An Intuitionistic Epistemic Logic for Sequential Consistency on Shared Memory

Yoichi Hirai*

Abstract

In the celebrated Gödel Prize winning papers, Herlihy, Shavit, Saks and Zaharoglou gave topological characterization of waitfree computation. In this paper, we characterize waitfree communication logically. First, we give an intuitionistic epistemic logic $\mathbf{K}\vee$ for asynchronous communication. The semantics for the logic $\mathbf{K}\vee$ is an abstraction of Herlihy and Shavit's topological model. In the same way Kripke model for intuitionistic logic informally describes an agent increasing its knowledge over time, the semantics of $\mathbf{K}\vee$ describes multiple agents passing proofs around and developing their knowledge together. On top of the logic $\mathbf{K}\vee$, we give an axiom type that characterizes sequential consistency on shared memory. The advantage of intuitionistic logic over classical logic then becomes apparent as the axioms for sequential consistency are meaningless for classical logic because they are classical tautologies. The axioms are similar to the axiom type for prelinerilty $(\varphi \supset \psi) \vee (\psi \supset \varphi)$. This similarity reflects the analogy between sequential consistency for shared memory scheduling and linearity for Kripke frames: both require total order on schedules or models. Finally, under sequential consistency, we give soundness and completeness between a set of logical formulas called waitfree assertions and a set of models called waitfree schedule models.

1 Introduction

Waitfree Computation The main purpose of this paper is to characterize waitfree communication logically (Theorem 4.5) in a language as simple as possible. Waitfreedom [11] is a restriction on distributed programs over shared memory. It forbids any process to wait for another process. Some tasks can be solved by a well-chosen waitfree protocol while the others cannot.

For example, it is waitfreely impossible for each one of two processes to attain the input value of the other process. On the other hand, it is waitfreely possible for either one of two processes to attain the input value of the other process. A waitfree protocol that solves this task is:

- process a tells the memory m that φ holds, and then m replies back to a,
- process b tells the memory m that ψ holds, and then m replies back to b.

After this protocol finishes, either φ has been communicated from a to b or ψ has been communicated from b to a. In the logic $\mathbf{K}\vee$, this fact is represented by a formula $(K_aK_{\mathsf{m}}K_a\varphi \wedge K_bK_{\mathsf{m}}K_b\psi) \supset (K_aK_b\psi \vee K_bK_a\varphi)$, which is deducible in $\mathbf{K}\vee$ with sequential consistency (Figure 2).

Herlihy and Shavit [12] characterized waitfree computation using simplicial topology (See Section 6). Using their characterization, Gafni and Koutsoupias [9] showed that it is undecidable whether a task is waitfreely solvable or not. In this paper we show that, when tasks are restricted to communication defined by a class of logical formulas we call waitfree assertions, it is decidable whether a task is waitfreely solvable or not (Subsection 4.1).

*University of Tokyo, Dept. of Computer Science, 7-3-1 Hongo, Tokyo 113-0033 Japan. yh@lyon.is.s.u-tokyo.ac.jp

Sequential Consistency The topological characterization by Herlihy and Shavit [12] implicitly assumes sequential consistency [17] for shared memory. Since we seek to use a simple language, we state sequential consistency explicitly in the language. We characterize sequential consistency with an axiom type $(K_{\mathsf{m}}\varphi \supset K_{\mathsf{m}}\psi) \vee (K_{\mathsf{m}}\psi \supset K_{\mathsf{m}}\varphi)$ in the logic $\mathbf{K}\vee$ for asynchronous computation. The axiom type is sound (Theorem 3.3) and strong complete (Theorem 3.9) for a class of models called sequential models where memory states are temporarily lined up in a total order.

Asynchronous Communication We define an intuitionistic modal propositional logic that we call $\mathbf{K}\vee$ and show soundness (Theorem 2.8) and strong completeness (Theorem 2.14) for Kripke semantics. The semantics of $\mathbf{K}\vee$ is simple: it has only one function for each agent in addition to the Kripke model for intuitionistic propositional logic. We deliberately identify the partial order in Kripke frame with the temporal relation. Intuitionistic logic can be seen as a logic describing an agent whose knowledge increases over time. The logic $\mathbf{K}\vee$ can be seen as a logic describing multiple agents that asynchronously communicate with each other and increase their knowledge. Although $\mathbf{K}\vee$ deals with communication, the logic has only epistemic modalities so that it has simpler syntax than many other logics for communication.

There are other choices: there have been proposed a huge number of epistemic logics for communication [3, 4, 5, 6, 10, 14, 18, 22, 23, 29] and a huge number of intuitionistic modal logics [1, 7, 21, 22, 24]. In both cases, when considered under Kripke semantics, the huge variety of logics comes from the diversity of relationships between two binary relations on the state space. In intuitionistic modal logic, the two relations are: (a) which state is prior to which state with regard to Kripke monotonicity and (b) the modality in which state refers to which state. In logics for communication, the two relations are: (a') which state is temporarily prior to which state and (b') from which state to which state communication occurs.

The semantics of $\mathbf{K}\vee$ uses a binary relation and *functions* on possible worlds instead of additional binary relations. This choice dramatically limits the room for design choice. Also, we identify relations (a) with (a') and (b) with (b') in order to make the language of $\mathbf{K}\vee$ simpler.

Structure of Paper Although this introduction so far is organized in the top-to-bottom order, the rest of this paper is in the opposite bottom-to-top order. Sections 2–4 respectively treat asynchronous computation in general, sequential consistency and waitfree communication.

2 Intuitionistic Epistemic Logic for Asynchronous Communication

2.1 Syntax

We fix a countably infinite set of propositional variables $PVar$ and a set of agents A. We use the metavariables $P, Q, \dots$ running over $PVar$ and $a, b, \dots$ running over A.

Definition 2.1. *We define a formula φ by the BNF:*

$$\varphi ::= \bot \mid P \mid (K_a\varphi) \mid (\varphi \vee \varphi) \mid (\varphi \wedge \varphi) \mid (\varphi \supset \varphi).$$

The unary operators connect more strongly than the binary operators. We sometimes omit the parentheses when no confusion occurs. We use $=$ for syntactic equality of formulas. The notation $(\neg\varphi)$ stands for $(\varphi \supset \bot)$. For a sequence of formulas $\Gamma = (\varphi_i)_{i\in I}$ or a set of formulas Γ, the notation $K_a\Gamma$ stands for the sequence $(K_a\varphi_i)_{i\in I}$ or the set $\{K_a\varphi \mid \varphi \in \Gamma\}$ respectively.

Definition 2.2. *We define the proof system of* $\mathbf{K}\vee$ *by Figure 1.*

For a set of formula Γ and a formula φ, $\Gamma \vdash \varphi$ denotes a relation where there is such a finite sequence Γ_0 that $\Gamma_0 \vdash \varphi$ is deducible and that Γ_0 contains only formulas in Γ.

$$\text{(axiom)}\ \frac{}{\varphi \vdash \varphi} \qquad \text{(weakening)}\ \frac{\Gamma \vdash \varphi}{\psi, \Gamma \vdash \varphi} \qquad \text{(contraction)}\ \frac{\varphi, \varphi, \Gamma \vdash \varphi'}{\varphi, \Gamma \vdash \varphi'}$$

$$\text{(exchange)}\ \frac{\Gamma, \varphi, \psi, \Gamma' \vdash \varphi'}{\Gamma, \psi, \varphi, \Gamma' \vdash \varphi'} \qquad (\wedge\text{-I})\ \frac{\Gamma \vdash \varphi \qquad \Gamma' \vdash \psi}{\Gamma, \Gamma' \vdash \varphi \wedge \psi} \qquad (\vee\text{-I}_0)\ \frac{\Gamma \vdash \varphi}{\Gamma \vdash \varphi \vee \psi} \qquad (\vee\text{-I}_1)\ \frac{\Gamma \vdash \varphi}{\Gamma \vdash \psi \vee \varphi}$$

$$(\wedge\text{-E}_0)\ \frac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \varphi} \qquad (\wedge\text{-E}_1)\ \frac{\Gamma \vdash \varphi \wedge \psi}{\Gamma \vdash \psi} \qquad (\vee\text{-E})\ \frac{\Gamma \vdash \psi_0 \vee \psi_1 \qquad \Gamma, \psi_0 \vdash \varphi \qquad \Gamma, \psi_1 \vdash \varphi}{\Gamma \vdash \varphi}$$

$$(\supset\text{-I})\ \frac{\varphi, \Gamma \vdash \psi}{\Gamma \vdash \varphi \supset \psi} \qquad (\supset\text{-E})\ \frac{\Gamma \vdash \psi_0 \supset \psi_1 \qquad \Gamma \vdash \psi_0}{\Gamma \vdash \psi_1} \qquad (\bot\text{-E})\ \frac{\Gamma \vdash \bot}{\Gamma \vdash \varphi} \qquad \text{(T)}\ \frac{}{K_a\varphi \vdash \varphi}$$

$$\text{(introspection)}\ \frac{}{K_a\varphi \vdash K_aK_a\varphi} \qquad \text{(nec)}\ \frac{\Gamma \vdash \varphi}{K_a\Gamma \vdash K_a\varphi} \qquad (\vee K)\ \frac{}{K_a(\varphi \vee \psi) \vdash (K_a\varphi) \vee K_a\psi}$$

Figure 1: Deduction rules of $\mathbf{K}\vee$.

2.2 Semantics

We define validity of a formula on a state in a model. A model is a Kripke model for propositional intuitionistic logic equipped with an additional mapping $f_a : W \to W$ for each agent $a \in A$ where W is the set of possible states. Informally[1], the function f_a represents the "view" of agent a. When the current state is $w \in W$, agent a sees that the current state is $f_a(w) \in W$, in other words, agent a knows everything valid in $f_a(w)$. Agent a also sees that agent b sees that the current state is $f_b(f_a(w)) \in W$ because we assume that all agents know the frame structure and the functions f_x explicitly or implicitly. This model is an abstraction of Herlihy and Shavit's model of waitfree computation [12]. See Section 6 for details.

Definition 2.3. *A model $\langle W, \preceq, (f_a)_{a\in A}, \rho\rangle$ is a tuple of following things:*

1. *$\langle W, \preceq\rangle$ is a partial order,*
2. *$f_a \colon W \to W$ is a function satisfying all of the following conditions for any $w \in W$:*
   (a) *(descending) $f_a(w) \preceq w$,*
   (b) *(idempotency) $f_a(f_a(w)) = f_a(w)$,*
   (c) *(monotonicity) $w \preceq v$ implies $f_a(w) \preceq f_a(v)$,*
3. *$\rho \colon PVar \to \mathscr{P}(W)$ is a function such that each $\rho(P)$ is upward-closed with respect to $\preceq$, i.e., $w' \succeq w \in \rho(P)$ implies $w' \in \rho(P)$.*

With the informal account in mind, the conditions on f_a have rationales: descending condition says an agent a recognizes only truth, idempotency says an agent a recognizes that a recognizes something whenever the agent a recognizes that thing, and monotonicity says an agent a does not forget things once they recognized. Differently from classical epistemic logic, there is no distinction between global states and local states.

Definition 2.4. *We define the validity relation $\models$ of a model $\langle W, \preceq, (f_a)_{a\in A}, \rho\rangle$, a state $w \in W$ of the model and a formula φ. Let us fix a model $M = \langle W, \preceq, f, \rho\rangle$ and abbreviate $M, w \models \varphi$ into $w \models \varphi$. The definition of $\models$ is inductive on the structure of φ.*

[1] This account is informal in that we do not attempt to define the terms "view" and "current state".

(Case $\varphi = \bot$) *$w \models \bot$ never holds.*

(Case $\varphi = P$) *$w \models P$ if and only if $w \in \rho(P)$.*

(Case $\varphi = K_a\psi$) *$w \models K_a\psi$ if and only if $f_a(w) \models \psi$.*

(Case $\varphi = \psi_0 \wedge \psi_1$) *$w \models \psi_0 \wedge \psi_1$ if and only if both $w \models \psi_0$ and $w \models \psi_1$ hold.*

(Case $\varphi = \psi_0 \vee \psi_1$) *$w \models \psi_0 \vee \psi_1$ if and only if either $w \models \psi_0$ or $w \models \psi_1$ holds.*

(Case $\varphi = \psi_0 \supset \psi_1$) *$w \models \psi_0 \supset \psi_1$ if and only if for any $w' \in W$, $w' \succeq w$ and $M, w' \models \psi_0$ imply $M, w' \models \psi_1$.*

Theorem 2.5 (Kripke monotonicity). *$M, w \models \varphi$ and $w \preceq v$ imply $M, v \models \varphi$.*

Proof. By simple structural induction on φ. □

Definition 2.6. *For a model M, a state w of M and a set of formulas Γ, we write $M, w \models \Gamma$ when $M, w \models \varphi$ holds for any formula $\varphi \in \Gamma$.*

Definition 2.7. *$\Gamma \models \varphi$ stands for the relation of a set of a formula Γ and a formula φ where $M, w \models \Gamma$ implies $M, w \models \varphi$ for any model M and a state $w \in M$.*

2.3 Soundness

Theorem 2.8 (Soundness). *$\Gamma \vdash \varphi$ implies $\Gamma \models \varphi$.*

Proof. We prove soundness with induction on the definition of $\vdash$. We fix a model M and we abbreviate $M, w \models \varphi$ into $w \models \varphi$.

(axiom)(weakening)(contraction)(exchangeL) Trivial.

($\supset$-I) Assume $\Gamma, \varphi \models \psi$. Assume $w \models \Gamma$. Also assume that there is such a state w' in M that $w' \succeq w$ and $w' \models \varphi$ hold. By Lemma 2.5, $w' \models \Gamma$ holds. Since $\Gamma, \varphi \models \psi$, the relation $\Gamma, w' \models \psi$ holds.

($\supset$-E) Assume $\Gamma \models \varphi \supset \psi$ and $\Gamma \models \varphi$. By the second assumption, $w \models \varphi$ holds. The first assumption says $w \models \varphi \supset \psi$. Since $w \succeq w$, the relation $w \models \psi$ holds.

($\wedge$-I)($\vee$-I$_0$)($\vee$-I$_1$)($\vee$-E)($\wedge$-E$_0$)($\wedge$-E$_1$) Trivial.

(T) Assume $w \models K_a\varphi$. By definition of $\models$, $f_a(w) \models \varphi$ holds. Since $f_a(w) \preceq w$, Lemma 2.5 says $w \models \varphi$.

(introspection) Assume $w \models \varphi$. By definition of $\models$, $f_a(w) \models \varphi$ holds. Since f is idempotent, $f_a(f_a(w)) \models \varphi$. Applying the definition of $\models$ again, we obtain $w \models K_a\varphi$.

(nec) Assume $\Gamma \models \varphi$ and $w \models K_a\Gamma$ hold. Since $f_a(w) \models \Gamma$, The first assumption says $f_a(w) \models \varphi$. By definition of $\models$, the relation $w \models K_a\varphi$ holds.

($\vee K_a$) Assume $\Gamma \models K_a(\varphi \vee \psi)$. For any state w of any model M, assume $w \models K_a(\varphi \vee \psi)$. By the definition of $\models$, $f_a(w) \models \varphi \vee \psi$. Applying the definition of $\models$ again, either $f_a(w) \models \varphi$ or $f_a(w) \models \psi$ holds. This implies either $w \models K_a\varphi$ or $w \models K_a\psi$ holds. We have $w \models K_a\varphi \vee K_a\psi$.

□

2.4 Strong Completeness

We show strong completeness for $\mathbf{K}\vee$ with canonical model construction as in [28, Ch. 2].

Definition 2.9. *A set of formulas Γ is saturated if and only if all of these conditions are satisfied:*

1. *Γ is deductively closed, i.e., $\Gamma \vdash \varphi \Rightarrow \varphi \in \Gamma$,*

2. *$\varphi \vee \psi \in \Gamma \Rightarrow \varphi \in \Gamma$ or $\psi \in \Gamma$,*

3. *$\Gamma \nvdash \bot$.*

Lemma 2.10 (Saturation lemma)**.** *For a set of formulas Γ with $\Gamma \nvdash \varphi$, there exists a saturated set Γ^ω with $\Gamma^\omega \nvdash \varphi$ and $\Gamma \subseteq \Gamma^\omega$.*

Proof. We can enumerate all formulas in a sequence $(\varphi_i)_{i\in\mathbb{N}^+}$. We define Γ^i inductively:

(Case $i = 0$**)** $\Gamma^0 = \Gamma$,

(Case $i > 0$**)** if $\{\varphi_i\} \cup \Gamma^{i-1} \nvdash \varphi$, $\Gamma^i = \{\varphi_i\} \cup \Gamma^{i-1}$; otherwise, $\Gamma^i = \Gamma^{i-1} \cup \{\varphi_i \supset \varphi\}$. Using these Γ^i, we define $\Gamma^\omega = \bigcup_{i\in\omega} \Gamma^i$.

Claim: $\Gamma^\omega \nvdash \varphi$. Seeking contradiction, assume $\Gamma^\omega \vdash \varphi$. Since only finite number of formulas in Γ are used to prove φ, there exists a minimal i with $\Gamma^i \vdash \varphi$. Since $\Gamma \nvdash \varphi$, $i \neq 0$. Either $\Gamma^i = \{\varphi_i\} \cup \Gamma^{i-1}$ or $\Gamma^i = \{\varphi_i \supset \varphi\} \cup \Gamma^{i-1}$. The first case is explicitly forbidden. In the second case, $\Gamma^{i-1}, \varphi_i \supset \varphi \vdash \varphi$ holds. That means $\Gamma^{i-1} \vdash (\varphi_i \supset \varphi) \supset \varphi$. Also, since we could not take the first case, $\Gamma^{i-1}, \varphi_i \vdash \varphi$ holds. That means $\Gamma^{i-1} \vdash \varphi_i \supset \varphi$. These combined, $\Gamma^{i-1} \vdash \varphi$ holds, which contradicts to the minimality of i.
Claim: Γ^ω is a saturated set.

Proof of Claim. 1. Assume $\Gamma^\omega \vdash \psi$. There is $i \in \mathbb{N}^+$ with $\varphi_i = \psi$. We know that $\Gamma^{i-1} \cup \{\varphi_i\} \nvdash \varphi$. It means $\psi \in \Gamma^\omega$.

2. Assume $\psi_0 \vee \psi_1 \in \Gamma^\omega$. Seeking contradiction, assume $\psi_0 \notin \Gamma^\omega$ and $\psi_1 \notin \Gamma^\omega$. By construction, $\Gamma^\omega \vdash \psi_0 \supset \varphi$ and $\Gamma^\omega \vdash \psi_1 \supset \varphi$. Since Γ^ω is deductively closed, by ($\vee$-E) rule, we have $\Gamma^\omega \vdash \varphi$, which contradicts to the previous fact.

3. Since $\Gamma^\omega \nvdash \varphi$, $\Gamma^\omega \nvdash \bot$.

□

Since $\Gamma = \Gamma_0$, Γ^ω contains Γ^0. The lemma is now proved. □

Definition 2.11 (Canonical model candidate)**.** *We define a tuple $M^C = \langle W^C, \preceq^C, (f_a^C)_{a\in A}, \rho^C \rangle$.*

- *W^C is the set of saturated sets of formulas,*

- *$\Gamma \preceq^C \Delta$ if and only if $\Gamma \subseteq \Delta$,*

- *$f_a^C(\Gamma) = \{\varphi \mid K_a\varphi \in \Gamma\}$,*

- *$\rho^C(P) = \{\Gamma \mid P \in \Gamma\}$.*

Lemma 2.12 (Canonical model)**.** *The tuple $M^C = \langle W^C, \preceq^C, (f_a^C)_{a\in A}, \rho^C \rangle$ is a model.*

Proof. First, let us check f_a^{C} is actually a function $W^{\mathsf{C}} \to W^{\mathsf{C}}$. Assume $\Gamma \in W^{\mathsf{C}}$.
Claim: $f_a(\Gamma)$ is a saturated set of formulas.

Proof of Claim. To prove the claim, we check each condition on the Definition 2.9 of saturated sets.

1. Assume $f_a^{\mathsf{c}}(\Gamma) \vdash \varphi$. By rule (nec),$K_a(f_a(\Gamma)) \vdash K_a\varphi$. Since $K_a(f_a^{\mathsf{c}}(\Gamma)) \subseteq \Gamma$, the relation $\Gamma \vdash K_a\varphi$ holds. Since Γ is deductively closed, $K_a\varphi \in \Gamma$. By definition of f_a^{c}, $\varphi \in f_a^{\mathsf{c}}(\Gamma)$.

2. Assume $\varphi \vee \psi \in f_a^{\mathsf{c}}(\Gamma)$. By definition of f_a^{c}, $K_a(\varphi \vee \psi) \in \Gamma$. By rule ($\vee K_a$), $K_a(\varphi \vee \psi) \vdash K_a\varphi \vee K_a\psi$. Since Γ is deductively closed, $K_a\varphi \vee K_a\psi \in \Gamma$. Since Γ is saturated, either $K_a\varphi \in \Gamma$ or $K_a\psi \in \Gamma$. By definition of f_a^{c}, either $\varphi \in f_a^{\mathsf{c}}(\Gamma)$ or $\psi \in f_a^{\mathsf{c}}(\Gamma)$.

3. Seeking contradiction, assume $f_a^{\mathsf{c}}(\Gamma) \vdash \bot$. Since $f_a^{\mathsf{c}}(\Gamma)$ is deductively closed, $\bot \in f_a^{\mathsf{c}}(\Gamma)$. By definition of f_a^{c}, $K_a\bot \in \Gamma$. Because of the rule (T), $\Gamma \vdash \bot$. This contradicts to the assumption of Γ being a saturated set.

□

Now, let us check each condition in Definition 2.3 to make sure the tuple is actually a model:

1. $\preceq^{\mathsf{c}}$ is a partial order because set theoretic inclusion $\subseteq$ is a partial order.

2. (a) $f_a^{\mathsf{c}}(\Gamma) \preceq \Gamma$ of the rule (T).
   (b) $f_a^{\mathsf{c}}(f_a^{\mathsf{c}}(\Gamma)) \subseteq f_a^{\mathsf{c}}(\Gamma)$ is now obvious from the previous line. Let us show the opposite. Assume $\varphi \in f_a^{\mathsf{c}}(\Gamma)$. By definition of f_a^{c}, $K_a\varphi \in \Gamma$. By the rule (introspection), $\Gamma \vdash K_aK_a\varphi$. Since Γ is deductively closed, $K_aK_a\varphi \in \Gamma$. Thus $\varphi \in f_a^{\mathsf{c}}(f_a^{\mathsf{c}}(\Gamma))$.
   (c) Assume $\Gamma \preceq \Delta$. Every $K_a\varphi \in \Delta$ is also in Γ. Thus $f_a^{\mathsf{c}}(\Gamma) \preceq f_a^{\mathsf{c}}(\Delta)$.

3. Assume $\Gamma' \succeq \Gamma \in \rho^{\mathsf{c}}(P)$. $P \in \Gamma$. So $P \in \Gamma'$. Thus $\Gamma' \in \rho^{\mathsf{c}}(P)$.

□

Lemma 2.13. *For a saturated set of formula Γ and the canonical model M^C, an equivalency $\varphi \in \Gamma \Leftrightarrow M^C, \Gamma \vdash \varphi$ holds.*

Proof. By induction on φ.

(Case $\varphi = \bot$) Neither side ever holds.

(Case $\varphi = P$) By definition of ρ^{c}, $\varphi \in \Gamma \Leftrightarrow \Gamma \in \rho(P) \Leftrightarrow M^{\mathsf{c}}, \Gamma \models P$.

(Case $\varphi = \psi_0 \wedge \psi_1$)(Case $\varphi = \psi_0 \vee \psi_1$)(Case $\varphi = K_a\psi$) Directly from the induction hypothesis.

(Case $\varphi = \psi_0 \supset \psi_1$) ($\Rightarrow$) Assume $M^{\mathsf{c}}, \Gamma \models \psi_0 \supset \psi_1$. Seeking contradiction, assume $\psi_0 \supset \psi_1 \notin \Gamma$. Since Γ is deductively closed, $\Gamma, \psi_0 \nvdash \psi_1$. By Lemma 2.10, there exists a saturated set Γ' with $\Gamma' \supseteq \Gamma \cup \{\psi_0\}$ and $\Gamma' \nvdash \psi_1$. By induction hypothesis, $M^{\mathsf{c}}, \Gamma' \models \psi_0$ but not $M^{\mathsf{c}}, \Gamma' \models \psi_1$. Since $\Gamma' \succeq \Gamma$, this contradicts to $M^{\mathsf{c}}, \Gamma \models \psi_0 \supset \psi_1$.
($\Leftarrow$) Assume $\psi_0 \supset \psi_1 \in \Delta$, $\Delta' \succeq \Delta$ and $M^{\mathsf{c}}, \Delta' \models \psi_0$. Showing $M^{\mathsf{c}}, \Delta' \models \psi_1$ is enough. By induction hypothesis, $\psi_0 \in \Delta'$. Since Δ' is deductively closed and $\psi_0 \supset \psi_1 \in \Delta'$, $\psi_1 \in \Delta'$. By induction hypothesis, $M^{\mathsf{c}}, \Delta' \models \psi_1$.

Now we have shown the lemma.

□

Theorem 2.14 (Strong completeness)**.** $\Gamma \models \varphi$ *implies* $\Gamma \vdash \varphi$.

Proof. We show the contraposition: assuming $\Gamma \not\vdash \varphi$, we show $\Gamma \not\models \varphi$. By Lemma 2.10, there is a saturated set of formula Γ' with $\Gamma' \not\vdash \varphi$ and $\Gamma' \supseteq \Gamma$. By Lemma 2.13, $M^{\mathrm{c}}, \Gamma' \models \Gamma$ but not $M^{\mathrm{c}}, \Gamma' \models \varphi$. This denies $\Gamma \models \varphi$. □

Is it decidable whether a formula is a theorem of **K**$\vee$ or not? Does **K**$\vee$ have finite model property? These are interesting problems. When the Law of Excluded Middle is added to **K**$\vee$, the obtained logic has all the theorems of both classical epistemic logic and the logic $(\mathbf{Alt})^A$. In classical epistemic logic, when there are more than two agents, Maddux's algebraic result in [19] implies that it is undecidable whether a formula is a theorem or not. The result in Maddux [19] also implies that classical epistemic logic does not have finite model property when there are more than two agents. On the other hand, the classical modality $(\mathbf{Alt})^n$, whose modality is defined by a function on Kripke frames, is axiomatizable and has finite model property [8] regardless of the number n of **Alt**-modalities. Since **K**$\vee$ is similar to both logics, it is interesting whether **K**$\vee$ has finite model property and decidability.

3 Axiom Type for Sequential Consistency

A schedule determines temporal partial order of events such as message sending and receiving. A correct program must behave correctly under every schedule. Shared memory consistency is a restriction on schedules. When a stronger memory consistency is posed, it is easier for programs to behave correctly. This is analogous to the fact that when a stronger condition on models implies more valid formulas.

In this section, we characterize sequential consistency with a set of axioms. Sequential consistency defined by Lamport [17] is essentially a condition requiring the states of memory lined up in a total order. We define a deduction system $\vdash_{SC}$ by adding an axiom type to **K**$\vee$ and characterize sequential consistency.

Henceforth, we assume $A = \{\mathsf{m}\} \cup P$ ($\mathsf{m} \notin P$), where P is the set of processes and m represents the shared memory.

Definition 3.1. *We let SC be the set of formula of the form* $(K_{\mathsf{m}}\varphi \supset K_{\mathsf{m}}\psi) \vee (K_{\mathsf{m}}\psi \supset K_{\mathsf{m}}\varphi)$.
We add a rule (SC) to the previous calculus $\vdash$: $(SC)\,\dfrac{}{\vdash \varphi}\ (\varphi \in SC)$
We define $\Gamma \vdash_{SC} \varphi$ *in the same way as* $\Gamma \vdash \varphi$.

Note that all axioms in the set *SC* are classical tautologies so that adding these axioms to classical logic is meaningless. This is the merit of using intuitionistic logic rather than classical logic.

Definition 3.2. *A sequential model is a model where for any states w and w' either $w \preceq w'$ or $w' \preceq w$ holds if $f_{\mathsf{m}}(w) = w$, $f_{\mathsf{m}}(w') = w'$ and there exists a state x with $x \preceq v$ and $x \preceq w$.*

3.1 Soundness

Lemma 3.3. $\vdash_{SC} \varphi \Rightarrow M \models \varphi$ *for any sequential model M.*

Proof. We extend the induction of Lemma 2.8 with a clause for the rule (SC).

(SC) Seeking contradiction, assume $M, w \not\models (K_{\mathsf{m}}\varphi \supset K_{\mathsf{m}}\psi) \vee (K_{\mathsf{m}}\psi \supset K_{\mathsf{m}}\varphi)$. The definition for $\models$ says that there exist states $w_0, w_1 \succeq w$ with $M, w_0 \models K_{\mathsf{m}}\varphi$, $M, w_1 \models K_{\mathsf{m}}\psi$, $M, w_1 \not\models K_{\mathsf{m}}\psi$ and $M, w_0 \not\models K_{\mathsf{m}}\varphi$. These and Kripke monotonicity (Lemma 2.5) contradicts to the assumption that M is a sequential model.

Other cases are the same as Lemma 2.8. □

3.2 Strong Completeness

Definition 3.4. *A set of formulas Γ is SC-saturated if and only if all of these conditions are satisfied:*

1. *Γ is SC-deductively closed, i.e., $\Gamma \vdash_{\mathrm{SC}} \varphi \Rightarrow \varphi \in \Gamma$,*
2. *$\varphi \vee \psi \in \Gamma \Rightarrow \varphi \in \Gamma$ or $\psi \in \Gamma$,*
3. *$\Gamma \nvdash_{\mathrm{SC}} \bot$.*

Lemma 3.5 (Saturation lemma)**.** *For a set of formulas Γ with $\Gamma \nvdash_{\mathrm{SC}} \varphi$, there exists a saturated set of formulas Γ^{ω} with $\Gamma^{\omega} \nvdash_{\mathrm{SC}} \varphi$ and $\Gamma \subset \Gamma^{\omega}$.*

Proof. The same as Lemma 2.10 where each $\vdash$ is replaced by $\vdash_{\mathrm{SC}}$. □

Definition 3.6 (Canonical model candidate for sequential consistency)**.** *We define a tuple $M^{SC} = \langle W^{\mathrm{SC}}, \preceq^{\mathrm{SC}}, (f_a^{\mathrm{SC}})_{a \in A}, \rho^{\mathrm{SC}} \rangle$ in the same way as Definition 2.11 of M^C except that W^{SC} is the set of SC-saturated sets of formulas.*

Lemma 3.7 (Canonical model for sequential consistency)**.** *The tuple M^{SC} is a sequential model.*

Proof. First, we can show, in the same way as before, that checking f_a^{SC} is actually a function $W^{\mathrm{SC}} \to W^{\mathrm{SC}}$. Also, checking each condition in Definition 2.3 is similar so that we see M^{SC} is actually a model. Finally, to see that the model M^{SC} is sequential, let Γ, Δ and Θ be states of M^{SC} and assume $\Theta \preceq^{\mathrm{SC}} \Delta$, $\Theta \preceq^{\mathrm{SC}} \Delta$, $f_{\mathsf{m}}^{\mathrm{SC}}(\Gamma) = \Gamma$ and $f_{\mathsf{m}}^{\mathrm{SC}}(\Delta) = \Delta$. We claim that either $\Delta \preceq^{\mathrm{SC}} \Gamma$ or $\Gamma \preceq^{\mathrm{SC}} \Delta$ holds. Seeking contradiction, deny the claim. Since the relation $\preceq^{\mathrm{SC}}$ is actually the set theoretic inclusion, there exist formulas φ and ψ with $\varphi \in \Gamma$, $\varphi \notin \Delta$, $\psi \in \Delta$ and $\psi \notin \Gamma$. Since $f_{\mathsf{m}}^{\mathrm{SC}}(\Gamma) = \Gamma$, $K_a\psi \notin \Gamma$ and $K_a\varphi \in \Gamma$ hold. Similarly, $K_a\varphi \notin \Delta$ and $K_a\psi \in \Delta$ hold. Since Θ is SC-saturated, $(K_a\varphi \supset K_a\psi) \vee (K_a\varphi \supset K_a\psi)$ is in Θ. The definition of saturation says either $K_a\varphi \supset K_a\psi \in \Theta$ or $K_a\psi \supset K_a\varphi \in \Theta$. Consequently, either $K_a\varphi \supset K_a\psi \in \Gamma$ or $K_a\psi \supset K_a\varphi \in \Delta$ holds. Each case leads to contradiction by deductive closedness of Γ and Δ. □

Lemma 3.8. *For an SC-saturated set of formulas Γ and the canonical model for sequential consistency M^{SC}, an equivalency $\varphi \in \Gamma \iff M^{SC}, \Gamma \vdash_{\mathrm{SC}} \varphi$ holds.*

This lemma can be proved in the same way as Lemma 2.13.

Theorem 3.9 (Strong completeness for sequential consistency)**.** *$\Gamma \vdash_{\mathrm{SC}} \varphi$ holds if $M \models \Gamma$ implies $M \models \varphi$ for every sequential model M.*

Proof. We show the contraposition: assuming $\Gamma \nvdash_{\mathrm{SC}} \varphi$, we show that there exists a sequential model M that satisfies $M \models \Gamma$ but not $M \models \varphi$. By Lemma 3.5, there is an SC-saturated set of formula Γ' with $\Gamma' \nvdash \varphi$ and $\Gamma' \supset \Gamma$. By Lemma 3.8, $M^{\mathrm{SC}}, \Gamma' \models \Gamma$ but not $M^{\mathrm{SC}}, \Gamma' \models \varphi$. □

Example Theorem In Introduction, we gave an example of theorems of $\vdash_{\mathrm{SC}}$: $(K_aK_{\mathsf{m}}K_a\varphi, K_bK\mathsf{m}K_b\psi) \supset (K_aK_b\psi \vee K_bK_a\varphi)$. We give a proof for this theorem in Figure 2.

4 Waitfree Computation

We define a class of formulas called waitfree assertions, which have a special finite model property (Theorem 4.5): if a waitfree assertion is consistent[2], there is a finite model of a special shape where the assertion is valid. The special shape mimics the scheduling of shared memory defined by Saks and Zaharoglou [25].

[2] A formula φ is consistent if and only if $\bot$ cannot be proved even if φ is added as an axiom.

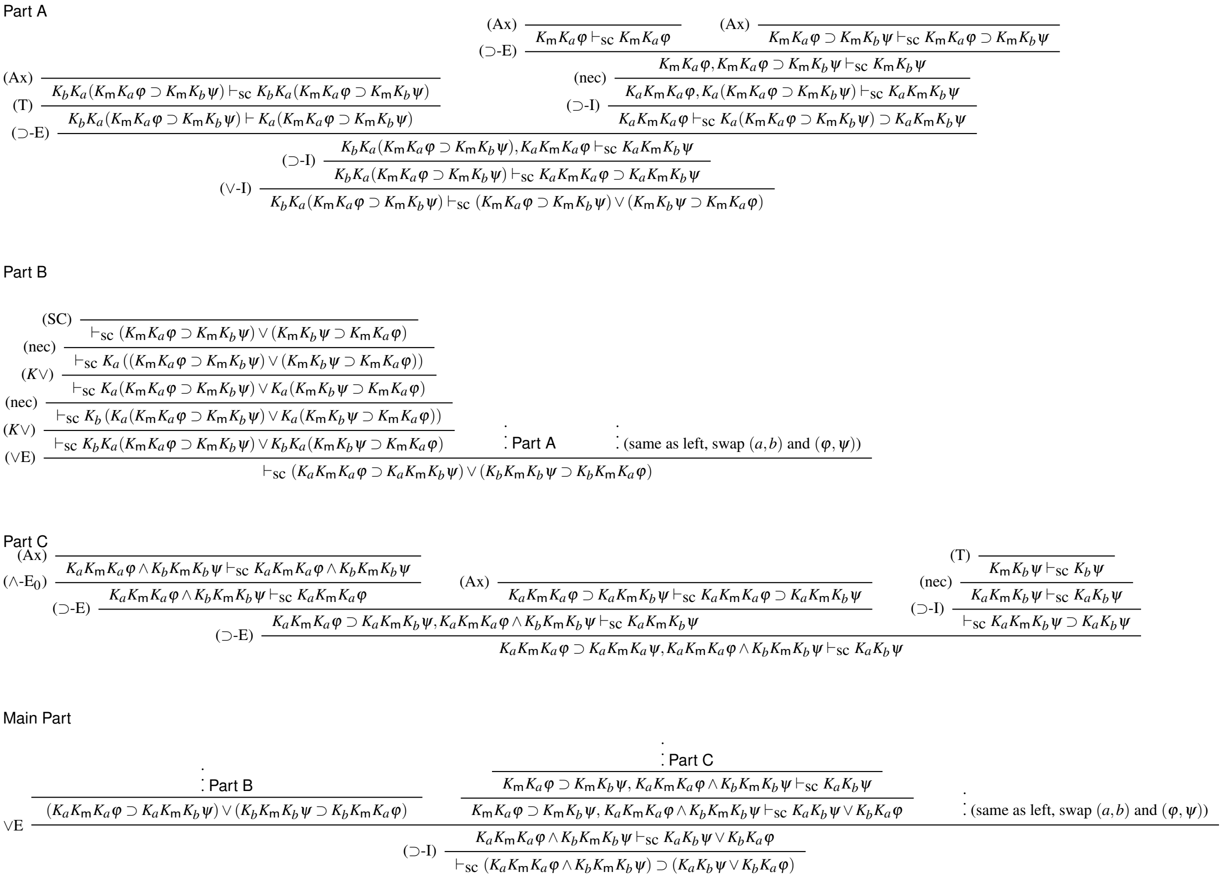
Part A
(Ax) $K_\mathsf{m}K_a\varphi \vdash_{\mathrm{sc}} K_\mathsf{m}K_a\varphi$
(Ax) $K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi$
(⊃-E) $K_\mathsf{m}K_a\varphi, K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_\mathsf{m}K_b\psi$
(nec) $K_aK_\mathsf{m}K_a\varphi, K_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_b\psi$
(⊃-I) $K_aK_\mathsf{m}K_a\varphi \vdash_{\mathrm{sc}} K_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \supset K_aK_\mathsf{m}K_b\psi$
(Ax) $K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vdash_{\mathrm{sc}} K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi)$
(T) $K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vdash K_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi)$
(⊃-E) $K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi), K_aK_\mathsf{m}K_a\varphi \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_b\psi$
(⊃-I) $K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi$
(∨-I) $K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vdash_{\mathrm{sc}} (K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee (K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi)$
Part B
(SC) $\vdash_{\mathrm{sc}} (K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee (K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi)$
(nec) $\vdash_{\mathrm{sc}} K_a((K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee (K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi))$
$(K\vee)$ $\vdash_{\mathrm{sc}} K_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee K_a(K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi)$
(nec) $\vdash_{\mathrm{sc}} K_b(K_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee K_a(K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi))$
$(K\vee)$ $\vdash_{\mathrm{sc}} K_bK_a(K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi) \vee K_bK_a(K_\mathsf{m}K_b\psi \supset K_\mathsf{m}K_a\varphi)$
Part A
(same as left, swap (a,b) and (φ,ψ))
(∨E) $\vdash_{\mathrm{sc}} (K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi) \vee (K_bK_\mathsf{m}K_b\psi \supset K_bK_\mathsf{m}K_a\varphi)$
Part C
(Ax) $K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi$
$(\wedge\text{-E}_0)$ $K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_a\varphi$
(Ax) $K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi$
(⊃-E) $K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi, K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_b\psi$
(T) $K_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_b\psi$
(nec) $K_aK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_b\psi$
(⊃-I) $\vdash_{\mathrm{sc}} K_aK_\mathsf{m}K_b\psi \supset K_aK_b\psi$
(⊃-E) $K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_a\psi, K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_b\psi$
Main Part
Part B
$(K_aK_\mathsf{m}K_a\varphi \supset K_aK_\mathsf{m}K_b\psi) \vee (K_bK_\mathsf{m}K_b\psi \supset K_bK_\mathsf{m}K_a\varphi)$
Part C
$K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi, K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_b\psi$
$K_\mathsf{m}K_a\varphi \supset K_\mathsf{m}K_b\psi, K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_b\psi \vee K_bK_a\varphi$
(same as left, swap (a,b) and (φ,ψ))
∨E $K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi \vdash_{\mathrm{sc}} K_aK_b\psi \vee K_bK_a\varphi$
(⊃-I) $\vdash_{\mathrm{sc}} (K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi) \supset (K_aK_b\psi \vee K_bK_a\varphi)$

Figure 2: A proof diagram for an example theorem $(K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi) \supset (K_aK_b\psi \vee K_bK_a\varphi)$ in $\vdash_{\mathrm{sc}}$.

Definition 4.1. *Assume there is a vector of atomic formulas $(I_p)_{p\in P}$. A waitfree protocol description φ is a formula of the form*

$$\varphi = \bigwedge_{a\in A} K_aK_\mathsf{m}K_a\cdots K_aI_a$$

where K_p and K_m appear alternatively in "$\cdots$". A waitfree task specification ψ is defined with the BNF:

$$\psi ::= K_p\psi \mid \psi\wedge\psi \mid \psi\vee\psi \mid I_p$$

where p stands for a process in P. A waitfree assertion is a formula $\varphi \supset \psi$ where φ is a waitfree protocol description and ψ is a waitfree task specification.

We are only interested in reasoning about a fixed protocol so that each process interacts with the memory for only finite times. In addition to this restriction, there is no process–process communication although there is process–memory communication so that a protocol can be described by a formula containing only a single process p and m. Finally, we forcefully decide that we are only interested in existence of knowledge at the end of protocols so that the requirement of a task can be represented in a positive formula. The formula $(K_aK_\mathsf{m}K_a\varphi \wedge K_bK_\mathsf{m}K_b\psi) \supset (K_aK_b\psi \vee K_bK_a\varphi)$ proved in Figure 2 is a waitfree assertion.

Definition 4.2. *A partial schedule $(\sigma_i)_{i\in I}$ is a finite sequence of subsets of P.*

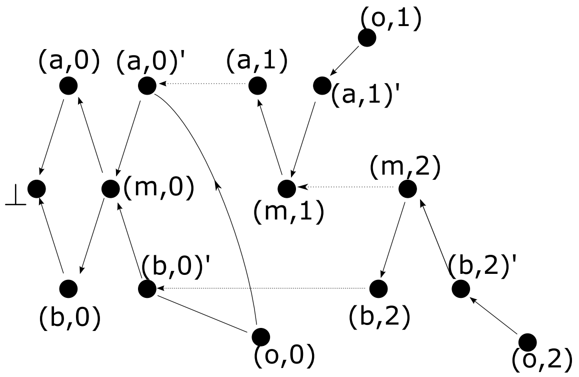

Figure 3: A model induced by the partial schedule $\{a,b\},\{a\},\{b\}$. A solid arrow pointing to (x,n) shows an f_x mapping. Dotted arrows show $\preceq$ relations. We omit inferable arrows and the valuation.

Definition 4.3. *For a process $p \in P$ and a partial schedule σ, $count_p(\sigma)$ is the cardinality $|\{i \in I \mid p \in \sigma_i\}|$.*

For a waitfree protocol description $\varphi = \bigwedge_{p\in P} K_p K_{\mathsf{m}} \cdots K_p I_p$, $count_p(\varphi)$ is the number of K_{m} occurrences in $K_p K_{\mathsf{m}} \cdots K_p I_p$.

A partial schedule σ is compatible to a waitfree protocol description φ if $count_p(\varphi) = count_p(\sigma)$ for any process $p \in P$.

Definition 4.4. *For a waitfree protocol description φ and a compatible partial schedule $(\sigma_i)_{i\in I}$, we define a waitfree schedule model $R(\varphi,\sigma) = \langle W, \preceq, (f_x)_{x\in A}, \rho\rangle$ as:*

- $W = \{(p,i) \in P \times \mathbb{N} \mid p \in \sigma_i\} \cup \{(p,i)' \in P \times \mathbb{N} \mid p \in \sigma_i\} \cup \{(\mathsf{m},i) \mid i \in I\} \cup \{(o,i) \mid i \in I\} \cup \{\bot\}$
- $(a,i) \preceq (m,i+1) \preceq (a,i)'$, $(x,j) \preceq (o,i)$ *if and only if* $j \leq i$, $\bot \preceq w$ *for all* $w \in W$, *and* $(x,j)' \preceq (o,i)$ *if and only if* $j \leq i$.
- $$f_a(w) = \begin{cases} \text{the least } (a,j) \text{ with } (a,j) \preceq w \text{ (if there exists such } (a,j)\text{)} \\ \quad \text{(the definition of } \preceq \text{ assures there is the least such } (a,j)\text{)}, \\ \bot \text{ (if such } (a,j) \text{ does not exist).} \end{cases}$$
- $\rho(I_a) = \{w \in W \mid (a,0) \preceq w\}$.

An example of a model induced by a partial schedule is shown in Figure 3.

Using the definitions above, we can state the logical characterization of waitfree communication.

Theorem 4.5 (Completeness for waitfree communication)**.** *Assume $\varphi \supset \psi$ is a waitfree assertion. The relation $\vdash_{SC} \varphi \supset \psi$ holds if the relation $R(\varphi,\sigma),(o,n) \models \psi$ holds for any compatible partial schedule σ where the state (o,n) is the last state of the waitfree model $R(\varphi,\sigma)$.*

To prove completeness, we only use special models called singleton models induced by a permutation of processes.

Definition 4.6. *For a set of processes P, we define $\mathsf{S}(P)$ to be the set of the permutations of P.*

Definition 4.7. *For $\pi \in \mathsf{S}(P)$ and $0 \leq k \leq |P|$, we define $SC(\pi,k)$ to be the set $\{K_{\mathsf{m}} K_a I_a \supset K_{\mathsf{m}} K_b I_b \mid$ there existi, jwithj $\leq i \leq k$, $\pi_i = a$, and $\pi_j = b\}$.*

Lemma 4.8. $\vdash_{\mathrm{SC}} \bigvee_{\pi\in\mathsf{S}(A)} SC(\pi,|P|)$ *holds.*

Proof. It suffices to use rule (SC) many times. □

Definition 4.9. *For a permutation π of P and a waitfree protocol description φ, we define a partial schedule $\sigma(\varphi,\pi)$ as*

$$\sigma(\varphi,\pi) = \overbrace{\pi_0,\cdots,\pi_0}^{count_{\pi_0}(\varphi)},\overbrace{\pi_1,\cdots,\pi_1}^{count_{\pi_1}(\varphi)},\cdots\cdots\cdots,\overbrace{\pi_n,\cdots,\pi_n}^{count_{\pi_n}(\varphi)}.$$

Definition 4.10. *A singleton model is a model of the form $R(\varphi,\sigma(\varphi,\pi))$. We abbreviate this to $R(\varphi,\pi)$.*

For a singleton model and an index $k \in I$, w_k denotes the minimum external observer state above all π_j states for $j < k$.

Definition 4.11. *For a waitfree protocol description $\varphi = \bigwedge_{a\in A}\overbrace{K_aK_{\mathsf{m}}K_a\cdots K_a}^{n_a}I_a$, we define the restriction $\varphi\restriction_{p,k} = \bigwedge_{a\in A\restriction_{p,k}}\overbrace{K_aK_{\mathsf{m}}K_a\cdots K_a}^{n_a}I_a$ where $A\restriction_{p,k} = \{a \mid p_j = a \text{ for some } j < k\}$.*

Lemma 4.12. $R(\varphi,\pi),(o,k) \models \psi \Longrightarrow SC(\pi,k) \vdash \varphi\restriction_{\pi,k} \supset \psi$.

Proof of Lemma 4.12. By induction on k.

(Case $k = 0$) We show a stronger proposition: $(o,0) \models \psi$ implies $f_{p_0}(o,0) \models \psi$, $\vdash \varphi\restriction_{p,0} \supset \psi$ and $\vdash \varphi\restriction_{p,0} \supset K_a\psi$ by inner induction on ψ.

- **(When ψ is an atomic formula P)** $P = I_{\pi_0}$ holds. Since $\varphi\restriction_{\pi,0} = K_{\pi_0}K_{\mathsf{m}}K_{\pi_0}\cdots K_{\mathsf{m}}K_{\pi_0}I_{\pi_0}$, $\vdash \varphi\restriction_{\pi,0} \supset K_{\pi_0}P$ holds. So, $SC(\pi,0) \vdash \varphi\restriction_{\pi,0} \supset K_{\pi_0}P$ holds. Consequently, $SC(\pi,0) \vdash \varphi\restriction_{\pi,0} \supset P$ also holds.
- **(When $\psi = \psi_0 \wedge \psi_1$ or $\psi_0 \vee \psi_1$)** Induction goes smoothly.
- **(When $\psi = K_a\psi'$)** Assume $(o,0) \models K_a\psi'$. Claim: $a = \pi_0$ holds. Seeking contradiction, assume $a \neq \pi_0$. That means $f_a((o,0)) = \bot$. However, waitfree task specification is satisfied at the state $\bot$. Contradiction. We have proved $a = \pi_0$. Using this, we can show that $f_a((o,0)) \models \psi'$ holds. By idempotency of f_a, $f_a(f_a((o,0))) \models \psi'$ holds. This means $f_a((o,0)) \models K_a\psi'$. Since $(o,0) \models \psi'$, by inner induction hypothesis, $\vdash \varphi\restriction_{\pi,0} \supset K_a\psi'_a$. By proof theoretic consideration, $\vdash \varphi\restriction_{\pi,0} \supset K_aK_a\psi'$ holds.

(Case $k = k' + 1$) Like the base case, we show a stronger proposition $(o,k) \models \psi \Leftrightarrow f_{\pi_k}((o,k)) \models \psi \Rightarrow SC(\pi,k) \vdash \varphi\restriction_{\pi,k} \supset \psi$ and $SC(\pi,k) \vdash \varphi\restriction_{\pi,k} \supset K_{\pi_k}\psi$, using inner induction on ψ.

- **(When $\psi = P$, an atomic formula)** Either $R(\varphi,\pi),w_{k'} \models P$ or $I_{\pi_k} = P$ holds. In the former case, by induction hypothesis. In the latter case, similarly as the base case.
- **(When $\psi = \psi_0 \wedge \psi_1$ or $\psi_0 \vee \psi_1$)** Induction goes smoothly.
- **(When $\psi = K_x\psi'$)** If $\pi_k \neq x$, $f_{\pi_k}((o,k)) \models K_x\psi'$ implies $(o,k') \models K_x\psi'$. By outer induction hypothesis, $SC(\pi,k') \vdash \varphi\restriction_{\pi,k'} \supset K_x\psi'$ and $SC(\pi,k') \vdash \varphi\restriction_{\pi,k'} \vdash \varphi\restriction_{\pi,k'} \supset K_x\psi'$ hold. Here, we can safely replace k' with k. If $\pi_k = x$, $(o,k) \models K_x\psi'$ imply $(o,k) \models \psi'$. By inner induction hypothesis, we obtain $SC(\pi,k) \vdash \varphi\restriction_{\pi,k} \supset K_x\psi'$. This also implies $SC(\pi,k) \vdash \varphi\restriction_{\pi,k} \supset K_xK_x\psi'$.

□

After showing this generalized lemma, proving Theorem 4.5 is easy.

Proof of Theorem 4.5. Since $R(\varphi,p),w_{|P|} \models \psi$, $SC(p,|P|) \vdash \varphi \supset \psi$. By Lemma 4.8, $\vdash_{\mathrm{SC}} \varphi \supset \psi$. □

Any models induced by a schedule is finite. For a waitfree assertion φ, it is decidable whether $\vdash_{\mathrm{SC}} \varphi$ holds or not.

4.1 Decidability of Solvability of Waitfree Task Specification

Definition 4.13. *A waitfree task specification ψ is solvable if there is such a waitfree protocol description φ that the relation $R(\varphi,\sigma),(o,n) \models \psi$ holds for any compatible partial schedule σ where the state (o,n) is the last state of the model $R(\varphi,\sigma)$.*

Fact. The set of solvable waitfree task specifications are recursively enumerable because the relation $\vdash_{\mathrm{SC}}$ is axiomatized.
Fact. The set of unsolvable waitfree task specifications are recursively enumerable because schedule-induced models are recursively enumerable.

These two facts imply that it is decidable whether a waitfree task specification is solvable or not. This does not contradict the undecidability of waitfreely solvable tasks by Gafni and Koutsoupias [9] because the undecidability proof utilizes tasks that cannot be expressed by waitfree task specifications. They use tasks involving consensus: the tasks involving making agreements among processes, where whether an output value is allowed or not depends on other processes' output values. Waitfree tasks specifications cannot describe such tasks.

5 Related Work

Ondrej Majer's Epistemic Logic with Relevant Agents [20] is similar to $\mathbf{K}\vee$ in that both logics have epistemic modalities and that both logics are not classical. However, the logic given in [20] contains only one modality K for knowledge. This implicitly assumes that there is a single agent, not multiple agents so that it is impossible for their logic to treat communication between multiple agents.

Many logics have both temporal and epistemic modalities. Ewald [7] proposes an intuitionistic logic with temporal modality. In Kobayashi and Yonezawa's logic [15], processes appear in formulas but time does not appear in formulas because time is implicit in the system of logic programming. This logic is different from $\mathbf{K}\vee$ in that this logic is based on linear logic and that their usage is logic programming.

6 Discussions

Waitfree Computation The Gödel Prize in 2004 was given to Herlihy and Shavit [12] and Saks and Zaharoglou [25]. This work was motivated by these works. Herlihy and Shavit [12] used subdivision of colored simplicial complex to model waitfree computation. Each vertex is colored by an agent. Each simplex contains vertices with distinct colors. A vertex may have an ancestor simplex called carrier. The minimum subset of $(S \cup V) \times (S \cup V)$ containing the ancestor relation and the relation $\in$ forms an order $\sqsubset$. We can define a partial $f_a : S \to S$ where S is the set of simplex in a simplicial complex by letting $f_a(s) = \{x\}$ where x is the maximum vertex below s (w.r.t. $\sqsubset$) whose color is a. When we add a bottom simplex $\perp$ and make f_a total, we can regard a simplicial complex as a model of $\mathbf{K}\vee$ as in an example (Figure 4).

Saks and Zaharoglou [25] use full-information protocols [30]. Even the shared variables remember the whole history. In every component, knowledge increases monotonically through time. This monotonicity suggests that their model can be analyzed effectively in Kripke models for intuitionistic logic. Saks and Zaharoglou [25] also suggest that "it will be worthwhile to explore the connection with the formal theory of distributed knowledge." This work is following their suggestion by treating waitfree communication in a formal way, especially using a logic with epistemic modalities.

Sequential Consistency or Linearizability Attiya and Welch [2] pointed out that sequential consistency [17] and linearizability [13] are often confused. We briefly make sure that the deduction system

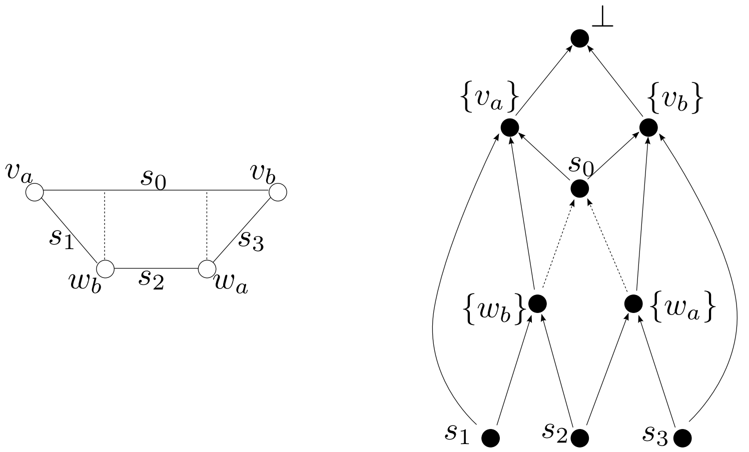

Figure 4: How subdivision of simplicial complexes is transformed into $\mathbf{K}\vee$ model. Left: A simplex $s_0 = \{v_a, v_b\}$ is subdivided into $s_1 = \{v_a, w_b\}, s_2 = \{w_a, w_b\}$ and $s_3 = \{w_a, v_b\}$. Right: $\mathbf{K}\vee$ frame obtained from the left subdivision.

$\vdash_{SC}$ does not characterize linearizability. Herlihy [13] stated that linearizability is a local property; in other words, when each memory object satisfies linearizability, the combined system also has linearizability. However, the axiom type *SC* is not local. To see that, assume there are two memory objects m and m'. The axiom type *SC* for m is $(K_{\mathsf{m}}\varphi \supset K_{\mathsf{m}}\psi) \vee (K_{\mathsf{m}}\psi \supset K_{\mathsf{m}}\varphi)$. The axiom type *SC* for m' is $(K_{\mathsf{m}'}\varphi \supset K_{\mathsf{m}'}\psi) \vee (K_{\mathsf{m}'}\psi \supset K_{\mathsf{m}'}\varphi)$. Even when both of these axiom types are available, the *mixed* axiom type $(K_{\mathsf{m}'}\varphi \supset K_{\mathsf{m}'}\psi) \vee (K_{\mathsf{m}'}\psi \supset K_{\mathsf{m}'}\varphi)$ is not derivable. This shows the characterized property is not local.

Other Consistency Models Steinke and Nutt [26] gave a lattice of consistency properties including: sequential consistency, causal consistency, processor consistency, PRAM consistency, cache consistency, slow consistency and local consistency. It is our future work modeling other consistency properties than sequential consistency.

Latency versus Throughput Our logic is more suitable for a situation where latency is more important than throughput. Since we consider time as the partial order of intuitionistic Kripke models, all knowledge must be preserved during time progress. Communication must be done in full-information manner (as in full-information protocols in [30]) because messages define the partial order. Our logic is advantageous when latency is important so that it is important to know how many message interactions are needed to accomplish a certain task. We plan to investigate network protocols with $\mathbf{K}\vee$.

Disjunction Distribution Over *K* Modality Since the semantics for modalities is defined by functions on Kripke frames, the disjunction distributes modalities in $\mathbf{K}\vee$. Kojima and Igarashi [16] avoids the distribution of modalities over disjunction by giving up functional modality. On the other hand, $\mathbf{K}\vee$ has distribution. We speculate that the difference comes from the different interpretations of modalities according to time: in [16] inner subformulas within the scope of the modality are interpreted in the future; while in $\mathbf{K}\vee$, inner subformulas within the scope of the modalities are interpreted in the past.

By translation of Suzuki [27], when A is a singleton set, $\mathbf{K}\vee$ corresponds to the intuitionistic predicate logic with singleton domain in the same manner the models of the logic L_3 of Ono [21] correspond to the models of intuitionistic predicate logic with constant domain. This fact suggests that the semantics of $\mathbf{K}\vee$ is very simple when there is only one agent. Simplicity was our aim at the beginning.

Acknowledgments The author thanks Masami Hagiya and Yoshihiko Kakutani for encouragements and valuable advice.

References

[1] N. Alechina, M. Mendler, V. de Paiva, and E. Hitter. Categorical and Kripke Semantics for Constructive S4 Modal Logic. In *Computer science logic: 15th international workshop, CSL 2001: proceedings*, pages 292–307. Springer, 2001.

[2] H. Attiya and J.L. Welch. Sequential consistency versus linearizability. *ACM Transactions on Computer Systems (TOCS)*, 12(2):122, 1994.

[3] P. Balbiani et al. 'Knowable' as 'known after an announcement'. *The Review of Symbolic Logic*, 1(03):305–334, 2008.

[4] A. Baltag, B. Coecke, and M. Sadrzadeh. Epistemic actions as resources. *Journal of Logic and Computation*, 17(3):555, 2007.

[5] P. Bieber and T. Onera-Cert. A logic of communication in hostile environment. In *Computer Security Foundations Workshop III, 1990. Proceedings*, pages 14–22, 1990.

[6] V. Costa and M. Benevides. Formalizing concurrent common knowledge as product of modal logics. *Logic Journal of IGPL*, 13(6):665, 2005.

[7] W.B. Ewald. Intuitionistic tense and modal logic. *The Journal of Symbolic Logic*, 51(1):166–179, 1986.

[8] D.M. Gabbay and V.B. Shehtman. Products of modal logics, part 1. *Logic journal of IGPL*, 6(1):73, 1998.

[9] E. Gafni and E. Koutsoupias. Three-processor tasks are undecidable. *SIAM Journal on Computing*, 28(3):970–983, 1999.

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

[11] M. Herlihy. Wait-free synchronization. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 13(1):124–149, 1991.

[12] M. Herlihy and N. Shavit. The topological structure of asynchronous computability. *Journal of the ACM (JACM)*, 46(6):858–923, 1999.

[13] M. Herlihy and J.M. Wing. Linearizability: A correctness condition for concurrent objects. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, 12(3):463–492, 1990.

[14] L. Jia and D. Walker. Modal Proofs as Distributed Programs(Extended Abstract). In *Programming languages and systems: 13th European Symposium on Programming, ESOP 2004: proceedings*, page 219. Springer, 2004.

[15] N. Kobayashi and A. Yonezawa. Asynchronous communication model based on linear logic. *Formal Aspects of Computing*, 7(2):113–149, 1995.

[16] K. Kojima and A. Igarashi. On constructive linear-time temporal logic. *Proc. of IMLA*, 8, 2008.

[17] L. Lamport. How to make a multiprocessor computer that correctly executes multiprocess progranm. *IEEE transactions on computers*, 100(28):690–691, 1979.

[18] C.J. Liau. Belief, information acquisition, and trust in multi-agent systems—A modal logic formulation. *Artificial Intelligence*, 149(1):31–60, 2003.

[19] R. Maddux. The equational theory of CA_3 is undecidable. *The Journal of Symbolic Logic*, 45(2):311–316, 1980.

[20] O. Majer and M. Peliš. Epistemic logic with relevant agents. In *The Logica Yearbook 2008*, pages 123–135. Kings College Publications, 2009.

[21] H. Ono. On some intuitionistic modal logics. *Publ. Res. Inst. Math. Sci.*, 13(3):687–722, 1977.

[22] D. Peleg. Communication in concurrent dynamic logic. *J. COMP. SYST. SCI.*, 35(1):23–58, 1987.

[23] J. Plaza. Logics of public communications. *Synthese*, 158(2):165–179, 2007.

[24] G. Plotkin and C Stirling. A framework for intuitionistic modal logics: extended abstract. In *TARK '86: Proceedings of the 1986 conference on Theoretical aspects of reasoning about knowledge*, pages 399–406. Morgan Kaufmann Publishers Inc., 1986.

[25] M. Saks and F. Zaharoglou. Wait-free k-set agreement is impossible: The topology of public knowledge. *SIAM journal on computing(Print)*, 29(5):1449–1483, 2000.

[26] R.C. Steinke and G.J. Nutt. A unified theory of shared memory consistency. *Journal of the ACM (JACM)*, 51(5):800–849, 2004.

[27] N.Y. Suzuki. Kripke bundles for intermediate predicate logics and Kripke frames for intuitionistic modal logics. *Studia Logica*, 49(3):289–306, 1990.

[28] A.S. Troelstra and D. van Dalen. *Constructivism in Mathematics: An Introduction: Vol.: 1*. North-Holland, 1988.

[29] J. van Benthem. The information in intuitionistic logic. *Synthese*, 167(2):251–270, 2009.

[30] T.Y.C. Woo and S.S. Lam. A lesson on authentication protocol design. *SIGOPS Oper. Syst. Rev.*, 28(3):24–37, 1994.