1 \circ_B s_2}(A) \subseteq Visit(C)$. Therefore, $T \vDash A \rhd C$ by Definition 11. ⊠

Completeness In the rest of this section we prove the completeness of Armstrong's axioms with respect to the perfect recall semantics. We start by defining a canonical transition system $T(X) = (V \cup \{\circlearrowleft\}, =, *, I, \{\Delta_i\}_{i \in I})$ for an arbitrary maximal consistent set of formulae $X \subseteq \Phi$. The set of states of this transition system consists of a single state for each view, plus one additional state that we denote by symbol $\circlearrowleft$. Informally, the additional state is a sink or a "black hole" state from which there is no way out. State h in transition system T_0 depicted in Figure 1 is an example of a black hole state. Note that the indistinguishability relation

on the states of the canonical transition system is equality relation $=$. That is, the agent has an ability to distinguish any two different states in the system. The fact that equality is suitable as an indistinguishability relation for the canonical transition system with perfect recall is surprising. The indistinguishability relation for the canonical transition system for memoryless strategies, discussed in the next section, is different from equality.

Each view $v \in V$ is also a state in the canonical transition system. The equivalence class of state v consists of the state itself: $[v] = \{v\}$. We define v^* to be class $[v]$.

Lemma 13 $u \in A$ iff $[u] \in A^*$ for each view $u \in V$ and each set $A \subseteq V$. ⊠

Informally, if set X contains formula $A \rhd B$, then we want the canonical transition system $T(X)$ to have a recall strategy to navigate from set $A^* = \{[a] \mid a \in A\} = \{\{a\} \mid a \in A\}$ to set $B^* = \{[b] \mid b \in B\} = \{\{b\} \mid b \in B\}$. It turns out that it is sufficient to have just a single instruction that transitions the system from any state in set A to a state in set B. We denote this instruction by pair (A, B).

Definition 15 $I = \{(A, B) \mid A \rhd B \in X\}$.

Recall that assumption $A \rhd B \in X$ requires sets A and B to be nonempty due to Definition 1.

As discussed above, for any instruction (A, B) we define the nondeterministic transition function $\Delta_{(A,B)}$ to transition the system from a state in A to a state in B. If used outside of set of states A, instruction (A, B) transitions the system into black hole state $\circlearrowleft$:

Definition 16

$$
\Delta_{(A,B)}(w) = \begin{cases} B, & \text{if } w \in A, \\ \{\circlearrowleft\}, & \text{otherwise.} \end{cases}
$$

This concludes the definition of the transition system $T(X)$.

Next, for any recall strategy s and any set of states $G \subseteq V$, we define a family of sets of states $\{G_n^s\}_{n \geq 0}$. Informally, set G_n^s is the set of all states from which strategy s "draws" the system into set G after at most n transitions. For any history $h = w_0, i_1, w_1, \ldots, i_n, w_n$, by $hd(h)$ we mean the state w_n.

Definition 17 *For any recall strategy s and any nonempty set $G \subseteq V$, let chain of $G_0^s \subseteq G_1^s \subseteq \cdots \subseteq V \cup \{\circlearrowleft\}$ be defined as*

1. $G_0^s = G$,
2. $G_{n+1}^s = G_n^s \cup \{hd(h) \mid h \in H \text{ and } \Delta_{s[\![h]\!]}(hd(h)) \subseteq G_n^s\}$ for all $n \geq 0$.

Note that this definition, in essence, has an existential quantifier over history h. Thus, informally, strategy s is allowed to "manipulate" the history in order to "draw" the system into set G.

Lemma 14 $\circlearrowleft \notin G_n^s$ for each $n \geq 0$.

Proof. We prove this statement by contradiction. Let n be the smallest non-negative integer number such that $\circlearrowleft \in G_n^s$. Note that $n \neq 0$ because $G_0^s = G \subseteq V$ by Definition 17. Thus, $n > 0$. Hence, by Definition 17, there exists a history h such that $hd(h) = \circlearrowleft$ and $\Delta_{s[\![h]\!]}(\circlearrowleft) \subseteq G_{n-1}^s$. Note that

$\Delta_{s[\![h]\!]}(\circlearrowleft) = \{\circlearrowleft\}$ by Definition 16. Therefore, $\circlearrowleft \in G^s_{n-1}$, which contradicts the choice of integer n. $\boxtimes$

Definition 18 $G^s_\infty = \bigcup_{k \geqslant 0} G^s_k$.

We now prove properties of the family of sets $\{G^s_n\}_{n \geqslant 0}$ that are needed to finish the proof of the completeness.

Lemma 15 $X \vdash \{g\} \rhd G^s_{n-1}$ *for each integer* $n \geqslant 1$ *and each state* $g \in G^s_n$.

Proof. By Definition 17, assumptions $n \geqslant 1$ and $g \in G^s_n$ imply that there is a history h such that $hd(h) = g$ and

$$\Delta_{s[\![h]\!]}(g) \subseteq G^s_{n-1}. \qquad (1)$$

Furthermore, by Definition 15, there are nonempty sets $A, B \subseteq V$ such that $s[\![h]\!] = (A, B)$.
Case I: $g \notin A$. Then, $\Delta_{s[\![h]\!]}(g) = \{\circlearrowleft\}$ by Definition 16. Hence, $\{\circlearrowleft\} \subseteq G^s_{n-1}$ by equation (1), which contradicts Lemma 14.
Case II: $g \in A$. Then, $\vdash \{g\} \rhd A$ by the Reflexivity axiom. At the same time, $X \vdash A \rhd B$ by Definition 15. Thus, by the Transitivity axiom, $X \vdash \{g\} \rhd B$.

Assumption $g \in A$ implies $\Delta_{s[\![h]\!]}(g) = B$, by Definition 16. Thus, $B \subseteq G^s_{n-1}$ due to equation (1). Hence, $\vdash B \rhd G^s_{n-1}$ by the Reflexivity axiom.

Finally, statements $X \vdash \{g\} \rhd B$ and $\vdash B \rhd G^s_{n-1}$ by the Transitivity axiom imply, using Modus Ponens inference rule twice, that $X \vdash \{g\} \rhd G^s_{n-1}$. $\boxtimes$

Lemma 16 *For any* $n \geqslant 1$ *and any views* $a_1, \ldots, a_n \in V$, *if* $X \vdash \{a_k\} \rhd B$ *for each* $k \leqslant n$, *then* $X \vdash \{a_1, \ldots, a_n\} \rhd B$.

Proof. We prove this statement by induction on n. In the base case, $X \vdash \{a_1\} \rhd B$ due to the assumption of the lemma.

By the induction hypothesis, $X \vdash \{a_1, \ldots, a_{n-1}\} \rhd B$. Thus, by the Augmentation axiom,

$$X \vdash \{a_1, \ldots, a_{n-1}, a_n\} \rhd B \cup \{a_n\}. \qquad (2)$$

At the same time, $X \vdash \{a_n\} \rhd B$ by the assumption of the lemma. Hence, by the Augmentation axiom $X \vdash B \cup \{a_n\} \rhd B$. Thus, $X \vdash \{a_1, \ldots, a_{n-1}, a_n\} \rhd B$ by statement (2) and the Transitivity axiom. $\boxtimes$

Lemma 17 $X \vdash G^s_n \rhd G^s_{n-1}$ *for each* $n \geqslant 1$.

Proof. The statement of the lemma follows from Lemma 15 and Lemma 16. $\boxtimes$

Lemma 18 $X \vdash G^s_n \rhd G$ *for each* $n \geqslant 0$.

Proof. We prove this statement by induction on integer n. In the base case, due to Definition 17, it suffices to show that $\vdash G \rhd G$, which is an instance of the Reflexivity axiom.

For the induction step, note that $X \vdash G^s_n \rhd G^s_{n-1}$ by Lemma 17. At the same time, $X \vdash G^s_{n-1} \rhd G$ by the induction hypothesis. Hence, $X \vdash G^s_n \rhd G$ by the Transitivity axiom. $\boxtimes$

Lemma 19 *There is* $n \geqslant 0$, *such that* $G^s_\infty = G^s_n$.

Proof. Since $G^s_0 \subseteq G^s_1 \subseteq G^s_2 \subseteq \cdots \subseteq V \cup \{\circlearrowleft\}$ and set V is finite, there must exist an integer n such that $G^s_n = \bigcup_{k \geqslant 0} G^s_k$. Therefore, $G^s_n = G^s_\infty$ by Definition 18. $\boxtimes$

Lemma 20 *Set* $(\Delta_{s[\![h]\!]}(hd(h))) \backslash G^s_\infty$ *is non-empty for each history* h *such that* $hd(h) \notin G^s_\infty$.

Proof. Suppose $\Delta_{s[\![h]\!]}(hd(h)) \subseteq G^s_\infty$ for some history h. It suffices to show that $hd(h) \in G^s_\infty$. Indeed, by Lemma 19 there is n such that $G^s_\infty = G^s_n$. Thus, $\Delta_{s[\![h]\!]}(hd(h)) \subseteq G^s_n$. Hence, $hd(h) \in G^s_{n+1}$ by Definition 17. Then, $hd(h) \in G^s_\infty$ by Definition 18. $\boxtimes$

Lemma 21 *For any positive integer* k *and any history* $w_0, i_1, w_1, \ldots, w_{k-1}$, *if* $w_{k-1} \notin G^s_\infty$, *then there is a state* $w_k \notin G^s_\infty$ *such that* $w_0, i_1, w_1, \ldots, w_{k-1}, i_k, w_k$ *is a history, where* $i_k = s[\![w_0, i_1, w_1, \ldots, w_{k-1}]\!]$.

Proof. By Lemma 20, set $(\Delta_{i_k}(w_{k-1})) \backslash G^s_\infty$ is not empty. Let w_k be any state such that $w_k \in \Delta_{i_k}(w_{k-1})$ and $w_k \notin G^s_\infty$. Then, $w_0, i_1, w_1, \ldots, w_{k-1}, i_k, w_k$ is a history by Definition 3. $\boxtimes$

Lemma 22 *For each state* $w_0 \notin G^s_\infty$ *there exists a path* $w_0, i_1, w_1, \ldots$ *under recall strategy* s *such that* $w_k \notin G^s_\infty$ *for each* $k \geqslant 0$.

Proof. Note that single-element sequence w_0 is a history by Definition 3. Due to Lemma 21, there is an infinite sequence $w_0, i_1, w_1, \ldots$ such that for each integer $k \geqslant 1$,

1. $w_0, i_1, w_1, \ldots, w_{k-1}$ is a history,

2. $i_k = s[\![w_0, i_1, w_1, \ldots, w_{k-1}]\!]$,

3. $w_k \notin G^s_\infty$.

By Definition 7, sequence $w_0, i_1, w_1, \ldots$ is a path under recall strategy s. $\boxtimes$

Lemma 23 *If* $X \vdash A \rhd B$, *then* $T(X) \vDash A \rhd B$.

Proof. Assumption $X \vdash A \rhd B$ implies $(A, B) \in I$ by Definition 15. Consider recall strategy s such that $s[\![h]\!] = (A, B)$ for each class of histories $[\![h]\!]$. Consider any path $w_0, i_1, w_1, \ldots$ under recall strategy s where $[w_0] \in A^*$. Then, $w_0 \in A$ by Lemma 13.

By Definition 11 and Definition 10 it suffices to show that $[w_1] \in B^*$. Indeed, $i_1 = (A, B)$ by choice of recall strategy s. Thus, $\Delta_{i_1}(w_0) = \Delta_{(A,B)}(w_0) = B$ by Definition 16 and due to the assumption $w_0 \in A$.

By Definition 7, sequence w_0, i_1, w_1 is a history. Hence, $w_1 \in \Delta_{i_1}(w_0)$ by Definition 3. Thus, $w_1 \in \Delta_{i_1}(w_0) = B$. Then, $[w_1] \in B^*$ by Lemma 13. $\boxtimes$

Lemma 24 *If* $T(X) \vDash E \rhd G$, *then* $X \vdash E \rhd G$.

Proof. By Definition 11, assumption $T(X) \vDash E \rhd G$ implies $Path_s(E) \subseteq Visit(G)$ for some recall strategy s.
Case I: $E \subseteq G^s_\infty$. By Lemma 19 there exists an integer $n \geqslant 0$, such that $E \subseteq G^s_n$. Thus, $\vdash E \rhd G^s_n$ by the Reflexivity axiom. At the same time, $X \vdash G^s_n \rhd G$ by Lemma 18. Therefore, $X \vdash E \rhd G$ by the Transitivity axiom.

Case II: $E \nsubseteq G^s_\infty$. Then, there is an element $w_0 \in E$ such that $w_0 \notin G^s_\infty$. Thus, by Lemma 22 there is a path $\pi = w_0, i_1, w_1, \dots$ under recall strategy s such that $w_k \notin G^s_\infty$ for all $k \geq 0$. Hence, $w_k \notin G$ for all $k \geq 0$ because $G = G^s_0 \subseteq G^s_\infty$ by Definition 17 and Definition 18. Thus, $[w_0] \in E^*$ and $[w_k] \notin G^*$ for all $k \geq 0$ by Lemma 13. Then, states in path $\pi \in Path_s(E)$ by Definition 9 and $\pi \notin Visit(G)$ by Definition 10. Therefore, $Path_s(E) \nsubseteq Visit(G)$, which contradicts the choice of strategy s. ⊠

Lemma 25 $X \vdash \varphi$ *iff* $T(X) \vDash \varphi$ *for each* $\varphi \in \Phi$.

Proof. We prove this lemma by induction on the structural complexity of φ. The base case follows from Lemma 23 and Lemma 24. The induction case follows from the maximality and the consistency of set X in the standard way. ⊠

We are now ready to state and prove the completeness theorem for the recall strategies.

Theorem 2 *If* $T \vDash \varphi$ *for every system* T, *then* $\vdash \varphi$.

Proof. Suppose $\nvdash \varphi$. Let X be a maximal consistent set containing formula $\neg\varphi$. Thus, $T(X) \vDash \neg\varphi$ by Lemma 25. Therefore, $T(X) \nvDash \varphi$. ⊠

Navigation with Memoryless Strategies

In this section we give a sound and complete axiomatization of navigability under memoryless strategies. We start by modifying Definition 11 to refer to memoryless strategies instead of recall strategies:

Definition 19 $T \vDash A \rhd B$ *if* $Path_s(A) \subseteq Visit(B)$ *for some memoryless strategy* s *of a transition system* T.

Axioms The logical system for memoryless strategies is the same as for recall strategies with the exception that the Transitivity axiom is replaced by the following principle:

3. Monotonicity: $A' \rhd B \to A \rhd B$, where $A \subseteq A'$.

This principle can be derived from Armstrong's axioms.

Theorem 3 *If* $\vdash \varphi$, *then* $T \vDash \varphi$ *for every system* T.

Proof. Soundness of the Reflexivity axiom and the Augmentation axiom is similar to the case of perfect recall, see Theorem 1. Soundness of the Monotonicity axiom follows from $Path_s(A) \subseteq Path_s(A')$, where $A \subseteq A'$. ⊠

Completeness In the rest of this section we prove completeness of our logical system with respect to the memoryless semantics. First, we define a canonical transition system $T(X) = (S, \sim, *, I, \{\Delta_i\}_{i \in I})$ for an arbitrary maximal consistent set of formulae $X \subseteq \Phi$.

Like in the perfect recall case, the canonical system has one state for each view and an additional "black hole" state ↺. Unlike the previous construction, the new canonical transition system has more additional states besides state ↺. Drawing on our original intuition of a transition system as a maze, we think about these new states as "wormholes". For any sets of states A and B in the maze there is a wormhole state $w(A,B)$ that can be used to travel one-way from set A to set B. Then, $S = V \cup \{↺\} \cup \{w(A,B) \mid A, B \subseteq V\}$.

The agent can distinguish any two different non-wormhole states, but she can not distinguish wormholes. In other words, each non-wormhole state $v \in V \cup \{↺\}$ forms its own indistinguishability class $[v] = \{v\}$, while all wormholes belong to the same single indistinguishability class of wormholes. Like in the perfect recall case, for each $v \in V$, we define v^* to be class $[v]$.

Lemma 26 $u \in A$ *iff* $[u] \in A^*$ *for each view* $u \in V$ *and each set* $A \subseteq V$. ⊠

Like in the canonical model for the perfect recall case, for sets $A, B \subseteq V$ such that $X \vdash A \rhd B$, we introduce an instruction (A, B) that can be used to navigate from set A to set B. Unlike the perfect recall case, we introduce such an instruction only if sets A and B are disjoint. This is an insignificant technical restriction that we use to simplify the proof of Lemma 29.

Definition 20

$$I = \{(A, B) \mid X \vdash A \rhd B \text{ and } A \cap B = \varnothing\}.$$

Recall that assumption $X \vDash A \rhd B$ implies that sets A and B are nonempty due to Definition 1.

In the perfect recall case, instruction (A, B) can be used to transition the system *directly* from a state in set A to a state in set B. In our case, this transition happens via the wormhole state $w(A,B)$. In other words, when instruction (A, B) is invoked in a state from set A, the system transitions into state $w(A,B)$. When the same instruction is invoked in $w(A,B)$, the system transitions into a state in B.

Definition 21

$$\Delta_{(A,B)}(u) = \begin{cases} \{w(A,B)\}, & \text{if } u \in A, \\ B, & \text{if } u = w(A,B), \\ \{↺\}, & \text{otherwise.} \end{cases}$$

Lemma 27 *If* $X \vdash A \rhd B$ *and sets* A *and* B *are disjoint, then* $T(X) \vDash A \rhd B$.

Proof. Assumptions $X \vdash A \rhd B$ and $A \cap B = \varnothing$ imply that $(A, B) \in I$, by Definition 20. Consider a memoryless strategy s such that $s(x) = (A, B)$ for each class x. By Definition 19, it suffices to show that $Path_s(A) \subseteq Visit(B)$.

Consider any $w_0, i_1, w_1, \dots \in Path_s(A)$. Then, by Definition 9, sequence $w_0, i_1, w_1, \dots$ is a path under strategy s such that $[w_0] \in A^*$. Hence, $w_0 \in A$ by Lemma 26. Thus,

$$\Delta_{(A,B)}(w_0) = \{w(A,B)\} \tag{3}$$

by Definition 21.

At the same time, $w_1 \in \Delta_{i_1}(w_0)$ by Definition 3. Hence, $w_1 \in \Delta_{s[w_0]}(w_0)$ by Definition 6. Thus, $w_1 \in \Delta_{(A,B)}(w_0)$ by the choice of strategy s. Then, $w_1 = w(A,B)$ by equation (3). Hence,

$$\Delta_{(A,B)}(w_1) = B \tag{4}$$

by Definition 21.

Similarly, $w_2 \in \Delta_{i_2}(w_1)$ by Definition 3. Hence, $w_2 \in \Delta_{s[w_1]}(w_1)$ by Definition 6. Thus, $w_2 \in \Delta_{(A,B)}(w_1)$

by the choice of strategy s. Then, $w_2 \in B$ by equation (4). Hence, $[w_2] \in B^*$ by Lemma 26. Therefore, $w_0, i_1, w_1, i_2, w_2, \dots \in Visit(B)$ by Definition 10. $\boxtimes$

Lemma 28 *If $X \vdash A \rhd B$, then $T(X) \vDash A \rhd B$.*

Proof. Suppose that $X \vdash A \rhd B$.

If $A \backslash B \neq \varnothing$. Thus, $X \vdash A \backslash B \rhd B$ by the Monotonicity Axiom. Hence, $T(X) \vDash A \backslash B \rhd B$ by Lemma 27. Therefore, $T(X) \vDash A \rhd B$ due to the soundness of the Augmentation axiom, see Theorem 3.

If $A \backslash B = \varnothing$, then $A \subseteq B$. Therefore, $T(X) \vDash A \rhd B$ due to the soundness of the Reflexivity axiom. $\boxtimes$

Recall that all wormhole states belong to a single indistinguishability class of wormholes. For any memoryless strategy s, let (A_s, B_s) be the instruction assigned by strategy s to the class of wormholes. Once strategy s is fixed, the states of the canonical transition system can be partitioned into five groups: set A_s, set $V \backslash A_s$, the single element set $\{\circlearrowleft\}$ containing the black hole state, the single element set $\{w(A_s, B_s)\}$ containing the wormhole state $w(A_s, B_s)$, and the set $\{w(C, D) \mid (C, D) \neq (A_s, B_s)\}$ of all other wormholes. Definition 21 restricts transitions under strategy s that are possible between these five groups of states. For example, from set $V \backslash A_s$ one can transition either into set $\{\circlearrowleft\}$ or into set $\{w(C, D) \mid (C, D) \neq (A_s, B_s)\}$. The arrows in Figure 2 show all possible transitions between these five groups of states allowed under Definition 21. These five groups of states can be further classified into "above the line" and "below the line" states, as shown. Notice that once the system transitions into one of the "below the line" states, it is trapped there and it will never be able to transition under the memoryless strategy s into an "above the line" state.
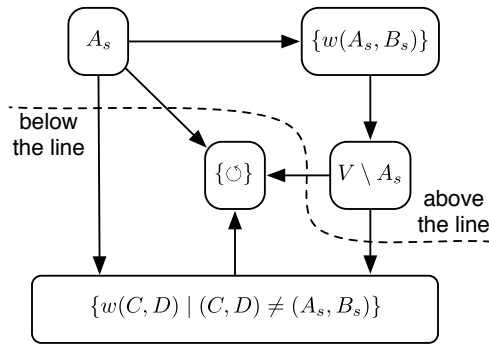

Figure 2: Transitions under memoryless strategy s.

Lemma 29 *If $T(X) \vDash E \rhd G$, then $X \vdash E \rhd G$.*

Proof. If $E \subseteq G$, then $\vdash E \rhd G$ by the Reflexivity axiom. In the rest of the proof we suppose that there is $e_0 \in E \backslash G$. By Definition 19, assumption $T(X) \vDash E \rhd G$ implies that there exists a memoryless strategy s such that $Path_s(E) \subseteq Visit(G)$.

First, we show that $E \backslash G \subseteq A_s$. Suppose there is a view $e_1 \in E$ such that $e_1 \notin G$ and $e_1 \notin A_s$. By Definition 3, single-element sequence e_1 is a history. Thus, by Lemma 2, there is a path $\pi \in Path_s(E)$ that starts with state e_1. Since path π starts in a state from set $V \backslash A_s$, all non-initial states of this path are "below the line", see Figure 2. Hence, neither of the states in path π belong to set G, because $e_1 \notin G$ and none of the states "below the line" are in set G either. Thus, $\pi \in Path_s(E)$ and $\pi \notin Visit(G)$, which contradicts the choice of strategy s. Therefore, $E \backslash G \subseteq A_s$. In particular $e_0 \in A_s$.

Second, we show that $s[e_0] = (A_s, B_s)$. Suppose that $s[e_0] = (C, D)$, where $(C, D) \neq (A_s, B_s)$. If $e_0 \notin C$, then $\pi = e_0, (C, D), \circlearrowleft, s[\circlearrowleft], \circlearrowleft, s[\circlearrowleft], \dots$ is a path under strategy s by Definition 6 and Definition 21. Note that $\pi \notin Visit(G)$ because $e_0 \notin G$ and $\pi \in Path_s(E)$. This contradicts $Path_s(E) \subseteq Visit(G)$. Similarly, if $e_0 \in C$, then sequence $\pi = e_0, (C, D), w(C, D), (A_s, B_s), \circlearrowleft, s[\circlearrowleft], \dots$ is a path such that $\pi \notin Visit(G)$ and $\pi \in Path_s(E)$, which again contradicts $Path_s(E) \subseteq Visit(G)$.

Third, we prove that $B_s \subseteq G$. Suppose that there is a state $b_0 \in B_s \backslash G$. By Definition 3, Definition 21, and the choice of instruction (A_s, B_s), sequence $e_0, (A_s, B_s), w(A_s, B_s), (A_s, B_s), b_0$ is a history. Thus, by Lemma 2, there is a path $\pi \in Path_s(E)$ that starts as $e_0, (A_s, B_s), w(A_s, B_s), (A_s, B_s), b_0$. Assumption $b_0 \in B_s \backslash G$ implies that $b_0 \in V \backslash A_s$ because sets A_s and B_s are disjoint by Definition 20. Thus, path π after state b_0 contains only states "below the line", see Figure 2, none of which are in set $G \subseteq V$. Recall also that $e_0, b_0 \notin G$ by the choice of states e_0 and b_0. Thus, $\pi \in Path_s(E)$ and $\pi \notin Visit(G)$, which again contradicts $Path_s(E) \subseteq Visit(G)$.

Note that $X \vdash A_s \rhd B_s$ by Definition 20. Hence, $X \vdash (A_s \cup G) \rhd (B_s \cup G)$ by the Augmentation axiom. Thus, $X \vdash (A_s \cup G) \rhd G$ because $B_s \subseteq G$. At the same time, $E \backslash G \subseteq A_s$ implies that $E \subseteq A_s \cup G$. Therefore, $X \vdash E \rhd G$ by the Monotonicity axiom.

$\boxtimes$

Lemma 30 $X \vdash \varphi$ *iff* $T(X) \vDash \varphi$ *for each* $\varphi \in \Phi$.

Proof. We prove this lemma by induction on the structural complexity of φ. The base case follows from Lemma 28 and Lemma 29. The induction case follows from the maximality and the consistency of set X in the standard way. $\boxtimes$

We are now ready to state and prove the completeness theorem for memoryless strategies.

Theorem 4 *If $T \vDash \varphi$ for every system T, then $\vdash \varphi$.*

Proof. Suppose $\nvdash \varphi$. Let X be a maximal consistent set containing formula $\neg\varphi$. Thus, $T(X) \vDash \neg\varphi$ by Lemma 30. Therefore, $T(X) \nvDash \varphi$. $\boxtimes$

Conclusion

In this paper we have shown that the properties of navigability under perfect recall strategies are exactly those described by Armstrong's axioms for functional dependency

in database theory. In the absence of perfect recall, the Transitivity axiom is no longer valid, but it could be replaced by the Monotonicity axiom.

References

Ågotnes, T., and Alechina, N. 2012. Epistemic coalition logic: completeness and complexity. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2 (AAMAS)*, 1099–1106. International Foundation for Autonomous Agents and Multiagent Systems.

Ågotnes, T.; Balbiani, P.; van Ditmarsch, H.; and Seban, P. 2010. Group announcement logic. *Journal of Applied Logic* 8(1):62 – 81.

Ågotnes, T.; van der Hoek, W.; and Wooldridge, M. 2009. Reasoning about coalitional games. *Artificial Intelligence* 173(1):45 – 79.

Alur, R.; Henzinger, T. A.; and Kupferman, O. 2002. Alternating-time temporal logic. *Journal of the ACM* 49(5):672–713.

Armstrong, W. W. 1974. Dependency structures of data base relationships. In *Information Processing 74 (Proc. IFIP Congress, Stockholm, 1974)*. Amsterdam: North-Holland. 580–583.

Belardinelli, F. 2014. Reasoning about knowledge and strategies: Epistemic strategy logic. In *Proceedings 2nd International Workshop on Strategic Reasoning, SR 2014, Grenoble, France, April 5-6, 2014*, volume 146 of *EPTCS*, 27–33.

Borgo, S. 2007. Coalitions in action logic. In *20th International Joint Conference on Artificial Intelligence*, 1822–1827.

Fervari, R.; Herzig, A.; Li, Y.; and Wang, Y. 2017. Strategically knowing how. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, 1031–1038.

Garcia-Molina, H.; Ullman, J.; and Widom, J. 2009. *Database Systems: The Complete Book*. Prentice-Hall, second edition.

Goranko, V. 2001. Coalition games and alternating temporal logics. In *Proceedings of the 8th conference on Theoretical aspects of rationality and knowledge*, 259–272. Morgan Kaufmann Publishers Inc.

Jamroga, W., and Ågotnes, T. 2007. Constructive knowledge: what agents can achieve under imperfect information. *Journal of Applied Non-Classical Logics* 17(4):423–475.

Jamroga, W., and van der Hoek, W. 2004. Agents that know how to play. *Fundamenta Informaticae* 63(2-3):185–219.

Li, Y., and Wang, Y. 2017. Achieving while maintaining: A logic of knowing how with intermediate constraints. In *Logic and Its Applications: Proceedings of 7th Indian Conference, ICLA 2017, Kanpur, India, January 5-7, 2017*, 154–167. Springer.

More, S. M., and Naumov, P. 2012. Calculus of cooperation and game-based reasoning about protocol privacy. *ACM Trans. Comput. Logic* 13(3):22:1–22:21.

Naumov, P., and Tao, J. 2017a. Coalition power in epistemic transition systems. In *Proceedings of the 2017 International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 723–731.

Naumov, P., and Tao, J. 2017b. Together we know how to achieve: An epistemic logic of know-how. In *16th conference on Theoretical Aspects of Rationality and Knowledge (TARK), July 24-26, 2017, EPTCS 251*, 441–453.

Pauly, M. 2001. *Logic for Social Software*. Ph.D. Dissertation, Institute for Logic, Language, and Computation.

Pauly, M. 2002. A modal logic for coalitional power in games. *Journal of Logic and Computation* 12(1):149–166.

Sauro, L.; Gerbrandy, J.; van der Hoek, W.; and Wooldridge, M. 2006. Reasoning about action and cooperation. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, AAMAS '06, 185–192. New York, NY, USA: ACM.

van Benthem, J. 2001. Games in dynamic-epistemic logic. *Bulletin of Economic Research* 53(4):219–248.

van der Hoek, W., and Wooldridge, M. 2003. Cooperation, knowledge, and time: Alternating-time temporal epistemic logic and its applications. *Studia Logica* 75(1):125–157.

van der Hoek, W., and Wooldridge, M. 2005. On the logic of cooperation and propositional control. *Artificial Intelligence* 164(1):81 – 119.

Wang, Y. 2015. A logic of knowing how. In *Logic, Rationality, and Interaction*. Springer. 392–405.

Wang, Y. 2016. A logic of goal-directed knowing how. *Synthese*.