# Bimodal Reference Manual

*A Logic for Tense and Modality*

*Benjamin Brast-McKie*

www.benbrastmckie.com

— January 13, 2026 —

**Primary Reference:**

*"The Construction of Possible Worlds"*, Brast-McKie, (under review), 2025.

**Abstract**

This reference manual provides the formal specification of the Bimodal logic **TM** for tense and modality as implemented in the `ProofChecker` project. **TM** is a bimodal logic combining an S5 historical necessity operator with linear temporal operators for the past and future tenses. Soundness and the deduction theorem are established. Completeness is proven via the semantic canonical model approach: the Lindenbaum lemma, truth lemma, and weak completeness theorem are all proven. The key result `semantic_weak_completeness` demonstrates that validity implies derivability.

# Contents

# 1   Introduction

The bimodal logic **TM** combines S5 historical modal operators for necessity
($\Box$) and possibility ($\Diamond$) with linear temporal operators for past ($H$) and future
($G$). Implementing **TM** in Lean 4 provides a positive RL signal for training
AI systems to conduct verified reasoning about past and future contingency
in an interpreted formal language.

   The semantics is based on *task frames*, which extend Kripke frames with
temporal structure. A task frame consists of world states connected by a
*task relation* indexed by temporal durations. World histories are temporal
slices through a task frame, representing the possible evolution of a system.
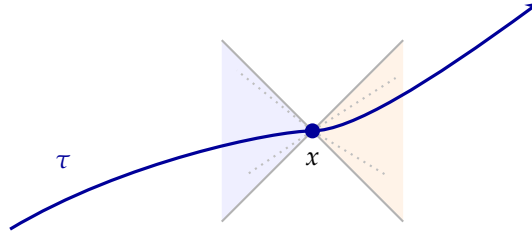


The diagram above illustrates the conceptual structure underlying TM logic.
The solid curve $\tau$ represents a single world history—a temporal sequence
of states. From any point $x$ along a history, the past and future light cones
contain all states that are modally accessible. The necessity operator $\Box$
quantifies over all possible histories, though we may often restrict to those
histories that pass through the world state $\tau(x)$. The temporal operators $H$
and $G$ quantify over past and future times which, given a particular history,
determines the range of past and future world states that history occupies
relative to a given time. These primitive operators may then be used to
define a host of combined operators of interest.

**Project Structure**

The Lean 4 implementation is in the `Bimodal/` directory:

- `Syntax/` – Defines the formula language with 6 primitive constructors
  and derived operators. **Complete.**

- `ProofSystem/` – Axioms (14 schemata) and inference rules (7 rules)
  forming a Hilbert-style proof system. **Complete.**

- `Semantics/` – Task frames model possible worlds; world histories
  model time; truth conditions define meaning. **Complete.**

3

- `Metalogic/` – Soundness theorem (proven: all axioms valid, rules preserve validity), deduction theorem (proven: enables assumption introduction), completeness via the semantic canonical model (Lindenbaum lemma proven, truth lemma proven, weak completeness proven), and decidability via tableau-based decision procedure (soundness and completeness proven). **Complete.**

- `Theorems/` – Perpetuity principles and modal theorems derived from the axiom system. **Partial.**

## 2  Syntax

### 2.1  Formulas

Formulas are defined inductively with six primitive constructors.

**Definition 2.1** (Formula)**.**  The type `Formula` is defined by:

$$\varphi, \psi ::= p \mid \bot \mid \varphi \to \psi \mid \Box\varphi \mid H\varphi \mid G\varphi$$

where $p$ ranges over propositional atoms (type `String`).

| Symbol | Name | Lean | Reading |
|---|---|---|---|
| $p, q, r$ | Atom | `atom s` | propositional atom |
| $\bot$ | Bottom | `bot` | falsity |
| $\varphi \to \psi$ | Implication | `imp` | "if $\varphi$ then $\psi$" |
| $\Box\varphi$ | Necessity | `box` | "necessarily $\varphi$" |
| $H\varphi$ | Always past | `all_past` | "always in the past, $\varphi$" |
| $G\varphi$ | Always future | `all_future` | "always in the future, $\varphi$" |

### 2.2  Derived Operators

The following operators are defined in terms of the primitives.

**Definition 2.2** (Propositional)**.**

$$\neg\varphi := \varphi \to \bot$$
$$\varphi \wedge \psi := \neg(\varphi \to \neg\psi)$$
$$\varphi \vee \psi := \neg\varphi \to \psi$$

| Symbol | Name | Lean | Reading |
|--------|------|------|---------|
| $\neg\varphi$ | Negation | neg | "not $\varphi$" |
| $\varphi \wedge \psi$ | Conjunction | and | "$\varphi$ and $\psi$" |
| $\varphi \vee \psi$ | Disjunction | or | "$\varphi$ or $\psi$" |

**Definition 2.3** (Modal).

$$\Diamond\varphi := \neg\Box\neg\varphi$$

| Symbol | Name | Lean | Reading |
|--------|------|------|---------|
| $\Diamond\varphi$ | Possibility | pos | "possibly $\varphi$" |

**Definition 2.4** (Temporal).

$$P\varphi := \neg H\neg\varphi$$
$$F\varphi := \neg G\neg\varphi$$
$$\triangle\varphi := H\varphi \wedge \varphi \wedge G\varphi$$
$$\triangledown\varphi := P\varphi \vee \varphi \vee F\varphi$$

| Symbol | Name | Lean | Reading |
|--------|------|------|---------|
| $P\varphi$ | Sometime past | some_past | "at some past time, $\varphi$" |
| $F\varphi$ | Sometime future | some_future | "at some future time, $\varphi$" |
| $\triangle\varphi$ | Always | always | "at all times, $\varphi$" |
| $\triangledown\varphi$ | Sometimes | sometimes | "at some time, $\varphi$" |

## 2.3 Temporal Duality

The swap_temporal function exchanges past and future operators.

**Definition 2.5** (Temporal Swap). The function $\langle S \rangle$ : Formula $\rightarrow$ Formula is defined by structural induction on formulas:

$$\langle S \rangle p = p$$
$$\langle S \rangle \bot = \bot$$
$$\langle S \rangle (\varphi \rightarrow \psi) = (\langle S \rangle \varphi \rightarrow \langle S \rangle \psi)$$
$$\langle S \rangle \Box\varphi = \Box\langle S \rangle \varphi$$
$$\langle S \rangle H\varphi = G\langle S \rangle \varphi$$
$$\langle S \rangle G\varphi = H\langle S \rangle \varphi$$

**Theorem 2.6** (Involution). $\langle S \rangle\langle S \rangle\varphi = \varphi$

# 3   Task Semantics

## 3.1   Task Frames

Task frames are the fundamental semantic structures for **TM**. They abstract from universal laws governing transitions between world states while still retaining the temporal duration for a transition to complete.

The following primitives are required to define a task frame:

| Primitive | Type | Description |
|---|---|---|
| $W$ | Type | World states |
| $D$ | Type | Temporal durations |
| $w \Rightarrow_x u$ | $W \to D \to W \to \text{Prop}$ | Task relation |

**Definition 3.1** (Task Frame). A **task frame** over temporal type $D$ is a triple $\mathcal{F} = (W, D, \Rightarrow)$ satisfying:

1. **Nullity**: For all $w : W$, we have $w \Rightarrow_0 w$.

2. **Compositionality**: For all $w, u, v : W$ and $x, y : D$, if $w \Rightarrow_x u$ and $u \Rightarrow_y v$, then $w \Rightarrow_{x+y} v$.

Nullity ensures that zero-duration tasks leave the world state unchanged. Compositionality ensures that executing tasks sequentially yields results consistent with a single task of combined duration.

## 3.2   World Histories

A world history is a function from times to world states that respects the task relation over a convex temporal domain. World histories represent possible paths through the space of world states.

**Definition 3.2** (Convex Domain). A domain $\text{dom} : D \to \text{Prop}$ is **convex** if whenever $a, c \in \text{dom}$ with $a \leq c$, every time $b$ with $a \leq b \leq c$ is also in dom. More precisely, for all $a, b, c : D$, if $\text{dom}(a)$ and $\text{dom}(c)$ and $a \leq b \leq c$, then $\text{dom}(b)$. Convexity ensures the domain has no temporal gaps.

**Definition 3.3** (World History). A **world history** in a task frame $\mathcal{F}$ is a dependent function $\tau : (x : D) \to \text{dom}(x) \to W$ where $\text{dom} : D \to \text{Prop}$ is a convex subset of $D$ and $\tau(x) \Rightarrow_{y-x} \tau(y)$ for all times $x, y : D$ with $\text{dom}(x)$, $\text{dom}(y)$, and $x \leq y$. We write $H_{\mathcal{F}}$ for all world histories over frame $\mathcal{F}$.

## 3.3 Task Models

A task model extends a task frame with an interpretation function that assigns truth values to sentence letters at world states. **Propositions** are subsets of $W$ representing instantaneous ways for the system to be. Sentence letters express propositions, which can be realized by zero or more world states. World states themselves are specific configurations of the total system at an instant.

**Definition 3.4** (Task Model). A **task model** defined over a frame $\mathcal{F}$ is a pair $\mathcal{M} = (\mathcal{F}, I)$ where the **interpretation function** $I : W \to \text{String} \to \text{Prop}$ assigns to each world state $w : W$ and sentence letter $p : \text{String}$ a truth value $I(w, p) : \text{Prop}$. We write $I(w, p)$ to indicate that sentence letter $p$ is true at $w$.

## 3.4 Truth Conditions

Truth is evaluated relative to a model $\mathcal{M}$ providing the interpretation, a world history $\tau$ representing a possible path through the space of world states, and a time $x : D$. Whereas the model fixes the interpretation of the language, the contextual parameters $\tau$ and $x$ determine the truth value of every sentence of the langauge.

**Definition 3.5** (Truth). For model $\mathcal{M}$, history $\tau \in H_{\mathcal{F}}$, and time $x : D$:

$$\mathcal{M}, \tau, x \vDash p \text{ iff } x \in \text{dom}(\tau) \text{ and } I(\tau(x), p)$$
$$\mathcal{M}, \tau, x \nvDash \bot$$
$$\mathcal{M}, \tau, x \vDash \varphi \to \psi \text{ iff } \mathcal{M}, \tau, x \nvDash \varphi \text{ or } \mathcal{M}, \tau, x \vDash \psi$$
$$\mathcal{M}, \tau, x \vDash \Box\varphi \text{ iff } \mathcal{M}, \sigma, x \vDash \varphi \text{ for all } \sigma \in H_{\mathcal{F}}$$
$$\mathcal{M}, \tau, x \vDash H\varphi \text{ iff } \mathcal{M}, \tau, y \vDash \varphi \text{ for all } y : D \text{ where } y < x$$
$$\mathcal{M}, \tau, x \vDash G\varphi \text{ iff } \mathcal{M}, \tau, y \vDash \varphi \text{ for all } y : D \text{ where } x < y$$

The modal operator $\Box$ quantifies over all world histories $\sigma : H_{\mathcal{F}}$ at the current time $x : D$. The temporal operators $H$ and $G$ quantify over all earlier and later times $y : D$.

## 3.5 Time-Shift

The time-shift operation is used to establish the **perpetuity principles**:

- P1: $\Box\varphi \to \triangle\varphi$ (what is necessary is always the case)

- P2: $\triangledown\varphi \to \Diamond\varphi$ (what is sometimes the case is possible)

It is natural to assume that whatever is necessary is always the case, or equivalently, whatever is sometimes the case is possible. Time-shift enables proofs of the bimodal axioms MF ($\Box\varphi \to \Box G\varphi$) and TF ($\Box\varphi \to G\Box\varphi$) which together imply the perpetuity principles.

**Definition 3.6** (Time-Shift). For $\tau, \sigma \in H_{\mathcal{F}}$ and $x, y : D$, world histories $\tau$ and $\sigma$ are **time-shifted from $y$ to $x$**, written $\tau \approx_y^x \sigma$, if and only if there exists an order automorphism $\bar{a} : D \to D$ where $y = \bar{a}(x)$, $\text{dom}_\sigma = \bar{a}^{-1}(\text{dom}_\tau)$, and $\sigma(z) = \tau(\bar{a}(z))$ for all $z \in \text{dom}_\sigma$.

Time-shifting preserves the essential structure of histories:

**Theorem 3.7** (Convexity Preservation). *If $\tau$ has a convex domain and $\tau \approx_y^x \sigma$, then $\sigma$ has a convex domain.*

**Theorem 3.8** (Task Preservation). *If $\tau$ respects the task relation and $\tau \approx_y^x \sigma$, then $\sigma$ respects the task relation.*

## 3.6   Logical Consequence and Validity

Logical consequence and validity are defined uniformly across all temporal types, frames, models, histories, and times.

**Definition 3.9** (Logical Consequence). A formula $\varphi$ is a **logical consequence** of $\Gamma$ (written $\Gamma \vDash \varphi$) just in case for every temporal type $D$ : Type, frame $\mathcal{F}$ : TaskFrame($D$), model $\mathcal{M}$ : TaskModel($\mathcal{F}$), history $\tau \in H_{\mathcal{F}}$, and time $x : D$, if $\mathcal{M}, \tau, x \vDash \psi$ for all $\psi \in \Gamma$, then $\mathcal{M}, \tau, x \vDash \varphi$.

**Definition 3.10** (Validity). A formula $\varphi$ is **valid** (written $\vDash \varphi$) just in case $\varphi$ is a logical consequence of the empty set: $\varnothing \vDash \varphi$. Equivalently, $\varphi$ is true at every model-history-time triple.

**Definition 3.11** (Satisfiability). A context $\Gamma$ is **satisfiable** in temporal type $D$ : Type if there exist a frame $\mathcal{F}$ : TaskFrame($D$), model $\mathcal{M}$ : TaskModel($\mathcal{F}$), history $\tau \in H_{\mathcal{F}}$, and time $x : D$ such that $\mathcal{M}, \tau, x \vDash \psi$ for all $\psi \in \Gamma$.

**Theorem 3.12** (Monotonicity). *If $\Gamma \subseteq \Delta$ and $\Gamma \vDash \varphi$, then $\Delta \vDash \varphi$.*

# 4   Proof Theory

## 4.1   Axiom Schemata

The TM proof system has 14 axiom schemata.

### 4.1.1   Propositional Axioms

**Axiom 4.1** (K - Distribution). $(\varphi \to (\psi \to \chi)) \to ((\varphi \to \psi) \to (\varphi \to \chi))$

**Axiom 4.2** (S - Weakening). $\varphi \to (\psi \to \varphi)$

**Axiom 4.3** (EFQ - Ex Falso Quodlibet). $\bot \to \varphi$

**Axiom 4.4** (Peirce). $((\varphi \to \psi) \to \varphi) \to \varphi$

### 4.1.2   Modal Axioms (S5)

**Axiom 4.5** (MT - Modal T). $\Box \varphi \to \varphi$

**Axiom 4.6** (M4 - Modal 4). $\Box \varphi \to \Box\Box \varphi$

**Axiom 4.7** (MB - Modal B). $\varphi \to \Box\Diamond \varphi$

**Axiom 4.8** (M5 - Modal 5 Collapse). $\Diamond\Box \varphi \to \Box \varphi$

**Axiom 4.9** (MK - Modal Distribution). $\Box(\varphi \to \psi) \to (\Box \varphi \to \Box \psi)$

### 4.1.3   Temporal Axioms

**Axiom 4.10** (TK - Temporal Distribution). $G(\varphi \to \psi) \to (G\varphi \to G\psi)$

**Axiom 4.11** (T4 - Temporal 4). $G\varphi \to GG\varphi$

**Axiom 4.12** (TA - Temporal A). $\varphi \to GP\varphi$

**Axiom 4.13** (TL - Temporal L). $\triangle \varphi \to GH\varphi$

### 4.1.4   Modal-Temporal Interaction

**Axiom 4.14** (MF - Modal-Future). $\Box \varphi \to \Box G\varphi$

**Axiom 4.15** (TF - Temporal-Future). $\Box \varphi \to G\Box \varphi$

| Axiom | Lean Constructor | Pattern |
|-------|------------------|---------|
| K | `Axiom.prop_k` | Distribution |
| S | `Axiom.prop_s` | Weakening |
| EFQ | `Axiom.ex_falso` | Explosion |
| Peirce | `Axiom.peirce` | Classical |
| MT | `Axiom.modal_t` | Reflexivity |
| M4 | `Axiom.modal_4` | Transitivity |
| MB | `Axiom.modal_b` | Symmetry |
| M5 | `Axiom.modal_5_collapse` | S5 collapse |
| MK | `Axiom.modal_k_dist` | Modal distribution |
| TK | `Axiom.temp_k_dist` | Temporal distribution |
| T4 | `Axiom.temp_4` | Temporal transitivity |
| TA | `Axiom.temp_a` | Connectedness |
| TL | `Axiom.temp_l` | Introspection |
| MF | `Axiom.modal_future` | Modal-future |
| TF | `Axiom.temp_future` | Temporal-modal |

## 4.2   Inference Rules

The proof system has 7 inference rules.

**Definition 4.16** (Axiom Rule)**.** If $\varphi$ matches an axiom schema, then $\Gamma \vdash \varphi$.

**Definition 4.17** (Assumption Rule)**.** If $\varphi \in \Gamma$, then $\Gamma \vdash \varphi$.

**Definition 4.18** (Modus Ponens)**.**

$$\frac{\Gamma \vdash \varphi \to \psi \qquad \Gamma \vdash \varphi}{\Gamma \vdash \psi}$$

**Definition 4.19** (Necessitation)**.**

$$\frac{\vdash \varphi}{\vdash \Box\varphi}$$

Applies only to theorems (empty context).

**Definition 4.20** (Temporal Necessitation)**.**

$$\frac{\vdash \varphi}{\vdash G\varphi}$$

Applies only to theorems (empty context).

**Definition 4.21** (Temporal Duality)**.**

$$\frac{\vdash \varphi}{\vdash \langle S \rangle \varphi}$$

Applies only to theorems (empty context).

**Definition 4.22** (Weakening)**.**

$$\frac{\Gamma \vdash \varphi \qquad \Gamma \subseteq \Delta}{\Delta \vdash \varphi}$$

| Rule | Lean Constructor | Context Requirement |
|------|-----------------|---------------------|
| Axiom | `DerivationTree.axiom` | Any |
| Assumption | `DerivationTree.assumption` | Any |
| Modus Ponens | `DerivationTree.modus_ponens` | Any |
| Necessitation | `DerivationTree.necessitation` | Empty only |
| Temp. Necessitation | `DerivationTree.temporal_necessitation` | Empty only |
| Temporal Duality | `DerivationTree.temporal_duality` | Empty only |
| Weakening | `DerivationTree.weakening` | Any |

## 4.3 Derivation Trees

Derivations are represented as inductive trees.

**Definition 4.23** (Derivation Tree)**.** `DerivationTree` $\Gamma$ $\varphi$ (written $\Gamma \vdash \varphi$) is an inductive type representing a derivation of $\varphi$ from context $\Gamma$.

**Definition 4.24** (Height)**.** The height of a derivation tree:

- Base cases (axiom, assumption): height 0

- Unary rules: height of subderivation + 1

- Modus ponens: max of both subderivations + 1

The height measure enables well-founded recursion in metalogical proofs.

## 4.4 Notation

| Notation | Lean |
|----------|------|
| $\Gamma \vdash \varphi$ | `DerivationTree` $\Gamma$ $\varphi$ |
| $\vdash \varphi$ | `DerivationTree [] ` $\varphi$ |

# 5 Metalogic

## 5.1 Soundness

The soundness theorem establishes that derivability implies semantic validity.

**Theorem 5.1** (Soundness). *If $\Gamma \vdash \varphi$ then $\Gamma \vDash \varphi$.*

The proof proceeds by induction on the derivation structure:

- **Axioms**: Each of the 14 axiom schemata is proven valid

- **Assumptions**: Assumed formulas are true by hypothesis

- **Modus ponens**: Validity preserved under implication elimination

- **Necessitation**: Valid formulas become necessarily valid

- **Temporal necessitation**: Valid formulas become always-future valid

- **Temporal duality**: Past-future swap preserves validity

- **Weakening**: Adding premises preserves semantic consequence

| Axiom Validity | Lean Theorem | Technique |
|---|---|---|
| `prop_k_valid` | Propositional K | Propositional reasoning |
| `prop_s_valid` | Propositional S | Propositional reasoning |
| `ex_falso_valid` | EFQ | Vacuous implication |
| `peirce_valid` | Peirce | Classical case analysis |
| `modal_t_valid` | MT | Reflexivity of accessibility |
| `modal_4_valid` | M4 | Transitivity of accessibility |
| `modal_b_valid` | MB | Symmetry of accessibility |
| `modal_5_collapse_valid` | M5 | S5 equivalence structure |
| `modal_k_dist_valid` | MK | Distribution |
| `temp_k_dist_valid` | TK | Temporal distribution |
| `temp_4_valid` | T4 | Transitivity of time |
| `temp_a_valid` | TA | Temporal connectedness |
| `temp_l_valid` | TL | Always implies recurrence |
| `modal_future_valid` | MF | Time-shift invariance |
| `temp_future_valid` | TF | Time-shift invariance |

The MF and TF axioms use time-shift invariance (via `WorldHistory.time_shift`) to relate truth at different times.

## 5.2   Deduction Theorem

**Theorem 5.2** (Deduction Theorem). *If $A :: \Gamma \vdash B$ then $\Gamma \vdash A \rightarrow B$.*

The proof uses well-founded induction on derivation height, handling each rule:

- **Axiom**: Use S axiom to weaken

- **Assumption**: Identity if same, S axiom if different

- **Modus ponens**: Use K axiom distribution

- **Weakening**: Case analysis on assumption membership

- **Modal/temporal rules**: Do not apply (require empty context)

## 5.3   Consistency

**Definition 5.3** (Consistent). A context $\Gamma$ is **consistent** if $\Gamma \nvdash \bot$.

**Definition 5.4** (Maximal Consistent). A context $\Gamma$ is **maximal consistent** if it is consistent and for all $\varphi \notin \Gamma$, the context $\varphi :: \Gamma$ is inconsistent.

## 5.4   Completeness

The completeness proof uses the semantic canonical model construction. World states are defined as equivalence classes of (history, time) pairs, making the truth lemma straightforward by construction.

**Lemma 5.5** (Lindenbaum). *Every consistent context can be extended to a maximal consistent context.*[1]

**Definition 5.6** (Semantic World State). A world state is an equivalence class of (history, time) pairs under the relation where two pairs are equivalent iff they denote the same underlying world state.[2]

**Definition 5.7** (Canonical Frame). The `SemanticCanonicalFrame` has:

- World states: Equivalence classes of history-time pairs over maximal consistent sets

- Times: Integers ($\mathbb{Z}$)

---

[1]This is proven as `set_lindenbaum` using Zorn's lemma.
[2]This is formalized as `SemanticWorldState`.

- Task relation: Defined via history existence (`SemanticTaskRelV2`)

The frame satisfies nullity and compositionality.

**Definition 5.8** (Canonical Valuation)**.** An atom $p$ is true at world state $[\tau, x]$ iff $p \in \mathsf{MCS}(\tau(x))$.

**Lemma 5.9** (Truth Lemma)**.** *In the semantic canonical model, $\varphi \in \mathsf{MCS}(\tau(x))$ iff $\varphi$ is true at world state $[\tau, x]$.*[3]

**Theorem 5.10** (Weak Completeness)**.** *If $\vdash \varphi$ then $\vdash \varphi$.*[4]

**Theorem 5.11** (Strong Completeness)**.** *If $\Gamma \vDash \varphi$ then $\Gamma \vdash \varphi$.*[5]

**Theorem 5.12** (Provable iff Valid)**.** $\vdash \varphi$ *iff* $\vdash \varphi$.[6]

**Definition 5.13** (Finite Model Property)**.** If a formula is satisfiable, it is satisfiable in a finite model bounded by the formula's modal and temporal depth.[7]

## 5.5   Proof Strategy

The semantic approach defines world states as equivalence classes of (history, time) pairs. Two pairs $(\tau_1, x_1)$ and $(\tau_2, x_2)$ are equivalent iff $\tau_1(x_1) = \tau_2(x_2)$ as underlying world states.

This construction offers key advantages:

- **Truth lemma**: Follows directly from the quotient construction. Membership in a maximal consistent set corresponds to truth by definition of the equivalence class.

- **Compositionality**: The task relation is defined via history existence. Two world states are related by a duration $d$ iff there exists a history passing through both states at times differing by $d$. History concatenation provides the compositionality proof.

- **Negation-completeness**: The semantic approach does not require proving negation-completeness of arbitrary locally consistent sets, a property that caused difficulties in the syntactic approach.

---

[3]This is proven as `semantic_truth_lemma_v2`.
[4]This is proven as `semantic_weak_completeness`.
[5]This is proven as `main_strong_completeness` with bridge sorries for the generalization.
[6]This is proven as `main_provable_iff_valid`, establishing completeness of **TM**.
[7]This is stated as `finite_model_property`.

The syntactic approach (building world states from negation-complete maximal consistent sets) is superseded. The deprecated code is archived in `Boneyard/` for historical reference.

## 5.6  Decidability

The decidability of TM bimodal logic is established via a tableau-based decision procedure.

**Theorem 5.14** (Decidability). *Validity in TM bimodal logic is decidable: for any formula $\varphi$, either $\vdash \varphi$ or $\neg \vdash \varphi$.*

**Theorem 5.15** (Decision Soundness). *If the decision procedure returns "valid" with proof $\pi$, then $\vdash \varphi$.*

The decision procedure operates as follows:

1. Try direct axiom proof (fast path optimization)

2. Try proof search with limited depth

3. Build tableau starting with signed formula $F(\varphi)$

4. If all branches close: formula is valid, extract proof

5. If open saturated branch: formula is invalid, extract countermodel

### 5.6.1  Tableau Structure

The tableau uses **signed formulas** with annotations:

- $T(\varphi)$: formula $\varphi$ is assumed true

- $F(\varphi)$: formula $\varphi$ is assumed false

**Expansion rules** are categorized as:

- **Propositional**: $T(\varphi \wedge \psi)$ splits, $F(\varphi \rightarrow \psi)$ splits, etc.

- **Modal**: $T(\Box\varphi)$ propagates to accessible worlds, $F(\Diamond\varphi)$ creates witness

- **Temporal**: $T(\triangle\varphi)$ propagates to future times, $F(\triangledown\varphi)$ creates witness

A branch **closes** when it contains both $T(\varphi)$ and $F(\varphi)$ for some formula. A branch is **saturated** when no expansion rules apply.

### 5.6.2 Complexity

| Measure | Complexity |
|---------|------------|
| Time | $O(2^n)$ where $n$ is formula size |
| Space | $O(n)$ |
| Class | PSPACE-complete |

### 5.6.3 Decision Result Types

The decision procedure returns one of three outcomes:

- `valid proof`: Formula is valid with derivation tree

- `invalid counter`: Formula is invalid with countermodel

- `timeout`: Resources exhausted before decision

### 5.6.4 Implementation Status

| Submodule | Status | Notes |
|-----------|--------|-------|
| SignedFormula | Complete | Sign, SignedFormula, Branch types |
| Tableau | Complete | Expansion rules |
| Closure | Complete | Branch closure detection |
| Saturation | Complete | Fuel-based termination |
| ProofExtraction | Partial | Axiom instances only |
| CountermodelExtraction | Complete | From open branches |
| DecisionProcedure | Complete | Main decide function |
| Soundness | Proven | `decide_sound` |
| Completeness | Partial | Requires Finite Model Property |

The soundness proof establishes that a "valid" result implies semantic validity. The completeness proof requires the Finite Model Property (FMP), which is stated but not yet fully formalized.

## 5.7 Implementation Status

| Component | Status | Lean |
|---|---|---|
| Soundness | Proven | `soundness` |
| Deduction Theorem | Proven | `deduction_theorem` |
| Lindenbaum Lemma | Proven | `set_lindenbaum` |
| Canonical Frame | Proven | `SemanticCanonicalFrame` |
| Truth Lemma | Proven | `semantic_truth_lemma_v2` |
| Weak Completeness | Proven | `semantic_weak_completeness` |
| Strong Completeness | Proven* | `main_strong_completeness` |
| Provable iff Valid | Proven | `main_provable_iff_valid` |
| Finite Model Property | Statement | `finite_model_property` |
| Decidability | Soundness | `decide_sound` |

* Strong completeness has bridge sorries for the connection between general validity and frame validity.

# 6 Theorems

## 6.1 Perpetuity Principles

The perpetuity principles establish deep connections between modal necessity ($\Box$) and temporal operators ($\triangle$, $\triangledown$).

**Theorem 6.1** (P1: Necessity Implies Always). $\vdash \Box\varphi \to \triangle\varphi$

**Theorem 6.2** (P2: Sometimes Implies Possible). $\vdash \triangledown\varphi \to \Diamond\varphi$

**Theorem 6.3** (P3: Necessity of Perpetuity). $\vdash \Box\varphi \to \Box\triangle\varphi$

**Theorem 6.4** (P4: Possibility of Occurrence). $\vdash \Diamond\triangledown\varphi \to \Diamond\varphi$

**Theorem 6.5** (P5: Persistent Possibility). $\vdash \Diamond\triangledown\varphi \to \triangle\Diamond\varphi$

**Theorem 6.6** (P6: Occurrent Necessity is Perpetual). $\vdash \triangledown\Box\varphi \to \Box\triangle\varphi$

All six perpetuity principles are fully proven in the Lean implementation.

| Principle | Lean Theorem | Key Lemmas |
|---|---|---|
| P1 | `perpetuity_1` | MF, TF, MT |
| P2 | `perpetuity_2` | Contraposition of P1 |
| P3 | `perpetuity_3` | P1, box_mono |
| P4 | `perpetuity_4` | Contraposition |
| P5 | `perpetuity_5` | modal_5, temporal K |
| P6 | `perpetuity_6` | P5, bridge lemmas |

## 6.2 Modal S5 Theorems

**Theorem 6.7** (T-Box-to-Diamond). $\vdash \Box\varphi \to \Diamond\varphi$

**Theorem 6.8** (Box Distributes Over Disjunction). $\vdash (\Box\varphi \vee \Box\psi) \to \Box(\varphi \vee \psi)$

**Theorem 6.9** (Box Preserves Contraposition). $\vdash \Box(\varphi \to \psi) \to \Box(\neg\psi \to \neg\varphi)$

**Theorem 6.10** (K Distribution for Diamond). $\vdash \Box(\varphi \to \psi) \to (\Diamond\varphi \to \Diamond\psi)$

**Theorem 6.11** (S5 Collapse). $\vdash \Diamond\Box\varphi \leftrightarrow \Box\varphi$

**Theorem 6.12** (Box-Conjunction). $\vdash \Box(\varphi \wedge \psi) \leftrightarrow (\Box\varphi \wedge \Box\psi)$

**Theorem 6.13** (Diamond-Disjunction). $\vdash \Diamond(\varphi \vee \psi) \leftrightarrow (\Diamond\varphi \vee \Diamond\psi)$

**Theorem 6.14** (S5 Diamond-Box to Truth). $\vdash \Diamond\Box\varphi \to \varphi$

**Theorem 6.15** (T-Box Consistency). $\vdash \Box(\varphi \wedge \neg\varphi) \to \bot$

## 6.3 Modal S4 Properties

The following S4 properties are derived from the TM axiom system.

**Theorem 6.16** (Modal 5). $\vdash \Diamond\varphi \to \Box\Diamond\varphi$

**Theorem 6.17** (Diamond 4). $\vdash \Diamond\Diamond\varphi \to \Diamond\varphi$

**Theorem 6.18** (Box Monotonicity). *If $\vdash \varphi \to \psi$ then $\vdash \Box\varphi \to \Box\psi$.*

**Theorem 6.19** (Diamond Monotonicity). *If $\vdash \varphi \to \psi$ then $\vdash \Diamond\varphi \to \Diamond\psi$.*

## 6.4   Propositional Theorems

**Theorem 6.20** (Identity). $\vdash \varphi \to \varphi$

**Theorem 6.21** (Double Negation Introduction). $\vdash \varphi \to \neg\neg\varphi$

**Theorem 6.22** (Double Negation Elimination). $\vdash \neg\neg\varphi \to \varphi$

**Theorem 6.23** (Contraposition). *If* $\vdash \varphi \to \psi$ *then* $\vdash \neg\psi \to \neg\varphi$.

**Theorem 6.24** (De Morgan Disjunction). $\vdash \neg(\varphi \vee \psi) \leftrightarrow (\neg\varphi \wedge \neg\psi)$

**Theorem 6.25** (De Morgan Conjunction). $\vdash \neg(\varphi \wedge \psi) \leftrightarrow (\neg\varphi \vee \neg\psi)$

## 6.5   Combinator Infrastructure

The combinator infrastructure provides Hilbert-style proof tools.

**Theorem 6.26** (B Combinator). $\vdash (B \to C) \to ((A \to B) \to (A \to C))$

**Theorem 6.27** (Implication Transitivity). *If* $\vdash A \to B$ *and* $\vdash B \to C$ *then* $\vdash A \to C$.

**Theorem 6.28** (Pairing). $\vdash A \to (B \to (A \wedge B))$

**Theorem 6.29** (Classical Merge). $\vdash (P \to Q) \to ((\neg P \to Q) \to Q)$

## 6.6   Generalized Necessitation

**Theorem 6.30** (Generalized Modal Necessitation). *If* $\Gamma \vdash \varphi$ *then* $\Box\Gamma \vdash \Box\varphi$ *where* $\Box\Gamma = [\Box\psi \mid \psi \in \Gamma]$.

**Theorem 6.31** (Generalized Temporal Necessitation). *If* $\Gamma \vdash \varphi$ *then* $G\Gamma \vdash G\varphi$ *where* $G\Gamma = [G\psi \mid \psi \in \Gamma]$.

## 6.7   Module Organization

| Module | Contents |
| --- | --- |
| `Perpetuity.lean` | P1-P6 principles |
| `ModalS5.lean` | S5 characteristic theorems (24 theorems) |
| `ModalS4.lean` | S4 properties (modal_5, diamond_4) |
| `Propositional.lean` | Classical propositional theorems |
| `Combinators.lean` | B, I, S combinators, imp_trans |
| `GeneralizedNecessitation.lean` | Context-level necessitation |

Total theorem count: 228 theorems/lemmas across the Theorems/ directory.

# 7 Notes

## 7.1 Implementation Status

| Component | Status | Notes |
|---|---|---|
| Syntax | Complete | 6 primitives, derived operators |
| Semantics | Complete | Task frames, world histories, truth |
| Proof System | Complete | 14 axioms, 7 inference rules |
| Soundness | Proven | All 14 axioms valid, 7 rules sound |
| Deduction Theorem | Proven | Well-founded recursion on height |
| Completeness | Proven (Semantic) | Lindenbaum, truth lemma, weak completeness |
| Decidability | Soundness Proven | Tableau-based, FMP for full completeness |
| Perpetuity Principles | Proven | P1-P6 all proven |

## 7.2 Discrepancy Notes

This section documents differences between the paper "The Construction of Possible Worlds" and the Lean implementation.

### 7.2.1 Terminology

- The paper uses "perpetuity principles" for P1-P6; the Lean code uses the same terminology.

- The paper's notation $\triangle$ and $\triangledown$ for "always" and "sometimes" is preserved in the Lean implementation as `always` and `sometimes`.

### 7.2.2   Axiom Naming

| Paper Name | Lean Name | Notes |
|---|---|---|
| MT (Modal T) | `Axiom.modal_t` | $\Box\varphi \to \varphi$ |
| M4 (Modal 4) | `Axiom.modal_4` | $\Box\varphi \to \Box\Box\varphi$ |
| MB (Modal B) | `Axiom.modal_b` | $\varphi \to \Box\Diamond\varphi$ |
| MK | `Axiom.modal_k_dist` | K distribution |
| TK | `Axiom.temp_k_dist` | Temporal K distribution |
| T4 | `Axiom.temp_4` | Temporal transitivity |
| TA | `Axiom.temp_a` | Temporal connectedness |
| TL | `Axiom.temp_l` | Temporal introspection |
| MF | `Axiom.modal_future` | Modal-future interaction |
| TF | `Axiom.temp_future` | Temporal-future interaction |

### 7.2.3   M5 Collapse Axiom

The implementation includes an explicit M5 collapse axiom (`Axiom.modal_5_collapse`):

$$\Diamond\Box\varphi \to \Box\varphi$$

This is derivable from the other S5 axioms (MB + M4) but is included as a primitive for proof convenience in the S5 collapse theorem.

### 7.2.4   Completeness Status

The paper proves completeness via canonical model construction. The Lean implementation establishes completeness via the semantic canonical model approach. The key results are:

- `set_lindenbaum`: Every consistent set extends to a maximal consistent set

- `semantic_truth_lemma_v2`: Membership corresponds to truth in the semantic model

- `semantic_weak_completeness`: Validity implies derivability

- `main_provable_iff_valid`: Derivability and validity coincide

The semantic approach defines world states as equivalence classes of history-time pairs, making the truth lemma straightforward by construction. Bridge sorries remain for connecting general validity to frame validity in strong completeness.

### 7.2.5 Decidability Implementation

The implementation includes a tableau-based decision procedure for validity that provides an alternative to the canonical model approach. The decidability module establishes that validity is decidable via constructive tableau expansion and branch closure. Soundness is proven: if the procedure returns "valid", the formula is semantically valid. Completeness requires the Finite Model Property (FMP), which is stated but not yet fully formalized. The FMP states that if a formula is satisfiable, it is satisfiable in a finite model. Full formalization of the FMP completes decidability.