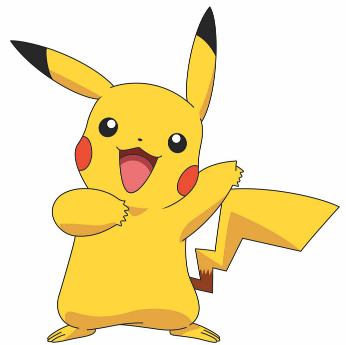
Sometimes Counting is not Enough: Why Model Enumeration Matters

Giuseppe Spallitta

NII Shonan Meeting 238
"Model counting: Theory meets practice"

Speaking about Pokémon...



Why the Pokémon motto:

"Gotta catch'em all"?

Because knowing **how many** Pokémon exist is sometimes not enough, in particular if we do not explicitly know who they are and what they can do.

SAT enumeration (AllSAT)

AllSAT

Enumerate all the models of a Propositional formula φ:

- a set of *total* models $\mathcal{TTA}(\varphi) \stackrel{\text{def}}{=} \{\eta_1, ..., \eta_n\}$
- a set of *partial* models $\mathcal{TA}(\varphi) \stackrel{\text{def}}{=} \{\mu_1, ..., \mu_k\}$ s.t.:
  - $\mu \models \varphi$ for all $\mu \in \mathcal{TA}(\varphi)$;
  - every $\eta \in \mathcal{TTA}(\varphi)$ is a super-assignment of some $\mu \in \mathcal{TA}(\varphi)$;

All-SAT: example

Let $\varphi = A_1 \vee A_2$:

- $\mathcal{TTA}(\varphi) = \{(A_1 \wedge A_2), (A_1 \wedge \neg A_2), (\neg A_1 \wedge A_2)\}$
- $\mathcal{TA}(\varphi)$ with repetitions $= \{A_1, A_2\}$

SAT enumeration (AllSAT)

Disjoint AllSAT

Enumerate all the models of a Propositional formula φ:

- a set of *total* models $\mathcal{TTA}(\varphi) \stackrel{\text{def}}{=} \{\eta_1, ..., \eta_n\}$
- a set of *disjoint* *partial* models $\mathcal{TA}(\varphi) \stackrel{\text{def}}{=} \{\mu_1, ..., \mu_k\}$ s.t.:
  - $\mu \models \varphi$ for all $\mu \in \mathcal{TA}(\varphi)$;
  - every $\eta \in \mathcal{TTA}(\varphi)$ is a super-assignment of some $\mu \in \mathcal{TA}(\varphi)$;
  - every pair $\mu_i, \mu_j \in \mathcal{TA}(\varphi)$ is mutually exclusive.

All-SAT: example

Let $\varphi = A_1 \vee A_2$:

- $\mathcal{TTA}(\varphi) = \{(A_1 \wedge A_2), (A_1 \wedge \neg A_2), (\neg A_1 \wedge A_2)\}$
- $\mathcal{TA}(\varphi)$ with repetitions $= \{A_1, A_2\}$
- $\mathcal{TA}(\varphi)$ without repetitions $= \{A_1, (\neg A_1 \wedge A_2)\}$ or $\{A_2, (A_1 \wedge \neg A_2)\}$

SAT enumeration (AllSAT)

Disjoint AllSAT

Enumerate all the models of a Propositional formula φ:

- a set of *total* models $\mathcal{TTA}(\varphi) \stackrel{\text{def}}{=} \{\eta_1, ..., \eta_n\}$
- a set of *disjoint* *partial* models $\mathcal{TA}(\varphi) \stackrel{\text{def}}{=} \{\mu_1, ..., \mu_k\}$ s.t.:
  - $\mu \models \varphi$ for all $\mu \in \mathcal{TA}(\varphi)$;
  - every $\eta \in \mathcal{TTA}(\varphi)$ is a super-assignment of some $\mu \in \mathcal{TA}(\varphi)$;
  - every pair $\mu_i, \mu_j \in \mathcal{TA}(\varphi)$ is mutually exclusive.

All-SAT: example

Let $\varphi = A_1 \vee A_2$:

- $\mathcal{TTA}(\varphi) = \{(A_1 \wedge A_2), (A_1 \wedge \neg A_2), (\neg A_1 \wedge A_2)\}$
- $\mathcal{TA}(\varphi)$ with repetitions $= \{A_1, A_2\}$
- $\mathcal{TA}(\varphi)$ without repetitions $= \{A_1, (\neg A_1 \wedge A_2)\}$ or $\{A_2, (A_1 \wedge \neg A_2)\}$

Notice: each partial model μ represents 2^K total models, where K is the number of unassigned atoms in μ
$\Rightarrow$ we want to enumerate partial models that are as short as possible

Why enumeration is interesting

Model counting already taught us a lesson

Performance is a combination of multiple factors: encodings, algorithms, preprocessing, and postprocessing can change runtime by orders of magnitude.

What about AllSAT?

Roadmap of this talk

- **Theory / solver view:** which design choices lead to effective and efficient enumeration?
- **Application view:** what does enumeration *enable* beyond just listing models?

We will illustrate these points with some of my recent results: design principles for enumeration, and new applications that benefit from them.

Outline

1. Backtracking approaches for enumeration

2. On CNF conversions for disjoint SAT and SMT enumeration

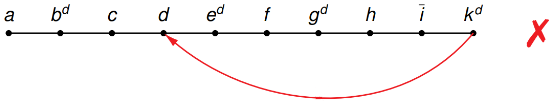
3. Weighted Model Integration via AllSMT

4. A simple framework for Canonical DDs Modulo Theories via AllSMT Enumeration

Blocking vs non-blocking AllSAT solvers

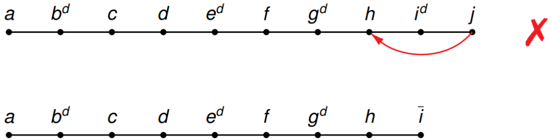
Blocking AllSAT solvers

- Relying on CDCL and non-chronological backtracking [1,2];
- Adding **blocking clauses** to force an assignment not to be repeated;



Non-blocking AllSAT solvers

- Backtrack on the most recent decision variable after a conflict [3].
- No blocking clause is introduced



1 H. Jin, H. Han, and F. Somenzi (2005). "Efficient Conflict Analysis for Finding All Satisfying Assignments of a Boolean Circuit". In: *Tools and Algorithms for the Construction and Analysis of Systems*. Springer Berlin Heidelberg

2 Y. Yu, P. Subramanyan, N. Tsiskaridze, and S. Malik (2014). *All-SAT using minimal blocking clauses*. In: *2014 27th International Conference on VLSI Design and 2014 13th International Conference on Embedded Systems*. IEEE

3 O. Grumberg, A. Schuster, and A. Yadgar (2004). "Memory Efficient All-Solutions SAT Solver and Its Application for Reachability Analysis". In: *Formal Methods in Computer-Aided Design*. Springer Berlin Heidelberg

Blocking VS non-blocking AllSAT solving

Blocking AllSAT solving

- Blocking clauses required to avoid repetitions; but they negatively affect unit propagation in the long term.

Non-blocking AllSAT solving

+ Blocking clause are no more required; we can mitigate the side effects on unit propagation

Blocking VS non-blocking AllSAT solving

Blocking AllSAT solving

- \- Blocking clauses required to avoid repetitions; but they negatively affect unit propagation in the long term.
- \+ Can easily escape regions of the search space with no solution thanks to CDCL;

Non-blocking AllSAT solving

- \+ Blocking clause are no more required; we can mitigate the side effects on unit propagation
- \- Regions of the search space with no solution cannot be escaped easily;

Blocking VS non-blocking AllSAT solving

Blocking AllSAT solving

- \- Blocking clauses required to avoid repetitions; but they negatively affect unit propagation in the long term.
- \+ Can easily escape regions of the search space with no solution thanks to CDCL;
- \+ Easy to retrieve partial assignments.

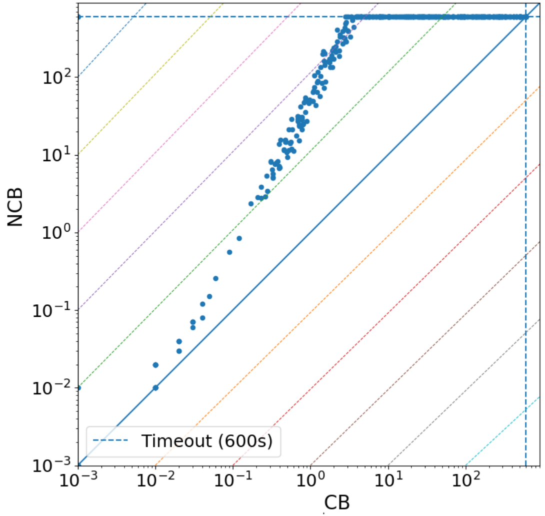
Non-blocking AllSAT solving

- \+ Blocking clause are no more required; we can mitigate the side effects on unit propagation
- \- Regions of the search space with no solution cannot be escaped easily;
- \- Difficult to retrieve partial assignments.

Core idea: CDCL with chronological BT and shrinking [4]

AllSAT engine

- Extend a **CDCL** solver with **chronological backtracking**
  - CDCL: still escapes unsatisfiable regions
  - Chronological backtracking: guarantees full systematic exploration
- Partial assignments are generated from total assignments, removing non-relevant variables...
- ... but adapted to chronological backtracking (yeah, the order slightly affects performances)



[4] G. Spallitta, R. Sebastiani, and A. Biere (2024a). *Disjoint Partial Enumeration without Blocking Clauses*. In: *AAAI*

From AllSAT to projected AllSAT and AllSMT [5]

Projected AllSAT (TabularAllSAT+)

- Separate variables into **relevant** vs **non-relevant** (projection set)
- Decision heuristic prioritizes **relevant** variables
- Non-relevant literals are skipped or stored only when needed

AllSMT (TabularAllSMT)

- Add $\mathcal{T}$-checking and $\mathcal{T}$-propagation (e.g., MathSAT) after Boolean unit propagation
- New $\mathcal{T}$-atoms are treated as non-relevant Boolean variables
- Same engine works for:
  - **AllSAT**, **projected AllSAT**, and **AllSMT**

[5] G. Spallitta, R. Sebastiani, and A. Biere (2024b). *Disjoint Projected Enumeration for SAT and SMT without Blocking Clauses*. In: *AIJ*

Outline

1. Backtracking approaches for enumeration

2. On CNF conversions for disjoint SAT and SMT enumeration

3. Weighted Model Integration via AllSMT

4. A simple framework for Canonical DDs Modulo Theories via AllSMT Enumeration

CNF encodings: what is good for SAT, but not always for enumeration

Standard practice

- Given a formula $\varphi(\mathcal{A})$, we encode it in CNF:
  - Tseitin, or
  - Plaisted&Greenbaum (P&G).
- They preserve satisfiability: $\varphi(\mathcal{A}) \equiv \exists \mathcal{B}. \varphi_{CNF}(\mathcal{A} \cup \mathcal{B})$.

Issue for **disjoint / projected enumeration**

- We care about **partial models** over $\mathcal{A}$.
- Tseitin and vanilla P&G can **force** subformulas to be true/false, refining one compact partial model into many larger ones.
- Result: the solver enumerates **many more disjoint models** than necessary for the original formula.

Let $\varphi \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{\Big(\overbrace{\Big(\overbrace{(A_3 \vee A_4)}^{B_2} \wedge \overbrace{(A_5 \vee A_6)}^{B_3} \Big)}^{B_4} \leftrightarrow A_7 \Big)}^{B_5}$

$$\text{Let } \varphi \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\,\overbrace{(\,\overbrace{(A_3 \vee A_4)}^{B_2} \wedge \overbrace{(A_5 \vee A_6)}^{B_3}\,)}^{B_4} \leftrightarrow A_7)}^{B_5}$$

$$\begin{aligned}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) && \wedge && // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) && \wedge && // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) && \wedge && // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) && \wedge && // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) && \wedge && // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) && \wedge && \\
& (B_1 \vee B_5) && &&
\end{aligned}$$

Let $\varphi \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2} \wedge \overbrace{(A_5 \vee A_6)}^{B_3})}^{B_4} \leftrightarrow A_7)}^{B_5}$

$$
\begin{array}{rlll}
\mathsf{CNF_{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) & \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) & \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) & \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) & \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) & \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) & \wedge & \\
& (B_1 \vee B_5) & &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

Let $\varphi \stackrel{\text{def}}{=} \overset{B_1}{\boxed{(A_1 \wedge A_2)}} \vee \overset{B_5}{\boxed{\Big(\overset{B_4}{\boxed{\Big(\overset{B_2}{\boxed{(A_3 \vee A_4)}} \wedge \overset{B_3}{\boxed{(A_5 \vee A_6)}} \Big)}} \leftrightarrow A_7 \Big)}}$

$$
\begin{array}{rll}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) \quad \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) \quad \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) \quad \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) \quad \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) \quad \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) \quad \wedge & \\
& (B_1 \vee B_5) &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

$\mu^{\boldsymbol{A}} \not\models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{Ts}}(\varphi)!$ i.e., $\nexists \eta^{\boldsymbol{B}}$ s.t. $(\mu^{\boldsymbol{A}} \cup \eta^{\boldsymbol{B}} \models \mathsf{CNF}_{\mathsf{Ts}}(\varphi))$

Let $\varphi \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2} \wedge \overbrace{(A_5 \vee A_6)}^{B_3})}^{B_4} \leftrightarrow A_7)}^{B_5}$

$$
\begin{array}{lll}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) \quad \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) \quad \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) \quad \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) \quad \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) \quad \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) \quad \wedge & \\
& (B_1 \vee B_5) &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

$$\mu^{\boldsymbol{A}} \not\models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{Ts}}(\varphi)! \qquad \text{i.e., } \nexists \eta^{\boldsymbol{B}} \text{ s.t. } (\mu^{\boldsymbol{A}} \cup \eta^{\boldsymbol{B}} \models \mathsf{CNF}_{\mathsf{Ts}}(\varphi))$$

E.g., $\eta^{\boldsymbol{B}} \stackrel{\text{def}}{=} \{\neg B_2, \ldots\}$

Let $\varphi \stackrel{\text{def}}{=} \underbrace{\boxed{(A_1 \wedge A_2)}}_{B_1} \vee \underbrace{\boxed{(\underbrace{\boxed{(\underbrace{\boxed{(A_3 \vee A_4)}}_{B_2} \wedge \underbrace{\boxed{(A_5 \vee A_6)}}_{B_3})}}_{B_4} \leftrightarrow A_7)}}_{B_5}$

$$
\begin{array}{llr}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) \quad \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) \quad \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) \quad \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) \quad \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) \quad \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) \quad \wedge & \\
& (B_1 \vee B_5) &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

$\mu^{\boldsymbol{A}} \not\models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{Ts}}(\varphi)$! i.e., $\nexists \eta^{\boldsymbol{B}}$ s.t. $(\mu^{\boldsymbol{A}} \cup \eta^{\boldsymbol{B}} \models \mathsf{CNF}_{\mathsf{Ts}}(\varphi))$

E.g., $\eta^{\boldsymbol{B}} \stackrel{\text{def}}{=} \{\neg B_2, \neg B_4, \ldots\}$

Let $\varphi \stackrel{\text{def}}{=} \underbrace{(A_1 \wedge A_2)}_{B_1} \vee \underbrace{(\underbrace{(\underbrace{(A_3 \vee A_4)}_{B_2} \wedge \underbrace{(A_5 \vee A_6)}_{B_3})}_{B_4} \leftrightarrow A_7)}_{B_5}$

$$
\begin{array}{lll}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) \quad \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) \quad \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) \quad \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) \quad \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) \quad \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) \quad \wedge & \\
& (B_1 \vee B_5) &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

$$\mu^{\boldsymbol{A}} \not\models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{Ts}}(\varphi)! \qquad \text{i.e., } \nexists \eta^{\boldsymbol{B}} \text{ s.t. } (\mu^{\boldsymbol{A}} \cup \eta^{\boldsymbol{B}} \models \mathsf{CNF}_{\mathsf{Ts}}(\varphi))$$

E.g., $\eta^{\boldsymbol{B}} \stackrel{\text{def}}{=} \{\neg B_2, \neg B_4, B_5, \ldots\}$

Let $\varphi \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2} \wedge \overbrace{(A_5 \vee A_6)}^{B_3})}^{B_4} \leftrightarrow A_7)}^{B_5}$

$$
\begin{array}{llr}
\mathsf{CNF}_{\mathsf{Ts}}(\varphi) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge (B_1 \vee \neg A_1 \vee \neg A_2) \quad \wedge & // B_1 \leftrightarrow (A_1 \wedge A_2) \\
& (B_2 \vee \neg A_3) \wedge (B_2 \vee \neg A_4) \wedge (\neg B_2 \vee A_3 \vee A_4) \quad \wedge & // B_2 \leftrightarrow (A_3 \vee A_4) \\
& (B_3 \vee \neg A_5) \wedge (B_3 \vee \neg A_6) \wedge (\neg B_3 \vee A_5 \vee A_6) \quad \wedge & // B_3 \leftrightarrow (A_5 \vee A_6) \\
& (\neg B_4 \vee B_2) \wedge (\neg B_4 \vee B_3) \wedge (B_4 \vee \neg B_2 \vee \neg B_3) \quad \wedge & // B_4 \leftrightarrow (B_2 \wedge B_3) \\
& (\neg B_5 \vee B_4 \vee \neg A_7) \wedge (\neg B_5 \vee \neg B_4 \vee A_7) \quad \wedge & // B_5 \leftrightarrow (B_4 \leftrightarrow A_7) \\
& (B_5 \vee B_4 \vee A_7) \wedge (B_5 \vee \neg B_4 \vee \neg A_7) \quad \wedge & \\
& (B_1 \vee B_5) &
\end{array}
$$

Problem

Consider $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$ s.t. $\mu^{\boldsymbol{A}} \models \varphi$.

$$\mu^{\boldsymbol{A}} \not\models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{Ts}}(\varphi)! \qquad \text{i.e., } \nexists \eta^{\boldsymbol{B}} \text{ s.t. } (\mu^{\boldsymbol{A}} \cup \eta^{\boldsymbol{B}} \models \mathsf{CNF}_{\mathsf{Ts}}(\varphi))$$

E.g., $\eta^{\boldsymbol{B}} \stackrel{\text{def}}{=} \{\neg B_2, \neg B_4, B_5, \neg B_1, \neg B_3\} \Longrightarrow (\neg A_1 \vee \neg A_2)$ and $(\neg A_5 \wedge \neg A_6)$ are not satisfied

An enumeration-friendly CNF: NNF + P&G [6,7]

Key idea

- First build an **NNF DAG** of φ (linear size).
- Then apply a **polarity-aware** P&G encoding on the DAG.

- Subformulas with both polarities are split into two NNF nodes: one for the positive, one for the negative occurrence.
- Each node gets its own label, and labels can be set to **false** without forcing the subformula to be evaluated.
- Intuitively: the CNF allows **"don't care"** on subformulas, which is exactly what we want for partial models.

Consequence

Every partial model of the original formula can be **extended** to a model of the NNF+P&G CNF, including really short ones.

G. Masina, G. Spallitta, and R. Sebastiani (2023a). *On CNF Conversion for Disjoint SAT Enumeration*. In: *26th International Conference on Theory and Applications of Satisfiability Testing*

G. Masina, G. Spallitta, and R. Sebastiani (2023b). *On CNF Conversion for SAT and SMT Enumeration*. In: *JAIR - Under submission*

Let $\varphi \stackrel{\text{def}}{=} (A_1 \wedge A_2) \vee (((A_3 \vee A_4) \wedge (A_5 \vee A_6)) \leftrightarrow A_7)$

Let $\varphi \stackrel{\text{def}}{=} \boxed{(A_1 \wedge A_2)} \vee \boxed{(\,(\,(A_3 \vee A_4) \wedge (A_5 \vee A_6)\,) \leftrightarrow A_7)}$

$$\mathsf{NNF}(\varphi) \stackrel{\text{def}}{=} \overset{B_1}{(A_1 \wedge A_2)} \vee \overset{B_7}{\Big(\overset{B_5}{\Big(\overset{B_4^-}{\Big(\overset{B_2^-}{(\neg A_3 \wedge \neg A_4)} \vee \overset{B_3^-}{(\neg A_5 \wedge \neg A_6)}\Big)} \vee A_7\Big)} \wedge \overset{B_6}{\Big(\overset{B_4^+}{\Big(\overset{B_2^+}{(A_3 \vee A_4)} \wedge \overset{B_3^+}{(A_5 \vee A_6)}\Big)} \vee \neg A_7\Big)}\Big)}$$

$$\text{Let } \varphi \stackrel{\text{def}}{=} (A_1 \wedge A_2) \vee (((A_3 \vee A_4) \wedge (A_5 \vee A_6)) \leftrightarrow A_7)$$

$$\mathsf{NNF}(\varphi) \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(\overbrace{(\neg A_3 \wedge \neg A_4)}^{B_2^-} \vee \overbrace{(\neg A_5 \wedge \neg A_6)}^{B_3^-})}^{B_4^-} \vee A_7)}^{B_5} \wedge \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2^+} \wedge \overbrace{(A_5 \vee A_6)}^{B_3^+})}^{B_4^+} \vee \neg A_7)}^{B_6})}^{B_7}$$

$$\begin{aligned}
\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi)) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge && // B_1 \rightarrow (A_1 \wedge A_2) \\
& (\neg B_2^- \vee \neg A_3) \wedge (\neg B_2^- \vee \neg A_4) \wedge && // B_2^- \rightarrow (\neg A_3 \wedge \neg A_4) \\
& (\neg B_3^- \vee \neg A_5) \wedge (\neg B_3^- \vee \neg A_6) \wedge && // B_3^- \rightarrow (\neg A_5 \wedge \neg A_6) \\
& (\neg B_4^- \vee B_2^- \vee B_3^-) \wedge && // B_4^- \rightarrow (B_2^- \vee B_3^-) \\
& (\neg B_5 \vee B_4^- \vee A_7) \wedge && // B_5 \rightarrow (B_4^- \vee A_7) \\
& (\neg B_2^+ \vee A_3 \vee A_4) \wedge && // B_2^+ \rightarrow (A_3 \vee A_4) \\
& (\neg B_3^+ \vee A_5 \vee A_6) \wedge && // B_3^+ \rightarrow (A_5 \vee A_6) \\
& (\neg B_4^+ \vee B_2^+) \wedge (\neg B_4^+ \vee B_3^+) \wedge && // B_4^+ \rightarrow (B_2^+ \wedge B_3^+) \\
& (\neg B_6 \vee B_4^+ \vee \neg A_7) \wedge && // B_6 \rightarrow (B_4^+ \vee \neg A_7) \\
& (\neg B_7 \vee B_5) \wedge (\neg B_7 \vee B_6) \wedge && // B_7 \rightarrow (B_5 \wedge B_6) \\
& (B_1 \vee B_7) \wedge && \\
& (\neg B_2^+ \vee \neg B_2^-) \wedge (\neg B_3^+ \vee \neg B_3^-) \wedge (\neg B_4^+ \vee \neg B_4^-) &&
\end{aligned}$$

Let $\varphi \stackrel{\text{def}}{=} (A_1 \wedge A_2) \vee (((A_3 \vee A_4) \wedge (A_5 \vee A_6)) \leftrightarrow A_7)$

$$\mathsf{NNF}(\varphi) \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(\overbrace{(\neg A_3 \wedge \neg A_4)}^{B_2^-} \vee \overbrace{(\neg A_5 \wedge \neg A_6)}^{B_3^-})}^{B_4^-} \vee A_7)}^{B_5} \wedge \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2^+} \wedge \overbrace{(A_5 \vee A_6)}^{B_3^+})}^{B_4^+} \vee \neg A_7)}^{B_6})}^{B_7}$$

$$\begin{array}{lll}
\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi)) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge & // B_1 \rightarrow (A_1 \wedge A_2) \\
& (\neg B_2^- \vee \neg A_3) \wedge (\neg B_2^- \vee \neg A_4) \wedge & // B_2^- \rightarrow (\neg A_3 \wedge \neg A_4) \\
& (\neg B_3^- \vee \neg A_5) \wedge (\neg B_3^- \vee \neg A_6) \wedge & // B_3^- \rightarrow (\neg A_5 \wedge \neg A_6) \\
& (\neg B_4^- \vee B_2^- \vee B_3^-) \wedge & // B_4^- \rightarrow (B_2^- \vee B_3^-) \\
& (\neg B_5 \vee B_4^- \vee A_7) \wedge & // B_5 \rightarrow (B_4^- \vee A_7) \\
& (\neg B_2^+ \vee A_3 \vee A_4) \wedge & // B_2^+ \rightarrow (A_3 \vee A_4) \\
& (\neg B_3^+ \vee A_5 \vee A_6) \wedge & // B_3^+ \rightarrow (A_5 \vee A_6) \\
& (\neg B_4^+ \vee B_2^+) \wedge (\neg B_4^+ \vee B_3^+) \wedge & // B_4^+ \rightarrow (B_2^+ \wedge B_3^+) \\
& (\neg B_6 \vee B_4^+ \vee \neg A_7) \wedge & // B_6 \rightarrow (B_4^+ \vee \neg A_7) \\
& (\neg B_7 \vee B_5) \wedge (\neg B_7 \vee B_6) \wedge & // B_7 \rightarrow (B_5 \wedge B_6) \\
& (B_1 \vee B_7) \wedge & \\
& (\neg B_2^+ \vee \neg B_2^-) \wedge (\neg B_3^+ \vee \neg B_3^-) \wedge (\neg B_4^+ \vee \neg B_4^-) &
\end{array}$$

Let $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$

$\mu^{\boldsymbol{A}} \models \varphi$

$\mu^{\boldsymbol{A}} \models \mathsf{NNF}(\varphi)$

$\mu^{\boldsymbol{A}} \models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi))$

$$\text{Let } \varphi \stackrel{\text{def}}{=} (A_1 \wedge A_2) \vee (((A_3 \vee A_4) \wedge (A_5 \vee A_6)) \leftrightarrow A_7)$$

$$\mathsf{NNF}(\varphi) \stackrel{\text{def}}{=} \overbrace{(A_1 \wedge A_2)}^{B_1} \vee \overbrace{(\overbrace{(\overbrace{(\overbrace{(\neg A_3 \wedge \neg A_4)}^{B_2^-} \vee \overbrace{(\neg A_5 \wedge \neg A_6)}^{B_3^-})}^{B_4^-} \vee A_7)}^{B_5} \wedge \overbrace{(\overbrace{(\overbrace{(A_3 \vee A_4)}^{B_2^+} \wedge \overbrace{(A_5 \vee A_6)}^{B_3^+})}^{B_4^+} \vee \neg A_7)}^{B_6})}^{B_7}$$

$$\begin{array}{lll}
\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi)) \stackrel{\text{def}}{=} & (\neg B_1 \vee A_1) \wedge (\neg B_1 \vee A_2) \wedge & // B_1 \rightarrow (A_1 \wedge A_2) \\
& (\neg B_2^- \vee \neg A_3) \wedge (\neg B_2^- \vee \neg A_4) \wedge & // B_2^- \rightarrow (\neg A_3 \wedge \neg A_4) \\
& (\neg B_3^- \vee \neg A_5) \wedge (\neg B_3^- \vee \neg A_6) \wedge & // B_3^- \rightarrow (\neg A_5 \wedge \neg A_6) \\
& (\neg B_4^- \vee B_2^- \vee B_3^-) \wedge & // B_4^- \rightarrow (B_2^- \vee B_3^-) \\
& (\neg B_5 \vee B_4^- \vee A_7) \wedge & // B_5 \rightarrow (B_4^- \vee A_7) \\
& (\neg B_2^+ \vee A_3 \vee A_4) \wedge & // B_2^+ \rightarrow (A_3 \vee A_4) \\
& (\neg B_3^+ \vee A_5 \vee A_6) \wedge & // B_3^+ \rightarrow (A_5 \vee A_6) \\
& (\neg B_4^+ \vee B_2^+) \wedge (\neg B_4^+ \vee B_3^+) \wedge & // B_4^+ \rightarrow (B_2^+ \wedge B_3^+) \\
& (\neg B_6 \vee B_4^+ \vee \neg A_7) \wedge & // B_6 \rightarrow (B_4^+ \vee \neg A_7) \\
& (\neg B_7 \vee B_5) \wedge (\neg B_7 \vee B_6) \wedge & // B_7 \rightarrow (B_5 \wedge B_6) \\
& (B_1 \vee B_7) \wedge & \\
& (\neg B_2^+ \vee \neg B_2^-) \wedge (\neg B_3^+ \vee \neg B_3^-) \wedge (\neg B_4^+ \vee \neg B_4^-) &
\end{array}$$

Let $\mu^{\boldsymbol{A}} \stackrel{\text{def}}{=} \{\neg A_3, \neg A_4, \neg A_7\}$

$\mu^{\boldsymbol{A}} \models \varphi$

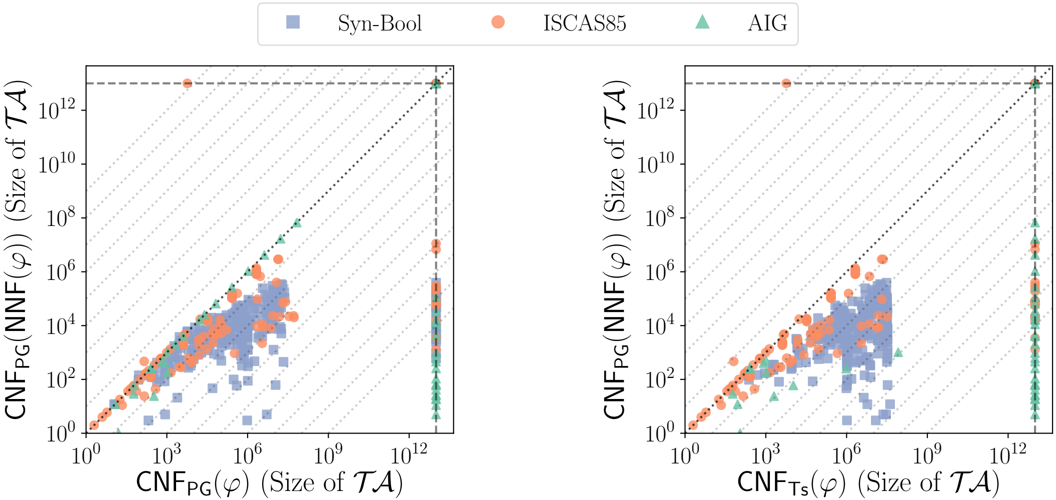
$\mu^{\boldsymbol{A}} \models \mathsf{NNF}(\varphi)$

$\mu^{\boldsymbol{A}} \models \exists \boldsymbol{B}.\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi))$

E.g.

$$\eta^{\boldsymbol{B}} \stackrel{\text{def}}{=} \{\neg B_2^+, B_5, B_6, B_2^-, B_4^-, \neg B_1, \neg B_3^-, \neg B_3^+, B_7, \neg B_4^+\}$$

Results



Syn-Bool
ISCAS85
AIG
10^0
10^2
10^4
10^6
10^8
10^{10}
10^{12}
10^0
10^3
10^6
10^9
10^{12}
$\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi))$ (Size of $\mathcal{TA}$)
$\mathsf{CNF}_{\mathsf{PG}}(\varphi)$ (Size of $\mathcal{TA}$)
$\mathsf{CNF}_{\mathsf{PG}}(\mathsf{NNF}(\varphi))$ (Size of $\mathcal{TA}$)
$\mathsf{CNF}_{\mathsf{Ts}}(\varphi)$ (Size of $\mathcal{TA}$)

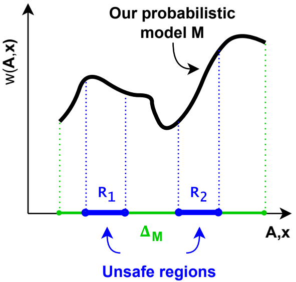
Outline

WMI: hybrid probabilistic inference

Weighted Model Integration (WMI)

- Continuous variables $\mathbf{x} \in \mathbb{R}^N$ and Booleans $\mathbf{A} \in \{0,1\}^M$.
- SMT($\mathcal{LRA}$) constraint $\varphi(\mathbf{x}, \mathbf{A})$ and non-negative weight $w(\mathbf{x}, \mathbf{A})$.
- **WMI** integrates w over all models of φ:

$$\mathrm{WMI}(\varphi, w) = \sum_{\mu_{\mathbf{A}}} \int_{\mathbf{x} \models \varphi(\mathbf{x}, \mu_{\mathbf{A}})} w(\mathbf{x}, \mu_{\mathbf{A}})\, d\mathbf{x}.$$



Goal: hybrid probabilistic inference

$$\Pr(R_1 \vee R_2 \mid M) = \frac{\mathrm{WMI}(\Delta_M \wedge (R_1 \vee R_2), w)}{\mathrm{WMI}(\Delta_M, w)}$$

Main challenge

- Each Boolean model induces a **continuous integral**.
- Complex weights (`ite`, piecewise densities) $\Rightarrow$ many regions.
- We need to **enumerate a small set of partial assignments** that capture all relevant integrals.

WMI-PA and its issue

The original WMI-PA procedure integrates $\mathcal{T}$-atoms from the weight function by using fresh Boolean variables and double implications...

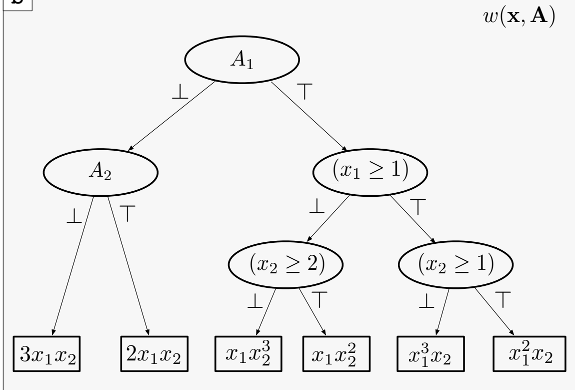
a

$$\varphi \stackrel{def}{=} \top,\ \chi \stackrel{def}{=} [\![x_1 \in [0,2]]\!] \wedge [\![x_2 \in [0,3]]\!]$$

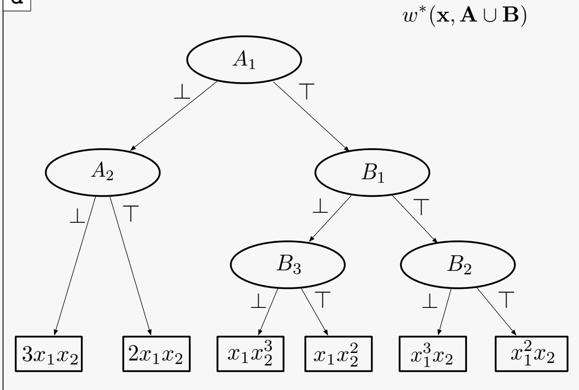
c

$$\varphi^*(\mathbf{x}, \mathbf{A} \cup \mathbf{B}) \stackrel{def}{=} \varphi \wedge \chi \wedge B_1 \leftrightarrow (x_1 \geq 1) \wedge B_2 \leftrightarrow (x_2 \geq 1) \wedge B_3 \leftrightarrow (x_2 \geq 2)$$

b



d



e

Weight-aware AllSMT: skeletons of the weight

Skeleton of a weight function

Given a weight w, build a Boolean **skeleton** sk(w) such that:

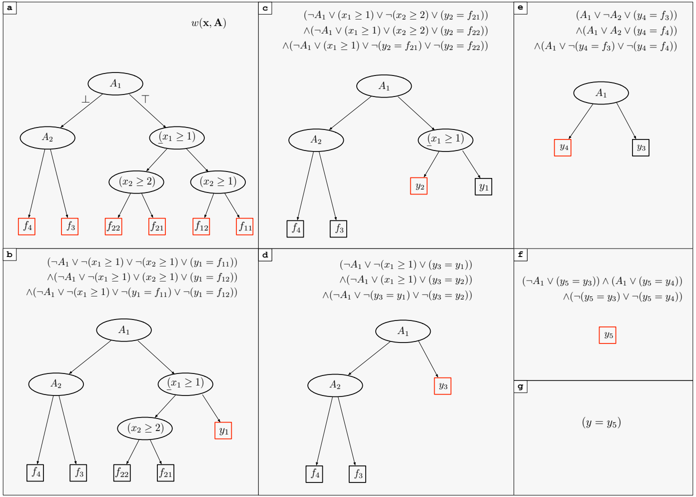
- its atoms are exactly the conditions appearing in w (from `ite`s, etc.);
- φ is equivalent to $\varphi \wedge \mathsf{sk}(w)$;
- any partial assignment to these conditions makes the corresponding $w(\mathbf{x}, \mu)$ **arithmetically computable**.

Effect

- SMT solver is **aware of the weight structure**.
- It enumerates only conditions that **change** the value of the integral.
- $\Rightarrow$ fewer regions, fewer integrals, faster WMI.

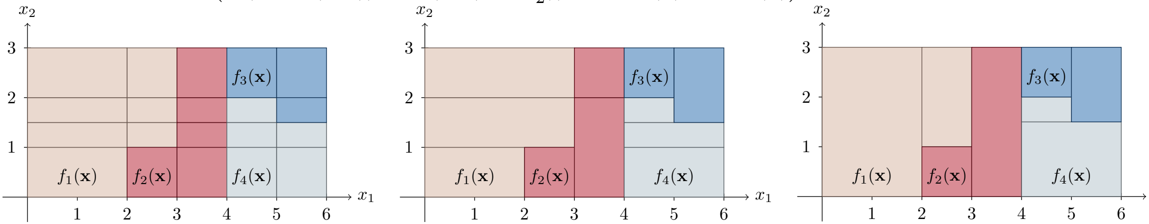
Example: implicit skeleton encoding

Bottom-up procedure rewriting the **skeleton** of w into a $\mathcal{LRA} \cup \mathcal{EUF}$ formula, $[\![y = w]\!]_{\mathcal{EUF}}$.



Impact of skeletons on WMI [8]

$$\varphi = \top$$
$$\mathrm{supp}(w) = (x_1 \geq 0) \wedge (x_1 \leq 6) \wedge (x_2 \geq 0) \wedge (x_2 \leq 3)$$
$$w = \text{If } (x_1 \leq 4) \text{ Then } \big(\text{If } (x_1 \leq 2) \vee ((x_1 \leq 3) \wedge (x_2 > 1)) \text{ Then } f_1(\mathbf{x}) \text{ Else } f_2(\mathbf{x})\big) \text{ Else}$$
$$\big(\text{If } (x_2 > 2) \vee ((x_1 > 5) \wedge (x_2 > \tfrac{3}{2})) \text{ Then } f_3(\mathbf{x}) \text{ Else } f_4(\mathbf{x})\big)$$



WMI-PA: **20** integrals — SA-WMI-PA: **11** integrals — Ideal case: **8** integrals (minimum)

Takeaway

- Skeleton-guided AllSMT drastically reduces the number of integrals.
- Still not optimal as the last configuration, but definitely an improvement... but can we do better?

[8] G. Spallitta, G. Masina, P. Morettin, A. Passerini, and R. Sebastiani (2022). *SMT-based Weighted Model Integration with Structure Awareness*. In: *UAI*

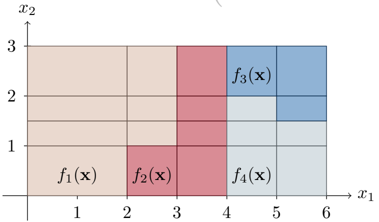
Impact of custom CNF-ization on WMI

$$\varphi = \top$$

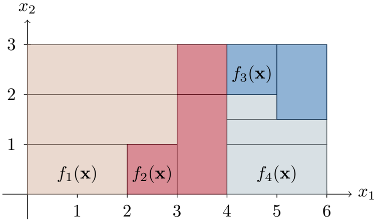
$$\mathrm{supp}(w) = (x_1 \geq 0) \wedge (x_1 \leq 6) \wedge (x_2 \geq 0) \wedge (x_2 \leq 3)$$

$$w = \textsf{If } (x_1 \leq 4) \textsf{ Then } \Big(\textsf{If } (x_1 \leq 2) \vee ((x_1 \leq 3) \wedge (x_2 > 1)) \textsf{ Then } f_1(\mathbf{x}) \textsf{ Else } f_2(\mathbf{x})\Big) \textsf{ Else}$$

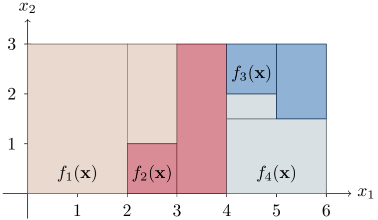
$$\Big(\textsf{If } (x_2 > 2) \vee ((x_1 > 5) \wedge (x_2 > \tfrac{3}{2})) \textsf{ Then } f_3(\mathbf{x}) \textsf{ Else } f_4(\mathbf{x})\Big)$$



WMI-PA: **20** integrals



SA-WMI-PA: **11** integrals



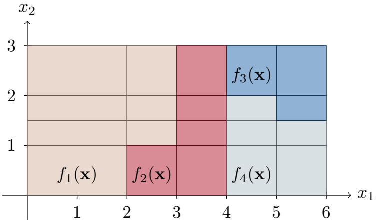
Ideal case (?): **8** integrals (minimum)

Weight function structure

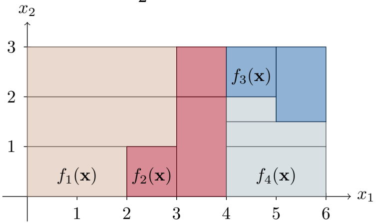
- What if the weight function treats non-atomic conditions?
- Any solver would just apply Tseitin to obtain a CNF form... but we know Tseitin is no good :(

Impact of custom CNF-ization on WMI [9]

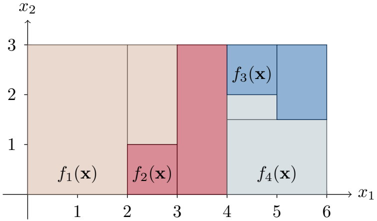
$$\varphi = \top$$
$$\mathrm{supp}(w) = (x_1 \geq 0) \wedge (x_1 \leq 6) \wedge (x_2 \geq 0) \wedge (x_2 \leq 3)$$
$$w = \text{If } (x_1 \leq 4) \text{ Then } \big(\text{If } (x_1 \leq 2) \vee ((x_1 \leq 3) \wedge (x_2 > 1)) \text{ Then } f_1(\mathbf{x}) \text{ Else } f_2(\mathbf{x})\big) \text{ Else}$$
$$\big(\text{If } (x_2 > 2) \vee ((x_1 > 5) \wedge (x_2 > \tfrac{3}{2})) \text{ Then } f_3(\mathbf{x}) \text{ Else } f_4(\mathbf{x})\big)$$



WMI-PA: **20** integrals



SA-WMI-PA: **11** integrals



SAE4WMI: **8** integrals (minimum)

Takeaway

- SAE4WMI is **faster** and more **compact** than previous WMI algorithms on synthetic and DET benchmarks thanks to this custom transformation, and reaches the optimal subdivision!

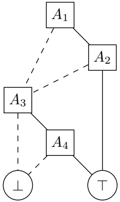
[9] G. Spallitta, G. Masina, P. Morettin, A. Passerini, and R. Sebastiani (2024). *Enhancing SMT-based Weighted Model Integration by Structure Awareness.* In: *Artif Intell*

Outline

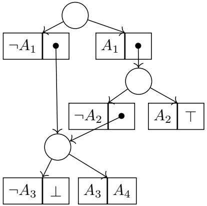
From Boolean DDs to DDs modulo theories

Boolean Decision Diagrams

- OBDDs and SDDs are **canonical** representations of Boolean functions.
- Widely used in formal verification and knowledge compilation.



OBDD for a simple Boolean function



SDD for the same function

Our goal

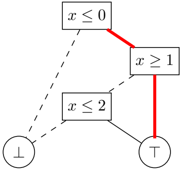
- Lift these ideas to SMT($\mathcal{T}$): **T-OBDDs / T-SDDs**.

Problems: T-inconsistent paths and T-canonicity

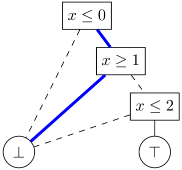
Naive lifting via Boolean abstraction

- Build a DD for $\mathcal{T}2\mathcal{B}(\varphi)$, then refine atoms back to theory atoms.
- This can produce:
    - **T-inconsistent paths** (branches infeasible in the theory).
    - Different diagrams for **T-equivalent** formulas.

$$\varphi = (x \leq 0) \wedge ((x \geq 1) \vee (x \leq 2))$$



Before: DD with T-inconsistent path



After: T-lemma removes infeasible branch

Key idea

Use **T-lemmas** (T-valid clauses) to rule out all T-inconsistent Boolean assignments *before* Boolean DD compilation.

A simple framework for T-DDs

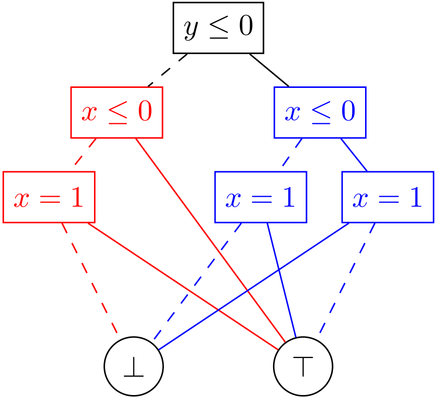
Framework

Given an SMT formula φ:

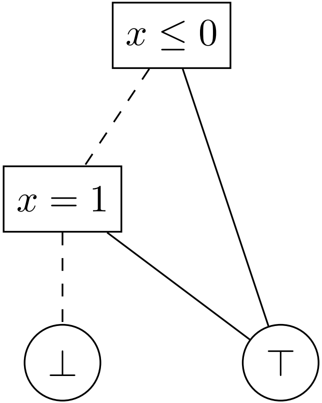
1. Run **AllSMT** to:
   - enumerate all T-consistent assignments to the atoms,
   - collect a set $\mathcal{C}$ of **T-lemmas** ruling out all T-inconsistent assignments.
2. Compile a Boolean DD (OBDD or SDD) for
$$\mathcal{T}2\mathcal{B}(\varphi \wedge \bigwedge_{C \in \mathcal{C}} C).$$
3. Refine this DD back to theory atoms to obtain a **T-DD**.

Effect of T-lemmas: an example

$$(\neg(y \leq 0) \vee ((x \leq 0) \vee (x = 1))) \wedge ((y \leq 0) \vee ((x \leq 0) \oplus (x = 1)))$$



Before: DD contains T-inconsistent regions



After: T-lemmas remove inconsistencies

What to remember

- **Enumeration $\neq$ listing models:** the target is a *small set* of *informative* partial assignments (cubes), not millions of total assignments.
- **Representation matters:** SAT-equivalent encodings can drastically change the short cubes retrieved; we need *enumeration-friendly* encodings.
- **Solver policy matters:** backtracking and shrinking/minimization policies control cube compactness, redundancy, and overall cost.
- **Structure-awareness pays off:** weights, theories, and learned lemmas let us enumerate *only what changes the answer* (e.g., for WMI and canonical T-based DDs).

Thanks for your attention!