Tractable Reasoning with Limited Belief

Christoph Schwering

UNSW Sydney

COMP4418, Week 8

Computational Aspects of Reasoning

- Good news:
  $\mathbf{O}\mathrm{KB} \models \alpha$ reduces to $\mathrm{KB} \models \phi_1, \ldots, \mathrm{KB} \models \phi_k$ (Representation Theorem)
  - No modal reasoning necessary (no $\mathbf{O}$, no $\mathbf{K}$)

Computational Aspects of Reasoning

- <u>Good news</u>:
  $\mathbf{O}\mathrm{KB} \models \alpha$ reduces to $\mathrm{KB} \models \phi_1, \ldots, \mathrm{KB} \models \phi_k$ (Representation Theorem)
  - No modal reasoning necessary (no $\mathbf{O}$, no $\mathbf{K}$)

- <u>Bad (old) news</u>:
  First-order case: $\mathrm{KB} \models \phi$ is only semidecidable
  - There is a procedure that always halts if $\mathrm{KB} \models \phi$
  - But it might not terminate if $\mathrm{KB} \not\models \phi$

Computational Aspects of Reasoning

- Good news:
  $\mathbf{O}\mathrm{KB} \models \alpha$ reduces to $\mathrm{KB} \models \phi_1, \ldots, \mathrm{KB} \models \phi_k$ (Representation Theorem)
  - No modal reasoning necessary (no $\mathbf{O}$, no $\mathbf{K}$)

- Bad (old) news:
  First-order case: $\mathrm{KB} \models \phi$ is only semidecidable
  - There is a procedure that always halts if $\mathrm{KB} \models \phi$
  - But it might not terminate if $\mathrm{KB} \not\models \phi$

  Propositional case: $\mathrm{KB} \models \phi$ is intractable (or P = NP)
  - $\mathrm{KB} \models \phi$ is co-NP-complete
  - co-NP contains all problems whose complement is in NP

Some Options We Have

1. We could use a standard theorem prover and set a timeout.
   But: The underlying logic would be hopelessly complex.

Some Options We Have

1. We could use a standard theorem prover and set a timeout.

   But: The underlying logic would be hopelessly complex.

2. We could restrict the expressivity of our representation language.
   - Horn logic
   - Description logics

   But: Humans can deal with very complex representations.

Logical Omniscience

Some sources of complexity:

- All tautologies are known
  E.g., $\models \mathbf{K}(p \vee \neg p)$

Logical Omniscience

Some sources of complexity:

- All tautologies are known
  E.g., $\models \mathbf{K}(p \vee \neg p)$

- Knowledge is closed under logical consequence
  E.g., $\models \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$

Logical Omniscience

Some sources of complexity:

- All tautologies are known
  E.g., $\models \mathbf{K}(p \vee \neg p)$
- Knowledge is closed under logical consequence
  E.g., $\models \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$
- Knowledge is closed under equivalence
  E.g., $\models \mathbf{K}p \leftrightarrow \mathbf{K}(p \wedge (q \vee \neg q))$

Logical Omniscience

Some sources of complexity:

- All tautologies are known
  E.g., $\models \mathbf{K}(p \vee \neg p)$
- Knowledge is closed under logical consequence
  E.g., $\models \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$
- Knowledge is closed under equivalence
  E.g., $\models \mathbf{K}p \leftrightarrow \mathbf{K}(p \wedge (q \vee \neg q))$
- Inconsistent knowledge implies knowing everything (incl. nonsense)
  E.g., $\models \mathbf{K}(p \wedge \neg p) \rightarrow \mathbf{K}q$

Limited Belief

- Want to weaken the entailment relation $\models$, call it $\approx$.

Limited Belief

- Want to weaken the entailment relation $\models$, call it $\approx\!\!\!\!|$.
- Should be sound: $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha \implies \mathbf{O}\mathrm{KB} \models \mathbf{K}\alpha$

Limited Belief

- Want to weaken the entailment relation $\models$, call it $\approx\!\!\!\!|$.
- Should be sound: $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha \implies \mathbf{O}\mathrm{KB} \models \mathbf{K}\alpha$
- Should be incomplete: $\mathbf{O}\mathrm{KB} \models \mathbf{K}\alpha \;\not\Longrightarrow\; \mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$

Limited Belief

- Want to weaken the entailment relation $\models$, call it $\approx\!\!\!\!\mid$.
- Should be sound: $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha \implies \mathbf{O}\mathrm{KB} \models \mathbf{K}\alpha$
- Should be incomplete: $\mathbf{O}\mathrm{KB} \models \mathbf{K}\alpha \not\Longrightarrow \mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$
- Should be easy to compute $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$

Limited Belief

- Want to weaken the entailment relation $\models$, call it $\approx\!\!\!\!|$.
- Should be sound: $\mathbf{O}\text{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha \implies \mathbf{O}\text{KB} \models \mathbf{K}\alpha$
- Should be incomplete: $\mathbf{O}\text{KB} \models \mathbf{K}\alpha \not\Longrightarrow \mathbf{O}\text{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$
- Should be easy to compute $\mathbf{O}\text{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$

This is different from restricting expressiveness:

- Horn logic, description logics restrict the *language*
- Limited belief restricts the *semantics* (mainly)

Overview of the Lecture

- **Limited Belief – First Attempt**
- Limited Belief – Second Attempt
- Data structures and algorithms for ASP solvers

Limited Belief — First Attempt

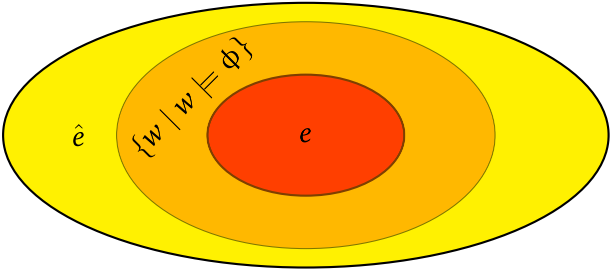
Idea: Allow more models as part of epistemic state e

Limited Belief — First Attempt

Idea: Allow more models as part of epistemic state e

Why?

- Larger e corresponds to fewer beliefs
- $e \models \mathbf{K}\phi \iff$ for all $w \in e$, $w \models \phi$
- For $\hat{e} \supseteq e$: $\hat{e} \models \mathbf{K}\phi \implies e \models \mathbf{K}\phi$

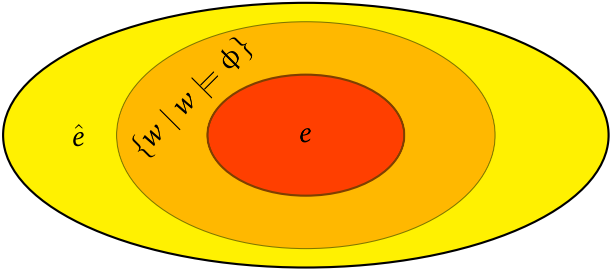


Limited Belief — First Attempt

Idea: Allow more models as part of epistemic state e

Why?

- Larger e corresponds to fewer beliefs
- $e \models \mathbf{K}\phi \iff$ for all $w \in e,\ w \models \phi$
- For $\hat{e} \supseteq e$: $\hat{e} \models \mathbf{K}\phi \implies e \models \mathbf{K}\phi$



<u>Simplification</u>: propositional logic for now, no nested $\mathbf{O}$, $\mathbf{K}$.

Multi-Valued Worlds

Definition: multi-valued world

A **multi-valued world** v is a function from the atomic propositions to $\{\{\}, \{0\}, \{1\}, \{0, 1\}\}$.

Multi-Valued Worlds

Definition: multi-valued world

A **multi-valued world** v is a function from the atomic propositions to $\{\{\}, \{0\}, \{1\}, \{0, 1\}\}$.

- $v[p] = \{\}$ means "never heard of it"
- $v[p] = \{0\}$ means "false"
- $v[p] = \{1\}$ means "true"
- $v[p] = \{0, 1\}$ means "conflicting information"

We will define *true support* and *false support*.

Multi-Valued Worlds

Definition: multi-valued world

A **multi-valued world** v is a function from the atomic propositions to $\{\{\}, \{0\}, \{1\}, \{0, 1\}\}$.

An **epistemic state** e is a set of multi-valued worlds.

- $v[p] = \{\}$ means "never heard of it"
- $v[p] = \{0\}$ means "false"
- $v[p] = \{1\}$ means "true"
- $v[p] = \{0, 1\}$ means "conflicting information"

We will define *true support* and *false support*.

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_{\mathrm{T}} P \iff v[P] \ni 1$
  $e, v \models_{\mathrm{F}} P \iff v[P] \ni 0$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_{\mathrm{T}} P \iff v[P] \ni 1$
  $e, v \models_{\mathrm{F}} P \iff v[P] \ni 0$
- $e, v \models_{\mathrm{T}} \neg\alpha \iff e, v \models_{\mathrm{F}} \alpha$
  $e, v \models_{\mathrm{F}} \neg\alpha \iff e, v \models_{\mathrm{T}} \alpha$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_{\mathrm{T}} P \iff v[P] \ni 1$
  $e, v \models_{\mathrm{F}} P \iff v[P] \ni 0$
- $e, v \models_{\mathrm{T}} \neg\alpha \iff e, v \models_{\mathrm{F}} \alpha$
  $e, v \models_{\mathrm{F}} \neg\alpha \iff e, v \models_{\mathrm{T}} \alpha$
- $e, v \models_{\mathrm{T}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{T}} \alpha$ or $e, v \models_{\mathrm{T}} \beta$
  $e, v \models_{\mathrm{F}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{F}} \alpha$ and $e, v \models_{\mathrm{F}} \beta$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_{\mathrm{T}} P \iff v[P] \ni 1$
  $e, v \models_{\mathrm{F}} P \iff v[P] \ni 0$
- $e, v \models_{\mathrm{T}} \neg\alpha \iff e, v \models_{\mathrm{F}} \alpha$
  $e, v \models_{\mathrm{F}} \neg\alpha \iff e, v \models_{\mathrm{T}} \alpha$
- $e, v \models_{\mathrm{T}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{T}} \alpha$ or $e, v \models_{\mathrm{T}} \beta$
  $e, v \models_{\mathrm{F}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{F}} \alpha$ and $e, v \models_{\mathrm{F}} \beta$
- $e, v \models_{\mathrm{T}} (\alpha \wedge \beta) \iff e, v \models_{\mathrm{T}} \alpha$ and $e, v \models_{\mathrm{T}} \beta$
  $e, v \models_{\mathrm{F}} (\alpha \wedge \beta) \iff e, v \models_{\mathrm{F}} \alpha$ or $e, v \models_{\mathrm{F}} \beta$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_T P \iff v[P] \ni 1$

  $e, v \models_F P \iff v[P] \ni 0$
- $e, v \models_T \neg\alpha \iff e, v \models_F \alpha$

  $e, v \models_F \neg\alpha \iff e, v \models_T \alpha$
- $e, v \models_T (\alpha \vee \beta) \iff e, v \models_T \alpha$ or $e, v \models_T \beta$

  $e, v \models_F (\alpha \vee \beta) \iff e, v \models_F \alpha$ and $e, v \models_F \beta$
- $e, v \models_T (\alpha \wedge \beta) \iff e, v \models_T \alpha$ and $e, v \models_T \beta$

  $e, v \models_F (\alpha \wedge \beta) \iff e, v \models_F \alpha$ or $e, v \models_F \beta$
- $e, v \models_T \mathbf{K}\alpha \iff$ for all v', $v' \in e \Rightarrow e, v' \models_T \alpha$

  $e, v \models_F \mathbf{K}\alpha \iff e, v \not\models_T \mathbf{K}\alpha$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_{\mathrm{T}} P \iff v[P] \ni 1$
  $e, v \models_{\mathrm{F}} P \iff v[P] \ni 0$
- $e, v \models_{\mathrm{T}} \neg\alpha \iff e, v \models_{\mathrm{F}} \alpha$
  $e, v \models_{\mathrm{F}} \neg\alpha \iff e, v \models_{\mathrm{T}} \alpha$
- $e, v \models_{\mathrm{T}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{T}} \alpha$ or $e, v \models_{\mathrm{T}} \beta$
  $e, v \models_{\mathrm{F}} (\alpha \vee \beta) \iff e, v \models_{\mathrm{F}} \alpha$ and $e, v \models_{\mathrm{F}} \beta$
- $e, v \models_{\mathrm{T}} (\alpha \wedge \beta) \iff e, v \models_{\mathrm{T}} \alpha$ and $e, v \models_{\mathrm{T}} \beta$
  $e, v \models_{\mathrm{F}} (\alpha \wedge \beta) \iff e, v \models_{\mathrm{F}} \alpha$ or $e, v \models_{\mathrm{F}} \beta$
- $e, v \models_{\mathrm{T}} \mathbf{K}\alpha \iff$ for all v', $v' \in e \Rightarrow e, v' \models_{\mathrm{T}} \alpha$
  $e, v \models_{\mathrm{F}} \mathbf{K}\alpha \iff e, v \not\models_{\mathrm{T}} \mathbf{K}\alpha$
- $e, v \models_{\mathrm{T}} \mathbf{O}\alpha \iff$ for all v', $v' \in e \Leftrightarrow e, v' \models_{\mathrm{T}} \alpha$
  $e, v \models_{\mathrm{F}} \mathbf{O}\alpha \iff e, v \not\models_{\mathrm{T}} \mathbf{O}\alpha$

The Semantics of True and False Support

Definition: semantics of true and false support

- $e, v \models_T P \iff v[P] \ni 1$

  $e, v \models_F P \iff v[P] \ni 0$
- $e, v \models_T \neg\alpha \iff e, v \models_F \alpha$

  $e, v \models_F \neg\alpha \iff e, v \models_T \alpha$
- $e, v \models_T (\alpha \vee \beta) \iff e, v \models_T \alpha$ or $e, v \models_T \beta$

  $e, v \models_F (\alpha \vee \beta) \iff e, v \models_F \alpha$ and $e, v \models_F \beta$
- $e, v \models_T (\alpha \wedge \beta) \iff e, v \models_T \alpha$ and $e, v \models_T \beta$

  $e, v \models_F (\alpha \wedge \beta) \iff e, v \models_F \alpha$ or $e, v \models_F \beta$
- $e, v \models_T \mathbf{K}\alpha \iff$ for all v', $v' \in e \Rightarrow e, v' \models_T \alpha$

  $e, v \models_F \mathbf{K}\alpha \iff e, v \not\models_T \mathbf{K}\alpha$
- $e, v \models_T \mathbf{O}\alpha \iff$ for all v', $v' \in e \Leftrightarrow e, v' \models_T \alpha$

  $e, v \models_F \mathbf{O}\alpha \iff e, v \not\models_T \mathbf{O}\alpha$

$\mathbf{O}\mathrm{KB} \approx \mathbf{K}\alpha \iff$ for all e, v, $e, v \models_T \mathbf{O}\mathrm{KB} \Rightarrow e, v \models_T \mathbf{K}\alpha$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v \models_{\mathrm{T}} (p \vee q \vee r) \wedge (p \vee q \vee \neg r)\}$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v \models_{\mathrm{T}} (p \vee q \vee r) \wedge (p \vee q \vee \neg r)\}$
  $= \{v \mid v \models_{\mathrm{T}} p$ or $v \models_{\mathrm{T}} q$ or $v \models_{\mathrm{T}} r$, and
  $v \models_{\mathrm{T}} p$ or $v \models_{\mathrm{T}} q$ or $v \models_{\mathrm{F}} r\}$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v \models_{\mathrm{T}} (p \vee q \vee r) \wedge (p \vee q \vee \neg r)\}$
  $= \{v \mid v \models_{\mathrm{T}} p$ or $v \models_{\mathrm{T}} q$ or $v \models_{\mathrm{T}} r$, and
  $v \models_{\mathrm{T}} p$ or $v \models_{\mathrm{T}} q$ or $v \models_{\mathrm{F}} r\}$
  $= \{v \mid v[p] \ni 1$ or $v[q] \ni 1$ or $v[r] = \{0, 1\}\}$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1$ or $v[q] \ni 1$ or $v[r] = \{0, 1\}\}$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \lor q)$
  $\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} (p \lor q)$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$
  $\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} (p \vee q)$
  $\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} p$ or $v \models_{\mathrm{T}} q$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$

$\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} (p \vee q)$
$\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} p \text{ or } v \models_{\mathrm{T}} q$
$\iff$ for all v, $v \in e \Rightarrow v[p] \ni 1 \text{ or } v[q] \ni 1$ ✗

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0,1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1$ or $v[q] \ni 1$ or $v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$
  $\iff e \models_{\mathrm{F}} \mathbf{K}p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$
  $\iff e \models_{\mathrm{F}} \mathbf{K}p$
  $\iff e \not\models_{\mathrm{T}} \mathbf{K}p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$

$\iff$ $e \models_{\mathrm{F}} \mathbf{K}p$

$\iff$ $e \not\models_{\mathrm{T}} \mathbf{K}p$

$\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{T}} p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$

$\iff$ $e \models_{\mathrm{F}} \mathbf{K}p$
$\iff$ $e \not\models_{\mathrm{T}} \mathbf{K}p$
$\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{T}} p$
$\iff$ for some v, $v \in e$ and $v[p] \not\ni 1$ ✓

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1$ or $v[q] \ni 1$ or $v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$
  $\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{T}} \neg p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$

  $\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{T}} \neg p$

  $\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{F}} p$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$

$\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{T}} \neg p$
$\iff$ for some v, $v \in e$ and $v \not\models_{\mathrm{F}} p$
$\iff$ for some v, $v \in e$ and $v[p] \not\ni 0$ ✓

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$ ✓

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$ ✓
- $e \models_{\mathrm{T}} \mathbf{K}(r \vee \neg r)$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$ ✓
- $e \models_{\mathrm{T}} \mathbf{K}(r \vee \neg r)$
  $\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} (r \vee \neg r)$

Examples

Let $e \models_T \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0,1\}\}$
- $e \models_T \mathbf{K}(p \vee q)$ ✗
- $e \models_T \neg\mathbf{K}p$ ✓
- $e \models_T \neg\mathbf{K}\neg p$ ✓
- $e \models_T \mathbf{K}(r \vee \neg r)$

  $\iff$ for all v, $v \in e \Rightarrow v \models_T (r \vee \neg r)$

  $\iff$ for all v, $v \in e \Rightarrow v \models_T r$ or $v \models_F r$

Examples

Let $e \models_T \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_T \mathbf{K}(p \vee q)$ ✗
- $e \models_T \neg\mathbf{K}p$ ✓
- $e \models_T \neg\mathbf{K}\neg p$ ✓
- $e \models_T \mathbf{K}(r \vee \neg r)$

$$\begin{aligned}
&\iff \text{for all } v,\ \ v \in e \Rightarrow v \models_T (r \vee \neg r) \\
&\iff \text{for all } v,\ \ v \in e \Rightarrow v \models_T r \text{ or } v \models_F r \\
&\iff \text{for all } v,\ \ v \in e \Rightarrow v[r] \ni 0 \text{ or } v[r] \ni 1
\end{aligned}$$

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \vee q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$ ✓
- $e \models_{\mathrm{T}} \mathbf{K}(r \vee \neg r)$

$\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} (r \vee \neg r)$
$\iff$ for all v, $v \in e \Rightarrow v \models_{\mathrm{T}} r$ or $v \models_{\mathrm{F}} r$
$\iff$ for all v, $v \in e \Rightarrow v[r] \ni 0$ or $v[r] \ni 1$
$\iff$ for all v, $v \in e \Rightarrow v[r] \neq \{\}$ ✗

Examples

Let $e \models_{\mathrm{T}} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0, 1\}\}$
- $e \models_{\mathrm{T}} \mathbf{K}(p \lor q)$ ✗
- $e \models_{\mathrm{T}} \neg\mathbf{K}p$ ✓
- $e \models_{\mathrm{T}} \neg\mathbf{K}\neg p$ ✓
- $e \models_{\mathrm{T}} \mathbf{K}(r \lor \neg r)$ ✗

Examples

Let $e \models_T \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $e = \{v \mid v[p] \ni 1 \text{ or } v[q] \ni 1 \text{ or } v[r] = \{0,1\}\}$
- $e \models_T \mathbf{K}(p \lor q)$ ✗
- $e \models_T \neg\mathbf{K}p$ ✓
- $e \models_T \neg\mathbf{K}\neg p$ ✓
- $e \models_T \mathbf{K}(r \lor \neg r)$ ✗

So $\mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$ really doesn't entail much...

Some Properties

- Not all tautologies are known
  $\not\approx \mathbf{K}(p \vee \neg p)$
  falsified by $e = \{v\}$ for $v[p] = \{\}$

Some Properties

- Not all tautologies are known
  $\not\approx \mathbf{K}(p \vee \neg p)$
  falsified by $e = \{v\}$ for $v[p] = \{\}$

- Knowledge is not closed under logical consequence
  $\not\approx \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$
  falsified by $e = \{v\}$ for $v[p] = \{0, 1\}$ and $v[q] \not\ni 1$

Some Properties

- Not all tautologies are known
  $\not\approx \mathbf{K}(p \vee \neg p)$
  falsified by $e = \{v\}$ for $v[p] = \{\}$

- Knowledge is not closed under logical consequence
  $\not\approx \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$
  falsified by $e = \{v\}$ for $v[p] = \{0, 1\}$ and $v[q] \not\ni 1$

- Knowledge is not closed under equivalence
  $\not\approx \mathbf{K}p \leftrightarrow \mathbf{K}(p \wedge (q \vee \neg q))$
  falsified by $e = \{v\}$ for $v[p] = 1$ and $v[q] = \{\}$

Some Properties

- Not all tautologies are known
  $\not\approx \mathbf{K}(p \vee \neg p)$
  falsified by $e = \{v\}$ for $v[p] = \{\}$

- Knowledge is not closed under logical consequence
  $\not\approx \mathbf{K}p \wedge \mathbf{K}(p \rightarrow q) \rightarrow \mathbf{K}q$
  falsified by $e = \{v\}$ for $v[p] = \{0, 1\}$ and $v[q] \not\ni 1$

- Knowledge is not closed under equivalence
  $\not\approx \mathbf{K}p \leftrightarrow \mathbf{K}(p \wedge (q \vee \neg q))$
  falsified by $e = \{v\}$ for $v[p] = 1$ and $v[q] = \{\}$

- Inconsistent knowledge does not imply knowing everything
  $\not\approx \mathbf{K}(p \wedge \neg p) \rightarrow \mathbf{K}q$
  falsified by $e = \{v\}$ for $v[p] = \{0, 1\}$ and $v[q] \not\ni 1$

Complexity

None of the sources of complexity we identified on slide 7 remains.

Is reasoning easier now?

Complexity

None of the sources of complexity we identified on slide 7 remains.

Is reasoning easier now?

Bad news: Not in general.

Theorem: complexity

$\mathbf{O}\mathrm{KB} \approx\!\!\!\!| \; \mathbf{K}\phi$ is co-NP-complete.

Complexity

None of the sources of complexity we identified on slide 7 remains.

Is reasoning easier now?

Bad news: Not in general.

Theorem: complexity

$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\phi$ is co-NP-complete.

Let $p_1, \ldots, p_n$ be the propositions in KB and ϕ.

$$\mathrm{KB} \models \phi \iff \mathbf{O}(\mathrm{KB} \wedge \underbrace{\textstyle\bigwedge_i (p_i \vee \neg p_i)}_{\text{prevent ``conflicting information''}}) \mathrel{|\!\!\approx} \mathbf{K}(\phi \vee \underbrace{\textstyle\bigvee_i (p_i \wedge \neg p_i)}_{\text{ignore ``never heard of'' worlds}})$$

Complexity (2)

Good news: Reasoning gets very easy when KB and ϕ are in CNF.

Theorem: decision procedure for CNF KB, ϕ

Let $\mathrm{KB} \stackrel{\text{def}}{=} c_1 \land \ldots \land c_m$ and $\phi \stackrel{\text{def}}{=} d_1 \land \ldots \land d_n$ for clauses c_i, d_j.

$\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}\phi$ is decidable in $\mathcal{O}(m \cdot n)$.

$\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}\phi \iff$ for every d_j, there is a c_i with $c_i \subseteq d_j$.

Ex.: $\mathbf{O}((p \lor \neg q) \land q) \mathrel{|\!\approx} \mathbf{K}(p \lor \neg q \lor r)$ since $\{p, \neg q\} \subseteq \{p, \neg q, r\}$.

$\mathbf{O}((p \lor \neg q) \land q) \mathrel{|\!\not\approx} \mathbf{K}p$ since $\{p, \neg q\} \not\subseteq \{p\}$, $\{q\} \not\subseteq \{p\}$.

Proof on paper.

The First-Order Case

Generalise to first-order $\mathcal{OL}$ (function symbols aside):

- Predicates: $P(t_1, \ldots, t_j)$ where t_i is variable or standard name
- Quantification: $\exists x\, \alpha$

The First-Order Case

Generalise to first-order $\mathcal{OL}$ (function symbols aside):

- Predicates: $P(t_1, \ldots, t_j)$ where t_i is variable or standard name
- Quantification: $\exists x\, \alpha$

Generalise the true and false support semantics to this language:

Definition: multi-valued world, first-order case

$P(\vec{n})$ is **primitive** iff all n_i are standard names.
A **multi-valued world** v is a function from the primitive atomic formulas to $\{\{\}, \{0\}, \{1\}, \{0, 1\}\}$.

- $e, v \models_{\mathrm{T}} \exists x\, \alpha \iff e, v \models_{\mathrm{T}} \alpha^x_n$ for some standard name n
  $e, v \models_{\mathrm{F}} \exists x\, \alpha \iff e, v \models_{\mathrm{F}} \alpha^x_n$ for every standard name n

Complexity in the First-Order Case

Bad news: Too complex.

Theorem: complexity, first-order case

$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}\alpha$ is undecidable.

Complexity in the First-Order Case

Bad news: Too complex.

Theorem: complexity, first-order case

$\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}\alpha$ is undecidable.

Let $P_1, \ldots, P_n$ be the predicate symbols in KB and ϕ.

$$\mathrm{KB} \models \phi \iff$$
$$\mathbf{O}(\mathrm{KB} \wedge \underbrace{\textstyle\bigwedge_i \forall \vec{x}\,(P_i(\vec{x}) \vee \neg P_i(\vec{x}))}_{\text{prevent "conflicting information"}}) \mathrel{|\!\approx} \mathbf{K}(\phi \vee \underbrace{\textstyle\bigvee_i \forall \vec{x}\,(P_i(\vec{x}) \wedge \neg P_i(\vec{x}))}_{\text{ignore "never heard of" worlds}})$$

Overview of the Lecture

- Limited Belief – First Attempt
- **Limited Belief – Second Attempt**
- Data structures and algorithms for ASP solvers

Limited Belief — Second Attempt

What went wrong in the First Attempt?

- “Believe or not”, no way of controlling how much to “think”
- Knowledge is very weak, not closed under forward chaining
- Still very complex in the worst case

Limited Belief — Second Attempt

What went wrong in the First Attempt?

- “Believe or not”, no way of controlling how much to “think”
- Knowledge is very weak, not closed under forward chaining
- Still very complex in the worst case

Idea: stratify beliefs into levels

- Use $\mathbf{K}_k\,\alpha$ to say “α is known at level k”
- Level 0: explicit beliefs
- Level $k + 1$: level k plus some inferences

Limited Belief — Second Attempt

What went wrong in the First Attempt?

- "Believe or not", no way of controlling how much to "think"
- Knowledge is very weak, not closed under forward chaining
- Still very complex in the worst case

Idea: stratify beliefs into levels

- Use $\mathbf{K}_k\,\alpha$ to say "α is known at level k"
- Level 0: explicit beliefs
- Level $k+1$: level k plus some inferences

Simplification: propositional logic for now, no nested $\mathbf{O}$, $\mathbf{K}$.

Easy Inferences

What should count as explicit belief?

- If $\alpha \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,\alpha$

Easy Inferences

What should count as explicit belief?

- If $\alpha \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}_0\,\alpha$
- If $p \in \mathrm{KB}$ and $p \rightarrow q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}_0\,q$

Easy Inferences

What should count as explicit belief?

- If $\alpha \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}_0\,\alpha$
- If $p \in \mathrm{KB}$ and $p \to q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}_0\,q$
- If $p \in \mathrm{KB}$ and $p \to q \in \mathrm{KB}$ and $q \to r \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\approx} \mathbf{K}_0\,r$

Easy Inferences

What should count as explicit belief?

- If $\alpha \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|}\!\approx \mathbf{K}_0\,\alpha$
- If $p \in \mathrm{KB}$ and $p \rightarrow q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|}\!\approx \mathbf{K}_0\,q$
- If $p \in \mathrm{KB}$ and $p \rightarrow q \in \mathrm{KB}$ and $q \rightarrow r \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|}\!\approx \mathbf{K}_0\,r$
- If $p_1, \ldots, p_j \in \mathrm{KB}$ and $p_1 \wedge \ldots \wedge p_j \rightarrow q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|}\!\approx \mathbf{K}_0\,q$

Easy Inferences

What should count as explicit belief?

- If $\alpha \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,\alpha$
- If $p \in \mathrm{KB}$ and $p \rightarrow q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,q$
- If $p \in \mathrm{KB}$ and $p \rightarrow q \in \mathrm{KB}$ and $q \rightarrow r \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,r$
- If $p_1, \ldots, p_j \in \mathrm{KB}$ and $p_1 \wedge \ldots \wedge p_j \rightarrow q \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,q$
- If $P(n) \in \mathrm{KB}$ and $\forall x\,(P(x) \rightarrow Q(x)) \in \mathrm{KB}$, then $\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_0\,Q(n)$

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \wedge Q(x))$:

$\mathbf{KB_1}$	$\mathbf{KB_2}$
$P(a) \vee P(e) \vee P(f)$	$P(a) \vee Q(e) \vee Q(c)$
$P(a) \vee P(e) \vee Q(f)$	$Q(d) \vee P(b) \vee Q(a)$
$P(a) \vee P(e) \vee P(c)$	$P(a) \vee P(e) \vee P(f)$
$P(a) \vee P(e) \vee Q(c)$	$P(c) \vee Q(e) \vee P(a)$
$Q(a) \vee P(b) \vee P(d)$	$Q(a) \vee Q(b) \vee Q(g)$
$Q(a) \vee P(b) \vee Q(c)$	$P(a) \vee P(e) \vee Q(f)$
$Q(a) \vee Q(b) \vee P(g)$	$Q(b) \vee Q(a) \vee P(g)$
$Q(a) \vee Q(b) \vee Q(g)$	$Q(a) \vee P(d) \vee P(b)$

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \land Q(x))$:

	KB₁	**KB₂**
✓	$P(a) \lor P(e) \lor P(f)$	$P(a) \lor Q(e) \lor Q(c)$
✓	$P(a) \lor P(e) \lor Q(f)$	$Q(d) \lor P(b) \lor Q(a)$
✓	$P(a) \lor P(e) \lor P(c)$	$P(a) \lor P(e) \lor P(f)$
✓	$P(a) \lor P(e) \lor Q(c)$	$P(c) \lor Q(e) \lor P(a)$
✓	$Q(a) \lor P(b) \lor P(d)$	$Q(a) \lor Q(b) \lor Q(g)$
✓	$Q(a) \lor P(b) \lor Q(c)$	$P(a) \lor P(e) \lor Q(f)$
✓	$Q(a) \lor Q(b) \lor P(g)$	$Q(b) \lor Q(a) \lor P(g)$
✓	$Q(a) \lor Q(b) \lor Q(g)$	$Q(a) \lor P(d) \lor P(b)$

Let $w \models P(a) \land P(d) \land Q(b) \land Q(c)$.
Then $w \models \mathrm{KB}_1$ but $w \not\models \exists x\,(P(x) \land Q(x))$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \wedge Q(x))$:

KB_1	**KB_2**
$P(a) \vee P(e) \vee P(f)$	$P(a) \vee Q(e) \vee Q(c)$
$P(a) \vee P(e) \vee Q(f)$	$Q(d) \vee P(b) \vee Q(a)$
$P(a) \vee P(e) \vee P(c)$	$P(a) \vee P(e) \vee P(f)$
$P(a) \vee P(e) \vee Q(c)$	$P(c) \vee Q(e) \vee P(a)$
$Q(a) \vee P(b) \vee P(d)$	$Q(a) \vee Q(b) \vee Q(g)$
$Q(a) \vee P(b) \vee Q(c)$	$P(a) \vee P(e) \vee Q(f)$
$Q(a) \vee Q(b) \vee P(g)$	$Q(b) \vee Q(a) \vee P(g)$
$Q(a) \vee Q(b) \vee Q(g)$	$Q(a) \vee P(d) \vee P(b)$

Let $w \models \mathrm{KB}_2$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \land Q(x))$:

KB_1	KB_2	
$P(a) \lor P(e) \lor P(f)$	$P(a) \lor Q(e) \lor Q(c)$	✓
$P(a) \lor P(e) \lor Q(f)$	$Q(d) \lor P(b) \lor Q(a)$	
$P(a) \lor P(e) \lor P(c)$	$P(a) \lor P(e) \lor P(f)$	✓
$P(a) \lor P(e) \lor Q(c)$	$P(c) \lor Q(e) \lor P(a)$	✓
$Q(a) \lor P(b) \lor P(d)$	$Q(a) \lor Q(b) \lor Q(g)$	
$Q(a) \lor P(b) \lor Q(c)$	$P(a) \lor P(e) \lor Q(f)$	✓
$Q(a) \lor Q(b) \lor P(g)$	$Q(b) \lor Q(a) \lor P(g)$	
$Q(a) \lor Q(b) \lor Q(g)$	$Q(a) \lor P(d) \lor P(b)$	

Let $w \models \mathrm{KB}_2 \land P(a)$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \land Q(x))$:

$\mathbf{KB_1}$	$\mathbf{KB_2}$	
$P(a) \lor P(e) \lor P(f)$	$P(a) \lor Q(e) \lor Q(c)$	✓
$P(a) \lor P(e) \lor Q(f)$	$Q(d) \lor P(b) \lor Q(a)$	
$P(a) \lor P(e) \lor P(c)$	$P(a) \lor P(e) \lor P(f)$	✓
$P(a) \lor P(e) \lor Q(c)$	$P(c) \lor Q(e) \lor P(a)$	✓
$Q(a) \lor P(b) \lor P(d)$	$Q(a) \lor Q(b) \lor Q(g)$	
$Q(a) \lor P(b) \lor Q(c)$	$P(a) \lor P(e) \lor Q(f)$	✓
$Q(a) \lor Q(b) \lor P(g)$	$Q(b) \lor Q(a) \lor P(g)$	
$Q(a) \lor Q(b) \lor Q(g)$	$Q(a) \lor P(d) \lor P(b)$	

Let $w \models \mathrm{KB}_2 \land P(a) \land \neg Q(a)$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \wedge Q(x))$:

$\mathbf{KB_1}$	$\mathbf{KB_2}$	
$P(a) \vee P(e) \vee P(f)$	$P(a) \vee Q(e) \vee Q(c)$	✓
$P(a) \vee P(e) \vee Q(f)$	$Q(d) \vee P(b) \vee Q(a)$	✓
$P(a) \vee P(e) \vee P(c)$	$P(a) \vee P(e) \vee P(f)$	✓
$P(a) \vee P(e) \vee Q(c)$	$P(c) \vee Q(e) \vee P(a)$	✓
$Q(a) \vee P(b) \vee P(d)$	$Q(a) \vee Q(b) \vee Q(g)$	
$Q(a) \vee P(b) \vee Q(c)$	$P(a) \vee P(e) \vee Q(f)$	✓
$Q(a) \vee Q(b) \vee P(g)$	$Q(b) \vee Q(a) \vee P(g)$	
$Q(a) \vee Q(b) \vee Q(g)$	$Q(a) \vee P(d) \vee P(b)$	✓

Let $w \models \mathrm{KB}_2 \wedge P(a) \wedge \neg Q(a) \wedge P(b)$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \land Q(x))$:

$\mathbf{KB_1}$	$\mathbf{KB_2}$	
$P(a) \lor P(e) \lor P(f)$	$P(a) \lor Q(e) \lor Q(c)$	✓
$P(a) \lor P(e) \lor Q(f)$	$Q(d) \lor P(b) \lor Q(a)$	✓
$P(a) \lor P(e) \lor P(c)$	$P(a) \lor P(e) \lor P(f)$	✓
$P(a) \lor P(e) \lor Q(c)$	$P(c) \lor Q(e) \lor P(a)$	✓
$Q(a) \lor P(b) \lor P(d)$	$Q(a) \lor Q(b) \lor Q(g)$	
$Q(a) \lor P(b) \lor Q(c)$	$P(a) \lor P(e) \lor Q(f)$	✓
$Q(a) \lor Q(b) \lor P(g)$	$Q(b) \lor Q(a) \lor P(g)$	
$Q(a) \lor Q(b) \lor Q(g)$	$Q(a) \lor P(d) \lor P(b)$	✓

Let $w \models \mathrm{KB}_2 \land P(a) \land \neg Q(a) \land P(b) \land \neg Q(b)$.

Hard Inferences

What should not count as explicit belief?

Things that are not obvious (requires to consider different cases).

For example, only one of the following KBs entails $\exists x\,(P(x) \land Q(x))$:

$\mathbf{KB_1}$	$\mathbf{KB_2}$	
$P(a) \lor P(e) \lor P(f)$	$P(a) \lor Q(e) \lor Q(c)$	✓
$P(a) \lor P(e) \lor Q(f)$	$Q(d) \lor P(b) \lor Q(a)$	✓
$P(a) \lor P(e) \lor P(c)$	$P(a) \lor P(e) \lor P(f)$	✓
$P(a) \lor P(e) \lor Q(c)$	$P(c) \lor Q(e) \lor P(a)$	✓
$Q(a) \lor P(b) \lor P(d)$	$Q(a) \lor Q(b) \lor Q(g)$	✓
$Q(a) \lor P(b) \lor Q(c)$	$P(a) \lor P(e) \lor Q(f)$	✓
$Q(a) \lor Q(b) \lor P(g)$	$Q(b) \lor Q(a) \lor P(g)$	✓
$Q(a) \lor Q(b) \lor Q(g)$	$Q(a) \lor P(d) \lor P(b)$	✓

Let $w \models \mathrm{KB}_2 \land P(a) \land \neg Q(a) \land P(b) \land \neg Q(b) \land P(g) \land Q(g)$.
Then $w \models \exists x\,(P(x) \land Q(x))$.

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)
- Level 1: one case split, i.e., branch on p and $\neg p$ for some atom p

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)
- Level 1: one case split, i.e., branch on p and $\neg p$ for some atom p
- Level k: k case splits

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)
- Level 1: one case split, i.e., branch on p and $\neg p$ for some atom p
- Level k: k case splits

Ex.: Let $\mathrm{KB} \stackrel{\text{def}}{=} ((p \vee r) \wedge (q \vee \neg r))$.

- At level 0, we do not know $(p \vee q)$

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)
- Level 1: one case split, i.e., branch on p and $\neg p$ for some atom p
- Level k: k case splits

Ex.: Let $\mathrm{KB} \stackrel{\text{def}}{=} ((p \vee r) \wedge (q \vee \neg r))$.

- At level 0, we do not know $(p \vee q)$
- At level 1, we can split cases for r:
  - $\mathrm{KB} \wedge r \implies q \implies (p \vee q)$
  - $\mathrm{KB} \wedge \neg r \implies p \implies (p \vee q)$

Semantic Model of Belief Levels

- Level 0: only simple inferences (unit propagation, subsumption)
- Level 1: one case split, i.e., branch on p and $\neg p$ for some atom p
- Level k: k case splits

Ex.: Let $\mathrm{KB} \stackrel{\text{def}}{=} ((p \vee r) \wedge (q \vee \neg r))$.

- At level 0, we do not know $(p \vee q)$
- At level 1, we can split cases for r:
  - $\mathrm{KB} \wedge r \implies q \implies (p \vee q)$
  - $\mathrm{KB} \wedge \neg r \implies p \implies (p \vee q)$

Semantic representation: set of clauses instead of set of worlds

- Set of worlds $\approx$ disjunction of conjunctions (DNF)
- Set of clauses $\approx$ conjunction of disjunctions (CNF)
- CNF is often more compact than DNF

Setups, Unit Propagation, Subsumption

- Identify clause $\ell_1 \vee \ldots \vee \ell_j$ with $\{\ell_1, \ldots, \ell_j\}$
- We write $\overline{\ell}$ to flip the sign of ℓ, e.g., $\overline{p}$ is $\neg p$, and $\overline{\neg p}$ is p
- Recall: empty clause is unsatisfiable

Setups, Unit Propagation, Subsumption

- Identify clause $\ell_1 \lor \ldots \lor \ell_j$ with $\{\ell_1, \ldots, \ell_j\}$
- We write $\overline{\ell}$ to flip the sign of ℓ, e.g., $\overline{p}$ is $\neg p$, and $\overline{\neg p}$ is p
- Recall: empty clause is unsatisfiable

Definition: unit propagation, subsumption, setup

A **setup** s is a (possibly infinite) set of ground clauses.
Unit propagation infers $c \setminus \{\overline{\ell}\}$ from c and ℓ.
Subsumption infers $c \cup d$ from c.
$\mathrm{UP}(s)$ closes s under unit propagation.
$\mathrm{UP}^{+}(s)$ adds subsumed clauses.
$\mathrm{UP}^{-}(s)$ removes subsumed clauses.

Examples

Ex.: $c_1 = (p \vee q \vee r)$, $c_2 = (p \vee q \vee \neg r)$

- $\mathrm{UP}(\{c_1, c_2\}) = \{c_1, c_2\}$
- $\mathrm{UP}(\{c_1, c_2, r\}) = \{c_1, c_2, r, (p \vee q)\}$
- $\mathrm{UP}(\{c_1, c_2, \neg r\}) = \{c_1, c_2, \neg r, (p \vee q)\}$
- $\mathrm{UP}^{+}(\{c_1, c_2, \neg r\}) = \{c_1, c_2, \neg r, (p \vee q)\} \cup \{c \mid c \supseteq \neg r \text{ or } c \supseteq (p \vee q)\}$
- $\mathrm{UP}^{-}(\{c_1, c_2, \neg r\}) = \{\neg r, (p \vee q)\}$

Examples

Ex.: $c_1 = (p \vee q \vee r), \;\; c_2 = (p \vee q \vee \neg r)$

- $\mathrm{UP}(\{c_1, c_2\}) = \{c_1, c_2\}$
- $\mathrm{UP}(\{c_1, c_2, r\}) = \{c_1, c_2, r, (p \vee q)\}$
- $\mathrm{UP}(\{c_1, c_2, \neg r\}) = \{c_1, c_2, \neg r, (p \vee q)\}$
- $\mathrm{UP}^{+}(\{c_1, c_2, \neg r\}) = \{c_1, c_2, \neg r, (p \vee q)\} \cup \{c \mid c \supseteq \neg r \text{ or } c \supseteq (p \vee q)\}$
- $\mathrm{UP}^{-}(\{c_1, c_2, \neg r\}) = \{\neg r, (p \vee q)\}$

Unit propagation = forward chaining

$\mathrm{UP}(s)$ can be computed in linear time (if s is finite).

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \mid\!\approx c \iff c \in \mathrm{UP}^{+}(s)$ if c is a clause
- $s \mid\!\approx (\alpha \vee \beta) \iff s \mid\!\approx \alpha$ or $s \mid\!\approx \beta$ if $(\alpha \vee \beta)$ is not a clause

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \approx\!\!\!\!| \; c \iff c \in \mathrm{UP}^+(s)$ if c is a clause
- $s \approx\!\!\!\!| \; (\alpha \vee \beta) \iff s \approx\!\!\!\!| \; \alpha$ or $s \approx\!\!\!\!| \; \beta$ if $(\alpha \vee \beta)$ is not a clause
- $s \approx\!\!\!\!| \; \neg(\alpha \vee \beta) \iff s \approx\!\!\!\!| \; \neg\alpha$ and $s \approx\!\!\!\!| \; \neg\beta$

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \mathrel{|\!\approx} c \iff c \in \mathrm{UP}^{+}(s)$ if c is a clause
- $s \mathrel{|\!\approx} (\alpha \vee \beta) \iff s \mathrel{|\!\approx} \alpha$ or $s \mathrel{|\!\approx} \beta$ if $(\alpha \vee \beta)$ is not a clause
- $s \mathrel{|\!\approx} \neg(\alpha \vee \beta) \iff s \mathrel{|\!\approx} \neg\alpha$ and $s \mathrel{|\!\approx} \neg\beta$
- $s \mathrel{|\!\approx} \neg\neg\alpha \iff s \mathrel{|\!\approx} \alpha$

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \mathrel{|}\!\approx c \iff c \in \mathrm{UP}^{+}(s)$ if c is a clause
- $s \mathrel{|}\!\approx (\alpha \vee \beta) \iff s \mathrel{|}\!\approx \alpha$ or $s \mathrel{|}\!\approx \beta$ if $(\alpha \vee \beta)$ is not a clause
- $s \mathrel{|}\!\approx \neg(\alpha \vee \beta) \iff s \mathrel{|}\!\approx \neg\alpha$ and $s \mathrel{|}\!\approx \neg\beta$
- $s \mathrel{|}\!\approx \neg\neg\alpha \iff s \mathrel{|}\!\approx \alpha$
- $s \mathrel{|}\!\approx \mathbf{K}_0\,\phi \iff s$ is *obviously inconsistent* or $s \mathrel{|}\!\approx \phi$
- $s \mathrel{|}\!\approx \mathbf{K}_{k+1}\,\phi \iff$ for some atomic proposition P,
  (1) $s \cup \{P\} \mathrel{|}\!\approx \mathbf{K}_k\,\phi$ and
  (2) $s \cup \{\neg P\} \mathrel{|}\!\approx \mathbf{K}_k\,\phi$

s is *obviously inconsistent* when $\mathrm{UP}(s)$ contains the empty clause.

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \mathrel{|}\!\!\approx c \iff c \in \mathrm{UP}^+(s)$ if c is a clause
- $s \mathrel{|}\!\!\approx (\alpha \vee \beta) \iff s \mathrel{|}\!\!\approx \alpha$ or $s \mathrel{|}\!\!\approx \beta$ if $(\alpha \vee \beta)$ is not a clause
- $s \mathrel{|}\!\!\approx \neg(\alpha \vee \beta) \iff s \mathrel{|}\!\!\approx \neg\alpha$ and $s \mathrel{|}\!\!\approx \neg\beta$
- $s \mathrel{|}\!\!\approx \neg\neg\alpha \iff s \mathrel{|}\!\!\approx \alpha$
- $s \mathrel{|}\!\!\approx \mathbf{K}_0 \phi \iff s$ is *obviously inconsistent* or $s \mathrel{|}\!\!\approx \phi$
- $s \mathrel{|}\!\!\approx \mathbf{K}_{k+1} \phi \iff$ for some atomic proposition P,
  (1) $s \cup \{P\} \mathrel{|}\!\!\approx \mathbf{K}_k \phi$ and
  (2) $s \cup \{\neg P\} \mathrel{|}\!\!\approx \mathbf{K}_k \phi$

- $s \mathrel{|}\!\!\approx \mathbf{O}\phi \iff s \mathrel{|}\!\!\approx \phi$ and $s' \not\mathrel{|}\!\!\approx \phi$ for all s' with $\mathrm{UP}^+(s') \subsetneq \mathrm{UP}^+(s)$

s is *obviously inconsistent* when $\mathrm{UP}(s)$ contains the empty clause.

Semantics of Limited Belief

Definition: semantics of limited belief

- $s \approx c \iff c \in \mathrm{UP}^{+}(s)$ — if c is a clause
- $s \approx (\alpha \vee \beta) \iff s \approx \alpha$ or $s \approx \beta$ — if $(\alpha \vee \beta)$ is not a clause
- $s \approx \neg(\alpha \vee \beta) \iff s \approx \neg\alpha$ and $s \approx \neg\beta$
- $s \approx \neg\neg\alpha \iff s \approx \alpha$
- $s \approx \mathbf{K}_0 \phi \iff s$ is *obviously inconsistent* or $s \approx \phi$
- $s \approx \mathbf{K}_{k+1} \phi \iff$ for some atomic proposition P,
  (1) $s \cup \{P\} \approx \mathbf{K}_k \phi$ and
  (2) $s \cup \{\neg P\} \approx \mathbf{K}_k \phi$
- $s \approx \neg\mathbf{K}_k \phi \iff s \not\approx \mathbf{K}_k \phi$
- $s \approx \mathbf{O}\phi \iff s \approx \phi$ and $s' \not\approx \phi$ for all s' with $\mathrm{UP}^{+}(s') \subsetneq \mathrm{UP}^{+}(s)$
- $s \approx \neg\mathbf{O}\phi \iff s \not\approx \mathbf{O}\phi$

s is *obviously inconsistent* when $\mathrm{UP}(s)$ contains the empty clause.

Examples

Let $s \mathrel{|}\!\!\approx \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$

Examples

Let $s \approx\!\!\!\!\models \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!\models \mathbf{K}_0(p \vee q)$

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \lor q)$
  $\iff$ s is obv. inconsistent or $s \mathrel{|\!\!\approx} (p \lor q)$

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$

  $\iff$ s is obv. inconsistent or $s \mathrel{|\!\!\approx} (p \vee q)$

  $\iff$ s is obv. inconsistent or $(p \vee q) \in \mathsf{UP}^+(s)$ ✗

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!| \; \mathbf{K}_0(p \vee q)$ ✗

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\, (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \lor q)$

$\iff$ for some atom P, (1) and (2) succeed:

(1) $s \cup \{P\} \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$

(2) $s \cup \{\neg P\} \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$

Examples

Let $s \approx\!\!\!\models \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\models \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\!\!\!\models \mathbf{K}_1(p \vee q)$

$\iff$ for some atom P, (1) and (2) succeed:

(1) $s \cup \{P\} \approx\!\!\!\models \mathbf{K}_0(p \vee q)$

$\iff$ $s \cup \{P\}$ is obv. inconsistent or $(p \vee q) \in \mathsf{UP}^+(s \cup \{P\})$

(2) $s \cup \{\neg P\} \approx\!\!\!\models \mathbf{K}_0(p \vee q)$

$\iff$ $s \cup \{\neg P\}$ is obv. inconsistent or $(p \vee q) \in \mathsf{UP}^+(s \cup \{\neg P\})$

Examples

Let $s \approx\!\!\!\models \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathrm{UP}^+(s) = \mathrm{UP}^+(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \approx\!\!\!\models \mathbf{K}_0(p \lor q)$ ✗
- $s \approx\!\!\!\models \mathbf{K}_1(p \lor q)$

$\Longleftarrow$ splitting on r succeeds:

(1) $s \cup \{r\} \approx\!\!\!\models \mathbf{K}_0(p \lor q)$

$\Longleftrightarrow$ $s \cup \{r\}$ is obv. inconsistent or $(p \lor q) \in \mathrm{UP}^+(s \cup \{r\})$

(2) $s \cup \{\neg r\} \approx\!\!\!\models \mathbf{K}_0(p \lor q)$

$\Longleftrightarrow$ $s \cup \{\neg r\}$ is obv. inconsistent or $(p \lor q) \in \mathrm{UP}^+(s \cup \{\neg r\})$

Examples

Let $s \approx\models \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathrm{UP}^{+}(s) = \mathrm{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \approx\models \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\models \mathbf{K}_1(p \vee q)$

$\Longleftarrow$ splitting on r succeeds:

(1) $s \cup \{r\} \approx\models \mathbf{K}_0(p \vee q)$

$\Longleftrightarrow$ $s \cup \{r\}$ is obv. inconsistent or $(p \vee q) \in \mathrm{UP}^{+}(s \cup \{r\})$ ✓
because UP infers $(p \vee q)$ from $(p \vee q \vee \neg r)$ and r

(2) $s \cup \{\neg r\} \approx\models \mathbf{K}_0(p \vee q)$

$\Longleftrightarrow$ $s \cup \{\neg r\}$ is obv. inconsistent or $(p \vee q) \in \mathrm{UP}^{+}(s \cup \{\neg r\})$ ✓
because UP infers $(p \vee q)$ from $(p \vee q \vee r)$ and $\neg r$

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 p$

Examples

Let $s \approx\!\!\!\!| \;\mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \neg\mathbf{K}_0 p$

$\iff$ s is not obv. inconsistent and $p \notin \mathsf{UP}^{+}(s)$ ✓

Examples

Let $s \approx\!\!\!\!\models \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!\models \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\!\!\!\!\models \mathbf{K}_1(p \vee q)$ ✓
- $s \approx\!\!\!\!\models \neg\mathbf{K}_0 p$ ✓

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0 (p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1 (p \vee q)$ ✓
- $s \mathrel{|\!\approx} \neg \mathbf{K}_0 p$ ✓
- $s \mathrel{|\!\approx} \neg \mathbf{K}_0 \neg p$

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 p$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 \neg p$
  $\iff$ s is not obv. inconsistent and $\neg p \notin \mathsf{UP}^{+}(s)$ ✓

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \lor q)$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 p$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 \neg p$ ✓

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \lor q \lor r), (p \lor q \lor \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \lor q)$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 p$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 \neg p$ ✓

But what about $\neg\mathbf{K}_1\neg p$? And $\neg\mathbf{K}_2\neg p$? And so on?

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 p$ ✓
- $s \mathrel{|\!\approx} \neg\mathbf{K}_0 \neg p$ ✓

But what about $\neg\mathbf{K}_1 \neg p$? And $\neg\mathbf{K}_2 \neg p$? And so on?

$\mathbf{K}_k$ is incomplete (see first example).
So how to find out with certainty that p is unknown?

Examples

Let $s \approx\!\!\!\!\!\models \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!\!\models \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\!\!\!\!\!\models \mathbf{K}_1(p \vee q)$ ✓
- $s \approx\!\!\!\!\!\models \neg\mathbf{K}_0 p$ ✓
- $s \approx\!\!\!\!\!\models \neg\mathbf{K}_0 \neg p$ ✓

But what about $\neg\mathbf{K}_1 \neg p$? And $\neg\mathbf{K}_2 \neg p$? And so on?

$\mathbf{K}_k$ is incomplete (see first example).
So how to find out with certainty that p is unknown?

Need a dual operator to $\mathbf{K}_k \phi$, call it $\mathbf{M}_k \phi$, to say that ϕ is possible.

Semantics of Limited Belief (2)

The semantics of unlimited $\mathbf{M}\,\alpha$ in $\mathcal{OL}$ is:

Definition: semantics $\mathbf{M}$

- $e, w \models \mathbf{M}\,\alpha \iff$ for some w, $w \in e$ and $e, w \models \alpha$

<u>Note</u>: $e, w \models \mathbf{M}\,\alpha \iff e, w \models \neg\mathbf{K}\neg\alpha$

Definition: semantics $\mathbf{M}_k$

- $s \mathrel{|\!\approx} \mathbf{M}_0\,\phi \iff s$ is *obviously consistent* and $s \mathrel{|\!\approx} \phi$
- $s \mathrel{|\!\approx} \mathbf{M}_{k+1}\,\phi \iff$ for some literal L, $s \cup \{L\} \mathrel{|\!\approx} \mathbf{M}_k\,\phi$
- $s \mathrel{|\!\approx} \neg\mathbf{M}_k\,\phi \iff s \mathrel{|\!\not\approx} \mathbf{M}_k\,\phi$

s is *obviously consistent* when $\mathrm{UP}^{-}(s)$ does not contain the empty clause and does not contain any clauses that contain complementary literals ($c_1, c_2 \in \mathrm{UP}^{-}(s)$, $P \in c_1$, $\neg P \in c_2$ for some P)

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!| \; \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\!\!\!\!| \; \mathbf{K}_1(p \vee q)$ ✓
- $s \approx\!\!\!\!| \; \mathbf{M}_0 p$

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$

  $\iff$ s is obv. consistent and $s \mathrel{|\!\!\approx} p$

Examples

Let $s \approx\!\!\!\!| \ \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathrm{UP}^+(s) = \mathrm{UP}^+(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \approx\!\!\!\!| \ \mathbf{K}_0(p \vee q)$ ✗
- $s \approx\!\!\!\!| \ \mathbf{K}_1(p \vee q)$ ✓
- $s \approx\!\!\!\!| \ \mathbf{M}_0 p$

$\Longleftrightarrow$ s is obv. consistent and $s \approx\!\!\!\!| \ p$

$\Longleftrightarrow$ s is obv. consistent and $p \in \mathrm{UP}^+(s)$ ✗

because s is not obv. consistent (r occurs pos. and neg. in $\mathrm{UP}^-(s)$) and also $p \notin \mathrm{UP}^+(s)$

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\approx} \mathbf{M}_0 p$ ✗

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{M}_1 p$

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{M}_1 p$

  $\iff$ for some atom P, $s \cup \{P\} \mathrel{|\!\!\approx} \mathbf{M}_0 p$

Examples

Let $s \approx\!\!\!\!| \; \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0 (p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1 (p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{M}_1 p$

  $\iff$ for some atom P, $s \cup \{P\} \mathrel{|\!\!\approx} \mathbf{M}_0 p$

  $\iff$ $s \cup \{P\}$ is obv. consistent and $s \cup \{P\} \mathrel{|\!\!\approx} p$

Examples

Let $s \approx\!\!\!\!\mid\; \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathrm{UP}^+(s) = \mathrm{UP}^+(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \mid\!\approx \mathbf{K}_0(p \lor q)$ ✗
- $s \mid\!\approx \mathbf{K}_1(p \lor q)$ ✓
- $s \mid\!\approx \mathbf{M}_0 p$ ✗
- $s \mid\!\approx \mathbf{M}_1 p$

  $\Longleftarrow$ $s \cup \{p\} \mid\!\approx \mathbf{M}_0 p$

  $\Longleftrightarrow$ $s \cup \{p\}$ is obv. consistent and $s \cup \{p\} \mid\!\approx p$ ✓
  because s is obv. consistent ($\mathrm{UP}^-(s \cup \{p\}) = \{p\}$)
  and $p \in \mathrm{UP}^+(s \cup \{p\})$

Examples

Let $s \mathrel{|\!\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^{+}(s) = \mathsf{UP}^{+}(\{(p \vee q \vee r), (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\!\approx} \mathbf{M}_1 p$ ✓

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \lor q \lor r) \land (p \lor q \lor \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \lor q \lor r),\ (p \lor q \lor \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \lor q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \lor q)$ ✓
- $s \mathrel{|\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\approx} \mathbf{M}_1 p$ ✓
- $s \mathrel{|\!\approx} \mathbf{M}_0 \neg p$ ✗

Examples

Let $s \mathrel{|\!\approx} \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|\!\approx} \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|\!\approx} \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|\!\approx} \mathbf{M}_0 p$ ✗
- $s \mathrel{|\!\approx} \mathbf{M}_1 p$ ✓
- $s \mathrel{|\!\approx} \mathbf{M}_0 \neg p$ ✗
- $s \mathrel{|\!\approx} \mathbf{M}_1 \neg p$ ✗

Examples

Let $s \mathrel{|}\!\!\approx \mathbf{O}((p \vee q \vee r) \wedge (p \vee q \vee \neg r))$.

- $\mathsf{UP}^+(s) = \mathsf{UP}^+(\{(p \vee q \vee r),\ (p \vee q \vee \neg r)\})$
- $s \mathrel{|}\!\!\approx \mathbf{K}_0(p \vee q)$ ✗
- $s \mathrel{|}\!\!\approx \mathbf{K}_1(p \vee q)$ ✓
- $s \mathrel{|}\!\!\approx \mathbf{M}_0 p$ ✗
- $s \mathrel{|}\!\!\approx \mathbf{M}_1 p$ ✓
- $s \mathrel{|}\!\!\approx \mathbf{M}_0 \neg p$ ✗
- $s \mathrel{|}\!\!\approx \mathbf{M}_1 \neg p$ ✗
- $s \mathrel{|}\!\!\approx \mathbf{M}_2 \neg p$ ✓

Some Properties

Theorem: monotonicity

$\approx \mathbf{K}_k \phi \rightarrow \mathbf{K}_{k+1} \phi$.
$\approx \mathbf{M}_k \phi \rightarrow \mathbf{M}_{k+1} \phi$.

Definition: proper$^+$ KB

A KB is proper$^+$ when it is a conjunction of clauses (CNF).

Let KB be proper$^+$ of the form $c_1 \wedge \ldots \wedge c_j$.

Theorem: unique-model property

$s \approx \mathbf{O}\mathrm{KB} \iff \mathrm{UP}^+(s) = \mathrm{UP}^+(\{c_1, \ldots, c_j\})$.

Some Properties (2)

Let KB be proper⁺.

Theorem: soundness

$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_k\,\phi \implies \mathbf{O}\mathrm{KB} \models \mathbf{K}\,\phi$.
$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{M}_k\,\phi \implies \mathbf{O}\mathrm{KB} \models \mathbf{M}\,\phi$.

Theorem: eventual completeness

$\mathbf{O}\mathrm{KB} \models \mathbf{K}\,\phi \implies \mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_k\,\phi$ for large enough k.
$\mathbf{O}\mathrm{KB} \models \mathbf{M}\,\phi \implies \mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{M}_k\,\phi$ for large enough k.

Theorem: complexity

$\mathbf{O}\mathrm{KB} \models \mathbf{K}\,\phi$ and $\mathbf{O}\mathrm{KB} \models \mathbf{M}\,\phi$ is tractable for small k:

$$\mathcal{O}(2^k \cdot (|\mathrm{KB}| + |\phi|)^{k+3}).$$

Generalisation of the Logic of Limited Belief

- Introspection
  - Extend semantics to keep track of original setup without splits
  - Representation theorem translates to limited belief

- First-order logic
  - $s \mathrel{|\!\approx} \exists x\, \phi \iff s \mathrel{|\!\approx} \phi^x_n$ for some n
  - Proper$^+$ means CNF without $\exists$, i.e., $\forall \vec{x} \bigwedge_i c_i$

Generalisation of the Logic of Limited Belief

- Introspection
  - Extend semantics to keep track of original setup without splits
  - Representation theorem translates to limited belief

- First-order logic
  - $s \mathrel{|\!\!\approx} \exists x\, \phi \iff s \mathrel{|\!\!\approx} \phi^x_n$ for some n
  - Proper$^+$ means CNF without $\exists$, i.e., $\forall \vec{x} \bigwedge_i c_i$

Let KB be proper$^+$.

Theorem: soundness

$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{K}_k\, \phi \implies \mathbf{O}\mathrm{KB} \models \mathbf{K}_k\, \phi.$
$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \mathbf{M}_k\, \phi \implies \mathbf{O}\mathrm{KB} \models \mathbf{M}_k\, \phi.$

Theorem: decidability

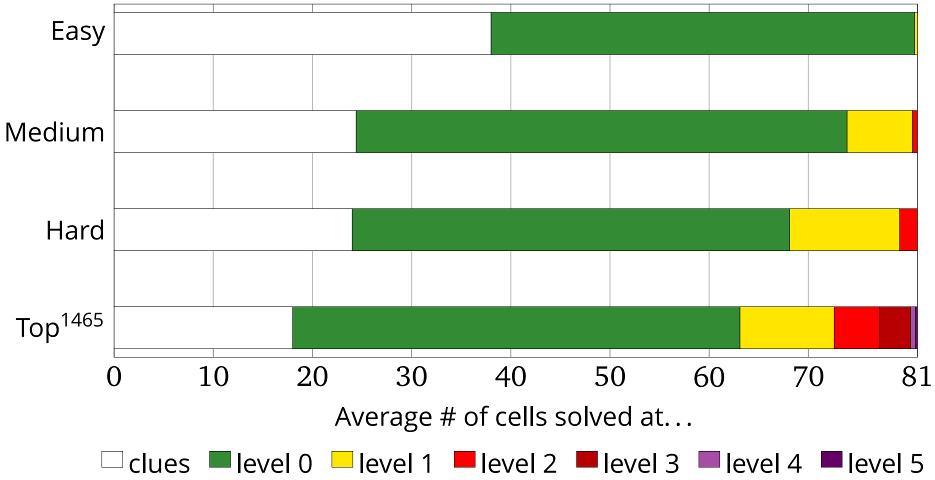
$\mathbf{O}\mathrm{KB} \mathrel{|\!\!\approx} \sigma$ is decidable.

Does Limited Belief Work?

Experiment: Sudoku

- Fill 9×9 board with numbers $1, \ldots, 9$ such that no identical numbers in rows, columns, 3×3 blocks
- Has a unique solution
- Difficulty depends on how many and which clues we get
  - Newspaper: easy ($\approx$ 38 clues), medium ($\approx$ 24 clues), hard ($\approx$ 24 clues)
  - Top 1465: extremely difficult (18 clues, proven minimum is 17)
- Question: do belief level and difficulty correlate?

Sudoku with Limited Belief



Overview of the Lecture

- Limited Belief – First Attempt
- Limited Belief – Second Attempt
- **Implementation Techniques**

Implementation of a Solver

- Same techniques can be used for
  - Propositional Satisfiability
  - Answer Set Programs
  - Limited Belief

Implementation of a Solver

- Same techniques can be used for
  - Propositional Satisfiability
  - Answer Set Programs
  - Limited Belief
- Solvers operate on sets of ground **clauses**
  - Recall: a clause is a disjunction of literals
  - SAT: find a model
  - ASP: find a stable model
  - Limited Belief: case splits, subsumption

Implementation of a Solver

- Same techniques can be used for
  - Propositional Satisfiability
  - Answer Set Programs
  - Limited Belief
- Solvers operate on sets of ground **clauses**
  - Recall: a clause is a disjunction of literals
  - SAT: find a model
  - ASP: find a stable model
  - Limited Belief: case splits, subsumption
- Data structures and algorithms:
  - Davis-Putnam-Logemann-Loveland (DPLL) algorithm
  - Watched-Literal Scheme
  - Conflict-Driven Clause Learning (CDCL)

While SAT is NP-complete for propositional logic and for ASP, modern solvers can solve large instances (millions of variables).

DPLL Algorithm

Definition: DPLL algorithm

A literal ℓ is **assigned** in s iff $\ell \in s$ or $\overline{\ell} \in s$.

Input: set of clauses s
Output: 1 iff s is satisfiable in propositional logic

DPLL(s) procedure:

1. If s contains the empty clause, return 0
2. If all literals are assigned in s, return 1
3. Select some unassigned literal ℓ
4. Return $\min\{\text{DPLL}(\text{UP}(s \cup \{\ell\})), \text{DPLL}(\text{UP}(s \cup \{\overline{\ell}\}))\}$

DPLL Algorithm

Definition: DPLL algorithm

A literal ℓ is **assigned** in s iff $\ell \in s$ or $\overline{\ell} \in s$.

Input: set of clauses s
Output: 1 iff s is satisfiable in propositional logic

DPLL(s) procedure:

1. If s contains the empty clause, return 0
2. If all literals are assigned in s, return 1
3. Select some unassigned literal ℓ
4. Return $\min\{\text{DPLL}(\text{UP}(s \cup \{\ell\})), \text{DPLL}(\text{UP}(s \cup \{\overline{\ell}\}))\}$

Theorem: sound and complete

DPLL is sound and complete for SAT in propositional logic.

DPLL Algorithm

Definition: DPLL algorithm

A literal ℓ is **assigned** in s iff $\ell \in s$ or $\bar{\ell} \in s$.

Input: set of clauses s
Output: 1 iff s is satisfiable in propositional logic

DPLL(s) procedure:

1. If s contains the empty clause, return 0
2. If all literals are assigned in s, return 1
3. Select some unassigned literal ℓ
4. Return $\min\{\text{DPLL}(\text{UP}(s \cup \{\ell\})), \text{DPLL}(\text{UP}(s \cup \{\bar{\ell}\}))\}$

Theorem: sound and complete

DPLL is sound and complete for SAT in propositional logic.

How to select literal? Prefer ones that trigger UP

Watched-Literal Scheme

- DPLL uses backtracking:
  1. Add ℓ to s, close under unit propagation
  2. (recursive calls)
  3. Remove ℓ from s, undo the unit propagation

Watched-Literal Scheme

- DPLL uses backtracking:
  1. Add ℓ to s, close under unit propagation
  2. (recursive calls)
  3. Remove ℓ from s, undo the unit propagation

- Watched-literal scheme
  - s implements as stack
  - Step 1 pushes onto s, leaves old clauses unchanged
  - Step 3 pops from (shrinks) s
  - s is kept closed under UP

Watched-Literal Scheme

- DPLL uses backtracking:
    1. Add ℓ to s, close under unit propagation
    2. (recursive calls)
    3. Remove ℓ from s, undo the unit propagation

- Watched-literal scheme
    - s implements as stack
    - Step 1 pushes onto s, leaves old clauses unchanged
    - Step 3 pops from (shrinks) s
    - s is kept closed under UP

Watched-Literal Scheme: Observation

Let s be a set of clauses which is closed under unit propagation. Let $c \in s$ with $|c| \geq 2$. If c contains at least two unassigned literals, select two of them as *watched* literals; otherwise select two of them randomly. When we add a new literal ℓ to s, then unit propagation of c with all the unit clauses in s together with ℓ produces a new unit clause *only if* $\overline{\ell}$ is one of the watched literals.

Why?

- Case 1: Suppose there are two unassigned literals in c that are not assigned initially. Then the watched literals ℓ_1, ℓ_2 are unassigned. Suppose unit propagation of c with all the unit clauses in s together with ℓ produces a new unit clause c'. Then $|c'| = 1 < |c|$, so either $\ell_1 \notin c$ or $\ell_2 \notin c$. Since ℓ_1, ℓ_2 were not assigned before adding ℓ, either ℓ_1 or ℓ_2 must be $\overline{\ell}$.
- Case 2: There are no two unassigned literals in c. Then there is at most one unassigned literal ℓ_1 in c, in which case we have ℓ_1 already as a unit clause in s.

Watched-Literal Scheme

Definition: watched literals

For every clause $c \in s$ with $|c| \geq 2$, mark two literals as **watched**.

Watched-Literal Scheme

Definition: watched literals

For every clause $c \in s$ with $|c| \geq 2$, mark two literals as **watched**.

AddUnit(ℓ) procedure:

1. Push ℓ onto s.
2. If $\overline{\ell} \in s$, mark s as inconsistent and return.
3. For every $c \in s$ with $|c| \geq 2$, check if $\overline{\ell}$ is watched.
   If yes, propagate the unit clauses from s with c to infer c'.
   If $|c'| = 0$, mark s as inconsistent.
   If $|c'| = 1$, add c' to s (i.e., recursive call to **AddUnit**(c')).
   If $|c'| > 1$, select literals from c' as new watched literals for c.

Watched-Literal Scheme

Definition: watched literals

For every clause $c \in s$ with $|c| \geq 2$, mark two literals as **watched**.

AddUnit(ℓ) procedure:

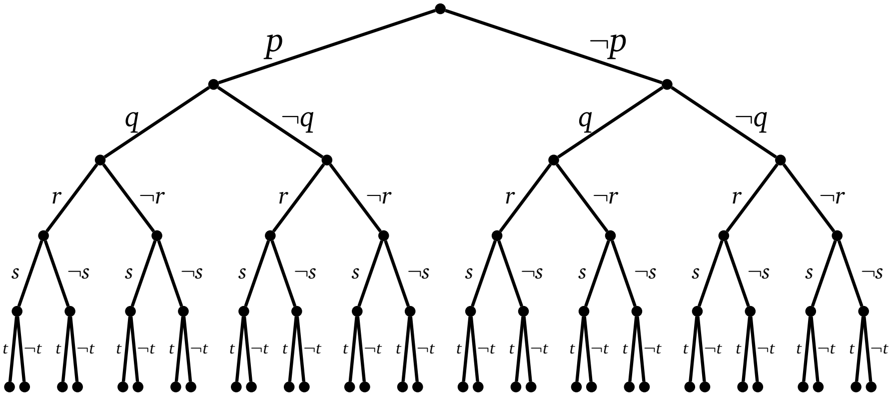
1. Push ℓ onto s.
2. If $\overline{\ell} \in s$, mark s as inconsistent and return.
3. For every $c \in s$ with $|c| \geq 2$, check if $\overline{\ell}$ is watched.
   If yes, propagate the unit clauses from s with c to infer c'.
   If $|c'| = 0$, mark s as inconsistent.
   If $|c'| = 1$, add c' to s (i.e., recursive call to **AddUnit**(c')).
   If $|c'| > 1$, select literals from c' as new watched literals for c.

Backtrack procedure:

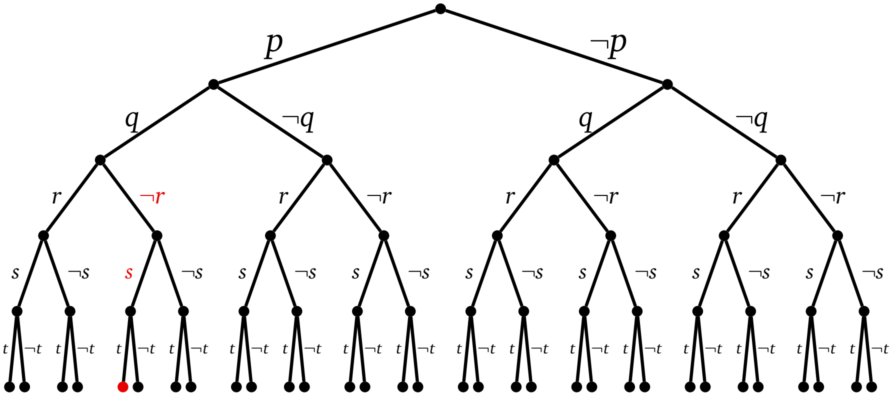
1. Store $n := |s|$
2. (recursive calls)
3. Pop from s until $|s| = n$

Example on paper

Conflict-Driven Clause Learning

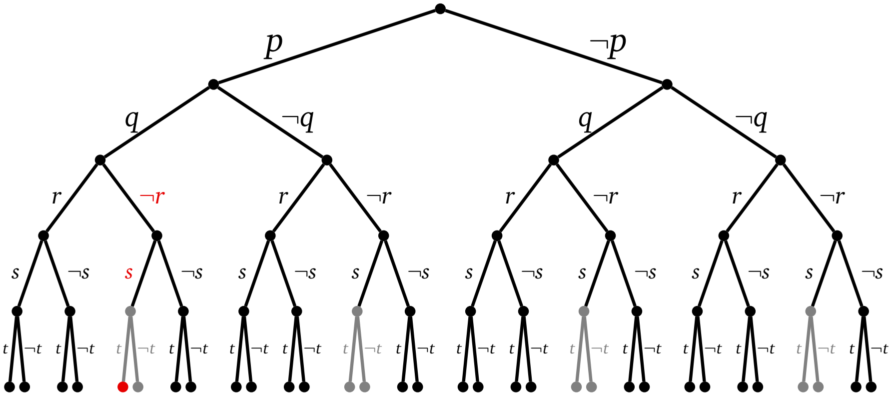


Conflict-Driven Clause Learning



Conflict caused by $\neg r$ and s! Add conflict clause $(r \vee \neg s)$.

Conflict-Driven Clause Learning



Conflict caused by $\neg r$ and s! Add conflict clause $(r \vee \neg s)$.

Implicitly prunes future subtrees.

Conflict-Driven Clause Learning

- Learn from conflicts
  - Why did a conflict occur?
  - Analyse, find reason, feed back negation of reason to s

Conflict-Driven Clause Learning

- Learn from conflicts
  - Why did a conflict occur?
  - Analyse, find reason, feed back negation of reason to s

- Consider a conflict caused by $\ell_1, \ldots, \ell_k$
  - UP of $\ell_1, \ldots, \ell_k$ with $c_1 \in s$ infers ℓ_{k+1}
  - UP of $\ell_1, \ldots, \ell_k$ with $c_2 \in s$ infers $\overline{\ell}_{k+1}$

  Then s entails $\overline{\ell}_1 \vee \ldots \vee \overline{\ell}_k$

Conflict-Driven Clause Learning

- Learn from conflicts
  - Why did a conflict occur?
  - Analyse, find reason, feed back negation of reason to s

- Consider a conflict caused by $\ell_1, \ldots, \ell_k$
  - UP of $\ell_1, \ldots, \ell_k$ with $c_1 \in s$ infers ℓ_{k+1}
  - UP of $\ell_1, \ldots, \ell_k$ with $c_2 \in s$ infers $\overline{\ell}_{k+1}$

  Then s entails $\overline{\ell}_1 \vee \ldots \vee \overline{\ell}_k$

Theorem: sound and complete

DPLL + CDCL is sound and complete for SAT in propositional logic.

Example on paper