

Reasoning with Many-Valued Interval Temporal Logic

Guillermo Badia¹ Carles Noguera² **Alberto Paparella³** Guido Sciavicco³
I. Eduard Stan⁴

November 28, 2024

¹School of Historical and Philosophical Inquiry, University of Queensland, Australia

²Department of Information Engineering and Mathematics, University of Siena, Italy

³Department of Mathematics and Computer Science, University of Ferrara, Italy

⁴Department of Informatics, Systems and Communication, University of Milano-Bicocca, Italy

Introduction

- Real-world scenarios involve uncertainty and temporal information
- Example:
 - Patient exhibits symptoms of “**depressed mood**” and “**insomnia**”
 - Symptoms:
 - Vary in intensity over time
 - Meet during certain intervals
 - Need to model:
 - Degrees of symptom severity
 - Temporal relationships between symptoms
- Traditional binary logic is insufficient for modeling such complexities

Limitations of Classical Logic

- **Binary truth values** (true or false)
- Cannot represent partial truths or degrees of certainty
- Difficult to model temporal relationships with uncertainty

- **Handle partial truths and uncertainty**
- Extend beyond the binary truth values of classical logic
- Examples:
 - Łukasiewicz logic
 - Gödel logic
 - Product logic
 - Intuitionistic logic

Interval Temporal Logic

- Focuses on reasoning over time intervals rather than time points
- Uses **Allen's interval relations**:
 - Before, During, Overlaps, Meets, etc.
- Useful for modeling temporal relationships between events

Many-Valued Interval Temporal Logic

- **Objective:** Model graded truths over time intervals
- Challenges:
 - Integrating many-valued truth with temporal relations
 - Developing reasoning systems to handle complexity

Our Contribution

- Developed a **sound and complete tableau system** for many-valued interval temporal logic
- Based on FL_{ew} -**algebras** to handle graded truth values
- Practical **implementation** for real-world applicability

Presentation Overview

Introduction

Preliminaries

Many-Valued Halpern and Shoham's Logic (MVHS)

Tableau System: Theory and Implementation

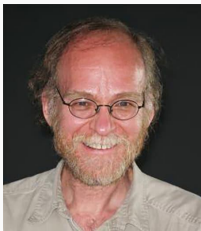
Experiments and Results

Conclusions and Future Work

Preliminaries

Halpern and Shoham's Interval Temporal Logic (HS)

- **HS** is a modal logic for reasoning about **time intervals**
- Uses modalities corresponding to **Allen's interval relations**
- Allows expression of **temporal relationships** between intervals
- Widely used in **temporal reasoning, representation, and planning** within AI



(a) Joseph Halpern.



(b) Yoav Shoham.

Allen's Interval Relations

relation	definition	example
after	$R_A([x, y], [w, z]) = = (y, w)$	
later	$R_L([x, y], [w, z]) = < (y, w)$	
begins	$R_B([x, y], [w, z]) = = (x, w) \wedge < (z, y)$	
ends	$R_E([x, y], [w, z]) = < (x, w) \wedge = (y, z)$	
during	$R_D([x, y], [w, z]) = < (x, w) \wedge < (z, y)$	
overlaps	$R_O([x, y], [w, z]) = < (x, w) \wedge < (w, y) \wedge < (y, z)$	

Table 1: Allen's interval relations.

Limitations of Classical HS

- HS is based on **classical (binary) logic**
 - Propositions and temporal relations are either **true** or **false**
- **Cannot handle:**
 - **Graded truths** (partial truth values)
 - **Uncertainty** or **imprecision**
- Inadequate for modeling **real-world scenarios** with partial information

- FL_{ew}-algebras [6]

$$\mathbf{A} = \langle A, \cap, \cup, \cdot, +, 0, 1 \rangle$$

are defined over **bounded integral commutative residuated lattices**

- A is the algebra's **domain**
- $\langle A, \cap, \cup, 0, 1 \rangle$ represents a **bounded complete lattice** with **upper bound 1** and **lower bound 0**
- $\langle A, \preceq \rangle$ corresponds to its **lattice-ordered set** ($\alpha \preceq \beta$ iff $\alpha = \alpha \cap \beta$)
- $\langle A, \cdot, 1 \rangle$ and $\langle A, +, 0 \rangle$ are **commutative monoids**, namely **t-norm** and **t-co-norm**, with both operations being monotone for \preceq (if $\gamma \preceq \alpha$ and $\delta \preceq \beta$, then $\gamma \cdot \delta \preceq \alpha \cdot \beta$ and $\gamma + \delta \preceq \alpha + \beta$)
- We also define an **implication** operation \hookrightarrow

$$\alpha \hookrightarrow \beta = \sup\{\gamma \mid \alpha \cdot \gamma \preceq \beta\}.$$

- \mathbf{A} is a **chain** if \preceq is a total order; **standard** if $A = [0, 1] \subset \mathbb{R}$; **finite** if A is finite. We will focus on **finite FL_{ew}-algebras**.

Example: Simple FL_{ew} -Algebra (G3)

- **Set of truth values:** $A = \{0, \alpha, 1\}$ with $0 \prec \alpha \prec 1$
- **Operations defined as:**

- **t-norm (\cdot):**

$$a \cdot b = \min(a, b)$$

- **t-co-norm ($+$):**

$$a + b = \max(a, b)$$

- **implication (\hookrightarrow):**

$$a \hookrightarrow b = \begin{cases} 1 & \text{if } a \preceq b \\ b & \text{otherwise} \end{cases}$$

- **Calculation example:**

- $\alpha \cdot 1 = \min(\alpha, 1) = \alpha$
- $\alpha + 1 = \max(\alpha, 1) = 1$
- $\alpha \hookrightarrow 1 = 1$ since $\alpha \prec 1$

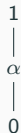


Figure 2: Lattice representing the order between the values in the designated FL_{ew} -algebra.

Relation to Other Algebras

- FL_{ew} -algebras encompass several known algebras:
 - Gödel algebras
 - MV algebras
 - Product algebras
 - Heyting algebras
- **Generalization** allows for unified treatment
- **Visual hierarchy**:

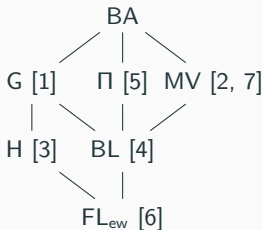


Figure 3: A partial taxonomy of well-known truth value algebras.

Many-Valued Halpern and Shoham's Logic (MVHS)

- **Propositional letters:** p, q, r, \dots
- **Truth constants:** $\alpha \in A$ (elements of the FL_{ew} -algebra)
- **Logical binary connectives**
 - Conjunction: \wedge
 - Disjunction: \vee
 - Implication: \rightarrow
- **Unary modalities**
 - $\langle X \rangle \varphi$ (there exists an interval related by X where φ holds)
 - $[X] \varphi$ (for all intervals related by X , φ holds)
- **Formulas** are built inductively using these elements

Many-Valued Linear Orderings and Strict Intervals

- Many-valued linear orders

$$\tilde{\mathbb{D}} = \langle D, \tilde{<}, \tilde{=} \rangle$$

- D is the **domain**
- $\tilde{<}, \tilde{=}: D \times D \rightarrow A$ are two functions mapping pairs of domain values to A of a FL_{ew} -algebra \mathbf{A} satisfying
 1. $\tilde{=}(x, y) = 1$ iff $x = y$
 2. $\tilde{=}(x, y) = \tilde{=}(y, x)$
 3. $\tilde{<}(x, x) = 0$
 4. $\tilde{<}(x, z) \succeq \tilde{<}(x, y) \cdot \tilde{<}(y, z)$
 5. if $\tilde{<}(x, y) \succ 0$ and $\tilde{<}(y, z) \succ 0$, then $\tilde{<}(x, z) \succ 0$
 6. if $\tilde{<}(x, y) = 0$ and $\tilde{<}(y, x) = 0$, then $\tilde{=}(x, y) = 1$
 7. if $\tilde{=}(x, y) \succ 0$, then $\tilde{<}(x, y) \prec 1$
- Many-valued strict intervals $\mathbb{I}(\tilde{\mathbb{D}}) = \{[x, y] \mid \tilde{<}(x, y) \succ 0\}$

Many-Valued Allen's Relations

relation	definition	example
after	$R_A([x, y], [w, z]) = = (y, w)$	
later	$R_L([x, y], [w, z]) = < (y, w)$	
begins	$R_B([x, y], [w, z]) = = (x, w) \wedge < (z, y)$	
ends	$R_E([x, y], [w, z]) = < (x, w) \wedge = (y, z)$	
during	$R_D([x, y], [w, z]) = < (x, w) \wedge < (z, y)$	
overlaps	$R_O([x, y], [w, z]) = < (x, w) \wedge < (w, y) \wedge < (y, z)$	

Table 2: Allen's interval relations.

Many-Valued Allen's Relations

relation	definition	example
after	$\tilde{R}_A([x, y], [w, z]) = \tilde{\equiv}(y, w)$	
later	$\tilde{R}_L([x, y], [w, z]) = \tilde{<}(y, w)$	
begins	$\tilde{R}_B([x, y], [w, z]) = \tilde{\equiv}(x, w) \cdot \tilde{<}(z, y)$	
ends	$\tilde{R}_E([x, y], [w, z]) = \tilde{<}(x, w) \cdot \tilde{\equiv}(y, z)$	
during	$\tilde{R}_D([x, y], [w, z]) = \tilde{<}(x, w) \cdot \tilde{<}(z, y)$	
overlaps	$\tilde{R}_O([x, y], [w, z]) = \tilde{<}(x, w) \cdot \tilde{<}(w, y) \cdot \tilde{<}(y, z)$	

Table 2: Many-valued Allen's interval relations.

Semantics of MVHS

- **Many-valued interval models** $\tilde{M} = \langle \mathbb{I}(\tilde{\mathbb{D}}), \tilde{V} \rangle$
 - **Valuation function** \tilde{V} : Assigns truth values from A to formulas at intervals
- **Atoms:**
 - $\tilde{V}(p, [x, y]) \in A$
 - $\tilde{V}(\alpha, [x, y]) = \alpha \in A$
- **Logical connectives:**
 - $\tilde{V}(\varphi \wedge \psi, [x, y]) = \tilde{V}(\varphi, [x, y]) \cdot \tilde{V}(\psi, [x, y])$
 - $\tilde{V}(\varphi \vee \psi, [x, y]) = \tilde{V}(\varphi, [x, y]) + \tilde{V}(\psi, [x, y])$
 - $\tilde{V}(\varphi \rightarrow \psi, [x, y]) = \tilde{V}(\varphi, [x, y]) \hookrightarrow \tilde{V}(\psi, [x, y])$
- **Modalities:**
 - $\tilde{V}(\langle X \rangle \varphi, [x, y]) = \bigcup_{[w, z] \in \mathbb{I}(\tilde{\mathbb{D}})} \left(\tilde{R}_X([x, y], [w, z]) \cdot \tilde{V}(\varphi, [w, z]) \right)$
 - $\tilde{V}([X] \varphi, [x, y]) = \bigcap_{[w, z] \in \mathbb{I}(\tilde{\mathbb{D}})} \left(\tilde{R}_X([x, y], [w, z]) \hookrightarrow \tilde{V}(\varphi, [w, z]) \right)$

- A formula φ is α -**satisfied** at $[x, y]$ in \tilde{M} if and only if

$$\tilde{V}(\varphi, [x, y]) \succeq \alpha$$

- A formula is α -**satisfiable** if and only if an interval exists in a multi-valued interval model where is α -satisfied
- A formula is α -**valid** if and only if it is α -satisfiable at every interval in every multi-valued interval model
- A formula is **valid** if and only if it is 1-valid

Application Example: Medical Diagnosis

- **Scenario:**

- Patient exhibits symptoms:

- “Depressed mood” (p)
- “Insomnia” (q)

- Symptoms vary in intensity over intervals

- Algebra’s domain $A = [0, 1] \subset \mathbb{R}$

- **Goal:** Determine the degree to which an interval of “depressed mood” meets a period of “insomnia”

- **Formula:**

$$\varphi = p \wedge \langle A \rangle q$$

Evaluating the Example

- **Assign truth values:**

- $\tilde{V}(p, [x, y]) = 0.7$
- $\tilde{V}(q, [w, z]) = 0.8$
- $\tilde{R}_A([x, y], [w, z]) = \tilde{\Xi}(y, w) = 0.9$

- **Then:**

$$\begin{aligned}\tilde{V}(\varphi, [x, y]) &= \tilde{V}(p \wedge \langle A \rangle q, [x, y]) \\ &= \tilde{V}(p, [x, y]) \cdot \tilde{V}(\langle A \rangle q, [x, y]) \\ &= 0.7 \cdot \tilde{R}_A([x, y], [w, z]) \cdot \tilde{V}(q, [w, z]) \\ &= 0.7 \cdot 0.9 \cdot 0.8 \\ &= 0.504\end{aligned}$$

- **Result:**

$$\tilde{V}(\varphi, [x, y]) = 0.504$$

- **Interpretation:** It is not always the case that a period of “depressed mood” is followed by a period of “insomnia,” but we can say that it happens in a non-negligible manner
- **Benefits of MVHS:**
 - Models **graded truths** over time intervals
 - Handles **uncertainty and partial information**
 - Provides **quantitative insights** into temporal relationships
 - Applicable to various **real-world scenarios** in AI

Tableau System: Theory and Implementation

Need for a Tableau System

- **Challenges in reasoning** with MVHS
 - Many-valued truth values increase the complexity
 - Temporal modalities over intervals add to the intricacy
- **Objective**
 - Develop a systematic method for determining **satisfiability** and **validity**
 - Ensure **soundness** and **completeness**
- **Solution: Fitting's style tableau system** adapted for MVHS over finite FL_{ew} -algebras

Overview of the Tableau Structure

- **Tree-like structure** with nodes and branches

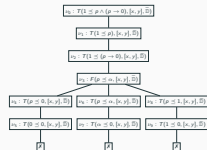


Figure 4: An example.

- Each node is associated with a **decoration**

$$Q(\beta \preceq \psi, [x, y], \tilde{\mathbb{D}}) \quad \text{or} \quad Q(\psi \preceq \beta, [x, y], \tilde{\mathbb{D}})$$

- Q is a **truth judgment** - either T (true) or F (false)
- $\beta \in A$ is a truth value from the FL_{ew} -algebra
- $\psi \in \text{sub}(\varphi)$ is a sub-formula of φ
- $[x, y]$ is an interval
- $\tilde{\mathbb{D}}$ is a many-valued linear order
- Branches represent possible **evaluations** and are associated with a finite many-valued linear order

Overview of the Tableau Procedure

- **Purpose:** Systematically explore possible valuations to determine:
 - **Satisfiability:** If starting from $T(\alpha \preceq \varphi)$ it finds an open branch (SAT-tableau), or
 - **Validity:** If starting from $F(\alpha \preceq \varphi)$ it closes all branches (VAL-tableau)
- **Expansion and branching:** Systematically apply expansion rules to generate new nodes
- **Closure:** Close branches that contain contradictions using branch closing rules
- **Termination**
 - If all branches are closed, the formula is unsatisfiable
 - If at least one open branch remains, a satisfying model exists

Expansion Rules: Reverse

$$(T \succeq) \frac{T(\beta \preceq \psi, [x, y], \tilde{\mathbb{D}})}{F(\psi \preceq \gamma, [x, y], c(B))}$$

where $\beta \neq 0$ and γ is any maximal
element not above β , i.e., $\gamma \not\preceq \beta$

$$(F \succeq) \frac{F(\beta \preceq \psi, [x, y], \tilde{\mathbb{D}})}{T(\psi \preceq \gamma_1, [x, y], c(B)) \mid \dots \mid T(\psi \preceq \gamma_n, [x, y], c(B))}$$

where $\beta \neq 0$ and $\gamma_1, \dots, \gamma_n$ are all maximal
elements not above β , i.e., $\gamma_1, \dots, \gamma_n \not\preceq \beta$

Figure 5: Reverse rules (1).

Expansion Rules: Reverse

$$(T \preceq) \frac{T(\psi \preceq \beta, [x, y], \tilde{\mathbb{D}})}{F(\gamma \preceq \psi, [x, y], c(B))}$$

where $\beta \neq 1$ and γ is any minimal
element not below β , i.e., $\gamma \not\preceq \beta$

$$(F \preceq) \frac{F(\psi \preceq \beta, [x, y], \tilde{\mathbb{D}})}{T(\gamma_1 \preceq \psi, [x, y], c(B)) \mid \dots \mid T(\gamma_n \preceq \psi, [x, y], c(B))}$$

where $\beta \neq 1$ and $\gamma_1, \dots, \gamma_n$ are all minimal
elements not below β , i.e., $\gamma_1, \dots, \gamma_n \not\preceq \beta$

Figure 6: Reverse rules (2).

Expansion Rules: Propositional

$$(T\wedge) \frac{T(\beta \preceq (\psi \wedge \xi), [x, y], \tilde{\mathbb{D}})}{T(\beta_1 \preceq \psi, [x, y], c(B)) \mid \dots \mid T(\beta_n \preceq \psi, [x, y], c(B)) \mid T(\gamma_1 \preceq \xi, [x, y], c(B)) \mid \dots \mid T(\gamma_n \preceq \xi, [x, y], c(B))}$$

where $\beta \neq 0$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta \preceq \beta_i \cdot \gamma_i$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta \preceq \beta'_i \cdot \gamma'_i$, $\beta'_i \preceq \beta_i$ and $\gamma'_i \preceq \gamma_i$.

$$(F\wedge) \frac{F(\beta \preceq (\psi \wedge \xi), [x, y], \tilde{\mathbb{D}})}{T(\psi \preceq \beta_1, [x, y], c(B)) \mid \dots \mid T(\psi \preceq \beta_n, [x, y], c(B)) \mid T(\xi \preceq \gamma_1, [x, y], c(B)) \mid \dots \mid T(\xi \preceq \gamma_n, [x, y], c(B))}$$

where $\beta \neq 0$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta \not\preceq \beta_i \cdot \gamma_i$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta \not\preceq \beta'_i \cdot \gamma'_i$, $\beta_i \preceq \beta'_i$ and $\gamma_i \preceq \gamma'_i$.

Figure 7: Propositional rules (1).

Expansion Rules: Propositional

$$(T\vee) \frac{T((\psi \vee \xi) \preceq \beta, [x, y], \mathbb{D})}{T(\psi \preceq \beta_1, [x, y], c(B)) \mid \dots \mid T(\psi \preceq \beta_n, [x, y], c(B)) \mid T(\xi \preceq \gamma_1, [x, y], c(B)) \mid \dots \mid T(\xi \preceq \gamma_n, [x, y], c(B))}$$

where $\beta \neq 1$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta_i + \gamma_i \preceq \beta$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta'_i + \gamma'_i \preceq \beta$, $\beta_i \preceq \beta'_i$ and $\gamma_i \preceq \gamma'_i$.

$$(F\vee) \frac{F((\psi \vee \xi) \preceq \beta, [x, y], \mathbb{D})}{T(\beta_1 \preceq \psi, [x, y], c(B)) \mid \dots \mid T(\beta_n \preceq \psi, [x, y], c(B)) \mid T(\gamma_1 \preceq \xi, [x, y], c(B)) \mid \dots \mid T(\gamma_n \preceq \xi, [x, y], c(B))}$$

where $\beta \neq 1$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta_i + \gamma_i \not\preceq \beta$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta'_i + \gamma'_i \not\preceq \beta$, $\beta'_i \preceq \beta_i$ and $\gamma'_i \preceq \gamma_i$.

Figure 8: Propositional rules (2).

Expansion Rules: Propositional

$$(T \hookrightarrow) \frac{T(\beta \preceq (\psi \hookrightarrow \xi), [x, y], \tilde{\mathbb{D}})}{T(\psi \preceq \beta_1, [x, y], c(B)) \mid \dots \mid T(\psi \preceq \beta_n, [x, y], c(B)) \mid T(\gamma_1 \preceq \xi, [x, y], c(B)) \mid \dots \mid T(\gamma_n \preceq \xi, [x, y], c(B))}$$

where $\beta \neq 0$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta \preceq \beta_i \hookrightarrow \gamma_i$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta \preceq \beta'_i \hookrightarrow \gamma'_i$, $\beta_i \preceq \beta'_i$ and $\gamma'_i \preceq \gamma_i$.

$$(F \hookrightarrow) \frac{F(\beta \preceq (\psi \hookrightarrow \xi), [x, y], \tilde{\mathbb{D}})}{T(\beta_1 \preceq \psi, [x, y], c(B)) \mid \dots \mid T(\beta_n \preceq \psi, [x, y], c(B)) \mid T(\xi \preceq \gamma_1, [x, y], c(B)) \mid \dots \mid T(\xi \preceq \gamma_n, [x, y], c(B))}$$

where $\beta \neq 0$, $(\beta_i, \gamma_i) \in \mathbf{A} \times \mathbf{A}$ so that $\beta \not\preceq \beta_i \hookrightarrow \gamma_i$ and there is no other $(\beta'_i, \gamma'_i) \in \mathbf{A} \times \mathbf{A}$ such that $\beta \not\preceq \beta'_i \hookrightarrow \gamma'_i$, $\beta'_i \preceq \beta_i$ and $\gamma_i \preceq \gamma'_i$.

Figure 9: Propositional rules (3).

Expansion Rules: Modalities

$$\begin{aligned}
 (T\Box) \quad & \frac{T(\beta \preceq [X]\psi, [x, y], \tilde{\mathbb{D}})}{T((\beta \cdot \gamma_1) \preceq \psi, [z_1, t_1], c(B))} \\
 & \quad \dots \\
 & \quad T((\beta \cdot \gamma_n) \preceq \psi, [z_n, t_n], c(B)) \\
 & \quad T(\beta \preceq [X]\psi, [x, y], c(B)) \\
 & \text{where } \gamma_i = \tilde{R}_X([x, y], [z_i, t_i]), [z_i, t_i] \in o(c(B)), \\
 & \quad \gamma_i \succ 0, \text{ and } \beta \cdot \gamma_i \neq 0 \\
 \\
 (F\Box) \quad & \frac{F(\beta \preceq [X]\psi, [x, y], \tilde{\mathbb{D}})}{F((\beta \cdot \gamma_1) \preceq \psi, [z_1, t_1], c(B)) \mid \dots \mid F((\beta \cdot \gamma_n) \preceq \psi, [z_n, t_n], c(B))} \\
 & \text{where } \gamma_i = \tilde{R}_X([x, y], [z_i, t_i]), [z_i, t_i] \in o(c(B)) \cup n(c(B)), \\
 & \quad \gamma_i \succ 0, \text{ and } \beta \cdot \gamma_i \neq 0
 \end{aligned}$$

Figure 10: Temporal rules (1).

Expansion Rules: Modalities

$$(T\Diamond) \frac{T(\langle X \rangle \psi \preceq \beta, [x, y], \tilde{\mathbb{D}})}{T((\psi \preceq (\gamma_1 \hookrightarrow \beta), [z_1, t_1], c(B)) \dots T(\psi \preceq (\gamma_n \hookrightarrow \beta), [z_n, t_n], c(B)) T(\langle X \rangle \psi \preceq \beta, [x, y], c(B))}$$

where $\gamma_i = \tilde{R}_X([x, y], [z_i, t_i])$, $[z_i, t_i] \in o(c(B))$,
 $\gamma_i \succ 0$, and $\gamma_i \hookrightarrow \beta \neq 1$

$$(F\Diamond) \frac{F(\langle X \rangle \psi \preceq \beta, [x, y], \tilde{\mathbb{D}})}{F(\psi \preceq (\gamma_1 \hookrightarrow \beta), [z_1, t_1], c(B)) \mid \dots \mid F(\psi \preceq (\gamma_n \hookrightarrow \beta), [z_n, t_n], c(B))}$$

where $\gamma_i = \tilde{R}_X([x, y], [z_i, t_i])$, $[z_i, t_i] \in o(c(B)) \cup n(c(B))$,
 $\gamma_i \succ 0$, and $\gamma_i \hookrightarrow \beta \neq 1$

Figure 11: Temporal rules (2).

Branch Closing Rules

$$(X1) \frac{T(\beta \preceq \gamma, [x, y], \tilde{\mathbb{D}})}{\mathbf{X}}$$

where $\beta \not\preceq \gamma$

$$(X2) \frac{F(\beta \preceq \gamma, [x, y], \tilde{\mathbb{D}})}{\mathbf{X}}$$

where $\beta \neq 0$, $\gamma \neq 1$, and $\beta \preceq \gamma$

$$(X3) \frac{F(0 \preceq \psi, [x, y], \tilde{\mathbb{D}})}{\mathbf{X}}$$

$$(X4) \frac{F(\psi \preceq 1, [x, y], \tilde{\mathbb{D}})}{\mathbf{X}}$$

$$(X5) \frac{\frac{T(\gamma \preceq \psi, [x, y], \tilde{\mathbb{D}})}{F(\beta \preceq \psi, [x, y], \tilde{\mathbb{D}})}}{\mathbf{X}}$$

where $\beta \preceq \gamma$

$$(X6) \frac{Q(\cdot, \cdot, \tilde{\mathbb{D}})}{\mathbf{X}}$$

where $\tilde{\mathbb{D}}$ is inconsistent

Figure 12: Branch closing rules.

Example: Tableau Construction

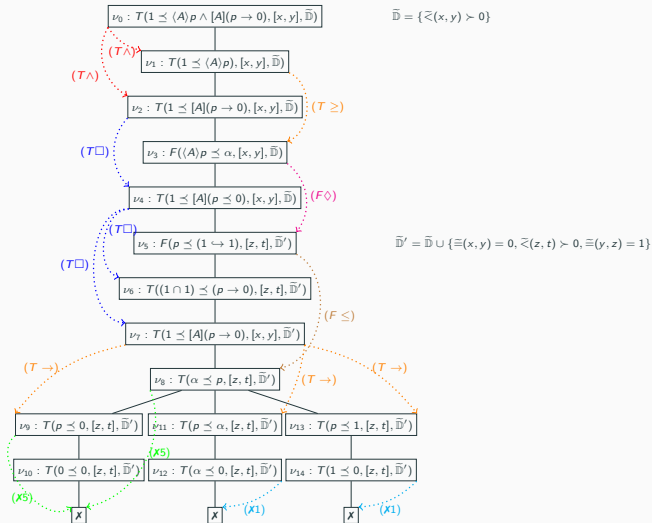


Figure 13: Some closed branches of the tableau for $\langle A \rangle p \wedge [A](p \rightarrow 0)$ and $1 \in G3$.

Soundness and Completeness over Finite FL_{ew} -Algebras

- **Soundness**

- If a formula φ is α -satisfiable, then there exists an opened tableau for φ and α
- The rules preserve logical consequence

- **Completeness**

- If a tableau is opened for φ and α , then φ is α -satisfiable.
- The method explores all necessary valuations

- **Implications**

- The tableau system is a reliable decision procedure for MVHS over **finite** FL_{ew} -algebras
- Provides a foundation for automated reasoning in MVHS

Implementation Overview

- **Programming Language:**

- **Julia**, chosen for its performance in numerical computations
- Benefits of Julia:
 - High-level syntax with efficient execution
 - Strong support for mathematical operations

- **Open-source Advocacy:**

- **Sole.jl** (SymbOlic LEarning)¹, a framework for representing, reasoning, and learning from structured and unstructured data
- **SoleReasoners.jl**, analytic tableau solvers for α -sat and α -val.

- **Representation of Algebras:**

- Finite FL_{ew} -algebras defined by specifying:
 - **Domain** (set of truth values A)
 - **Truth tables** for \cap , \cup , \cdot , $+$ (\hookrightarrow is derived internally)
- **Validation:**
 - A one-time check ensures the algebra satisfies the FL_{ew} -axioms.
- Wrapped in the **ManyValuedLogics** submodule of **SoleLogics.jl**

¹<https://github.com/aclai-lab/Sole.jl>

Code Examples: Gödel Algebra (G3)

```
1  using SoleLogics
2  using SoleLogics.ManyValuedLogics
3  using SoleReasoners
4
5  α = FiniteTruth("α")
6  d3 = Vector{FiniteTruth}([⊥, α, ⊤])
7  # a ∧ b = min{a, b}
8  n = BinaryOperation(d3, Dict{Tuple{FiniteTruth, FiniteTruth}, FiniteTruth}({
9      (⊥, ⊥) => ⊥, (⊥, α) => ⊥, (⊥, ⊤) => ⊥,
10     (α, ⊥) => ⊥, (α, α) => α, (α, ⊤) => α,
11     (⊤, ⊥) => ⊥, (⊤, α) => α, (⊤, ⊤) => ⊤
12 }
13 ))
14 # a ∨ b = max{a, b}
15 u = BinaryOperation(d3, Dict{Tuple{FiniteTruth, FiniteTruth}, FiniteTruth}({
16     (⊥, ⊥) => ⊥, (⊥, α) => α, (⊥, ⊤) => ⊤,
17     (α, ⊥) => α, (α, α) => α, (α, ⊤) => ⊤,
18     (⊤, ⊥) => ⊤, (⊤, α) => ⊤, (⊤, ⊤) => ⊤
19 }
20 ))
21 · = n # In Gödel algebras, n and · are the same operator
22 + = u # In Gödel algebras, u and + are the same operator
23 G3 = FiniteFlewAlgebra(d3, n, u, ·, +, ⊥, ⊤)
24
25 diamondA = diamond(IA_A) # {A}
26 boxA = box(IA_A) # [A]
27 p = Atom("p") # propositional letter "p"
28 φ = ∧(diamondA(p), boxA(¬(p, ⊥))) # φ := {A}p ∧ [A](p → ⊥)
29 mvhsalphasat(⊤, φ, G3) # false
```

Figure 14: Evaluation code example for $T(\top \preceq \varphi)$ where $\varphi = \langle A \rangle p \wedge [A](p \rightarrow 0)$ and $\top \in G3$.

Code Examples: Heyting Algebra (H4)

```
1 using SoleLogics
2 using SoleLogics.ManyValuedLogics
3 using SoleReasoners
4
5  $\alpha$  = FiniteTruth("α")
6  $\beta$  = FiniteTruth("β")
7 d4 = Vector{FiniteTruth}([ $\perp$ ,  $\alpha$ ,  $\beta$ ,  $\top$ ])
8 n = BinaryOperation(d4, Dict{Tuple{FiniteTruth, FiniteTruth}, FiniteTruth}(
9     ( $\perp$ ,  $\perp$ ) =>  $\perp$ , ( $\perp$ ,  $\alpha$ ) =>  $\perp$ , ( $\perp$ ,  $\beta$ ) =>  $\perp$ , ( $\perp$ ,  $\top$ ) =>  $\perp$ ,
10    ( $\alpha$ ,  $\perp$ ) =>  $\perp$ , ( $\alpha$ ,  $\alpha$ ) =>  $\alpha$ , ( $\alpha$ ,  $\beta$ ) =>  $\perp$ , ( $\alpha$ ,  $\top$ ) =>  $\alpha$ ,
11    ( $\beta$ ,  $\perp$ ) =>  $\perp$ , ( $\beta$ ,  $\alpha$ ) =>  $\perp$ , ( $\beta$ ,  $\beta$ ) =>  $\beta$ , ( $\beta$ ,  $\top$ ) =>  $\beta$ ,
12    ( $\top$ ,  $\perp$ ) =>  $\perp$ , ( $\top$ ,  $\alpha$ ) =>  $\alpha$ , ( $\top$ ,  $\beta$ ) =>  $\beta$ , ( $\top$ ,  $\top$ ) =>  $\top$ 
13 ))
14 u = BinaryOperation(d4, Dict{Tuple{FiniteTruth, FiniteTruth}, FiniteTruth}(
15     ( $\perp$ ,  $\perp$ ) =>  $\perp$ , ( $\perp$ ,  $\alpha$ ) =>  $\alpha$ , ( $\perp$ ,  $\beta$ ) =>  $\beta$ , ( $\perp$ ,  $\top$ ) =>  $\top$ ,
16     ( $\alpha$ ,  $\perp$ ) =>  $\alpha$ , ( $\alpha$ ,  $\alpha$ ) =>  $\alpha$ , ( $\alpha$ ,  $\beta$ ) =>  $\top$ , ( $\alpha$ ,  $\top$ ) =>  $\top$ ,
17     ( $\beta$ ,  $\perp$ ) =>  $\beta$ , ( $\beta$ ,  $\alpha$ ) =>  $\top$ , ( $\beta$ ,  $\beta$ ) =>  $\beta$ , ( $\beta$ ,  $\top$ ) =>  $\top$ ,
18     ( $\top$ ,  $\perp$ ) =>  $\top$ , ( $\top$ ,  $\alpha$ ) =>  $\top$ , ( $\top$ ,  $\beta$ ) =>  $\top$ , ( $\top$ ,  $\top$ ) =>  $\top$ 
19 ))
20  $\cdot$  = n # In Heyting algebras, n and  $\cdot$  are the same operator
21 + = u # In Heyting algebras, u and + are the same operator
22 H4 = FiniteFLewAlgebra(d4, n, u,  $\cdot$ , +,  $\perp$ ,  $\top$ )
23
24 diamondA = diamond(IA_A) # {A}
25 boxA = box(IA_A) # [A]
26 p = Atom("p") # propositional letter "p"
27  $\varphi$  =  $\wedge$ (diamondA(p), boxA( $\neg$ (p,  $\perp$ ))) #  $\varphi := \{A\}p \wedge [A](p \rightarrow \perp)$ 
28 mvhsalphasat( $\top$ ,  $\varphi$ , H4) # false
```

Figure 15: Evaluation code example for $T(\top \preceq \varphi)$ where

$\varphi = \langle A \rangle p \wedge [A](p \rightarrow 0)$ and $\top \in H4$.

Key Challenges and Solutions

- Computational complexity increases with:
 - **Size of the algebra:** More truth values to consider
 - **Complexity of the formula:** More nodes and branches
- **Optimization techniques:**
 - Implemented **priority queues** to manage node expansion efficiently
 - **Parallelization:** Expanded independent branches using multi-core processors
- **Pruning strategies:** Periodically clean priority queues to remove expanded or closed nodes
- **Efficient data structures:**
 - Designed compact representations for nodes and branches
 - Minimized memory usage to handle large tableaux

Experiments and Results

Experiments and Results

- Six representative finite FL_{ew} -algebras:



- G3 and MV3 (resp. G4 and MV4) differ because of the t-norm but share the same lattice structure
- Each algebra tested on the same 500 random formulas with heights up to 5 (i.e., 32 symbols)
- $\alpha \succ 0$ chosen randomly
- Branch priority policy kept random

Experiments and Results

- Impact of using different FL_{ew} -Algebras
- All tests were conducted on a machine equipped with 2 Intel Xeon Gold 28-Core CPUs and 224GB of RAM
- Timeout of 30 seconds

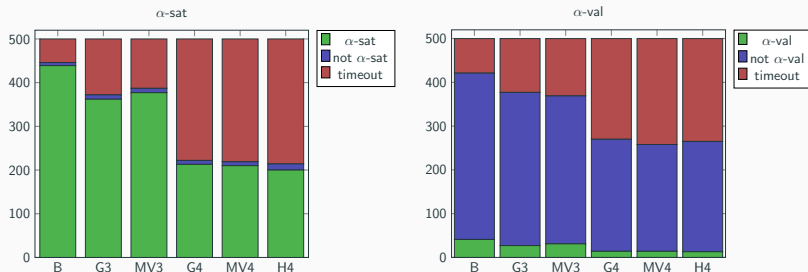


Figure 16: Results on common many-valued algebras for formulas of height up to 5 with a timeout of 30 seconds.

Conclusions and Future Work

Conclusions and Future Work

- Presented a **customizable and flexible framework** for many-valued interval temporal logic
- FL_{ew} -algebras facilitate **reasoning** in systems with **uncertainty** and **graded truths**
- Developed a **sound and complete tableau system** for MVHS
- Ready-to-use **open-source implementation** with user-definable finite (FL_{ew} -)algebras²
- **Tested** the tableau system over different **finite algebras**
- Future work:
 - Support for **Interval Spatial Logic**
 - Support for **non-finite algebras**
 - Real-world applications (**Many-Expert Decision Tree Learning**)

²<https://github.com/aclai-lab/Sole.jl>

Questions?



M. Baaz, N. Preining, and R. Zach.

First-order Gödel logics.

Annals of Pure and Applied Logic, 147:23–47, 2007.



C. C. Chang.

Algebraic analysis of many valued logics.

Transactions of the American Mathematical society, 88(2):467–490, 1958.



L. Esakia, G. Bezhanishvili, W. Holliday, and A. Evseev.

Heyting Algebras: Duality Theory.

Springer, 2019.



P. Hájek.

Basic fuzzy logic and bl-algebras.

Soft computing, 2:124–128, 1998.



P. Hájek.

The Metamathematics of Fuzzy Logic.

Kluwer, 1998.



H. Ono and Y. Komori.

Logics without the contraction rule.

The Journal of Symbolic Logic, 50(1):169–201, 1985.



A. Rose.

Formalisations of further \aleph_0 -valued Łukasiewicz propositional calculi.

Journal of Symbolic Logic, 43(2):207–210, 1978.