Four Flavors of Entailment

Sibylle Möhle[1], Roberto Sebastiani[2], and Armin Biere[1]

[1]Institute for Formal Models and Verification
LIT Secure and Correct Systems Lab

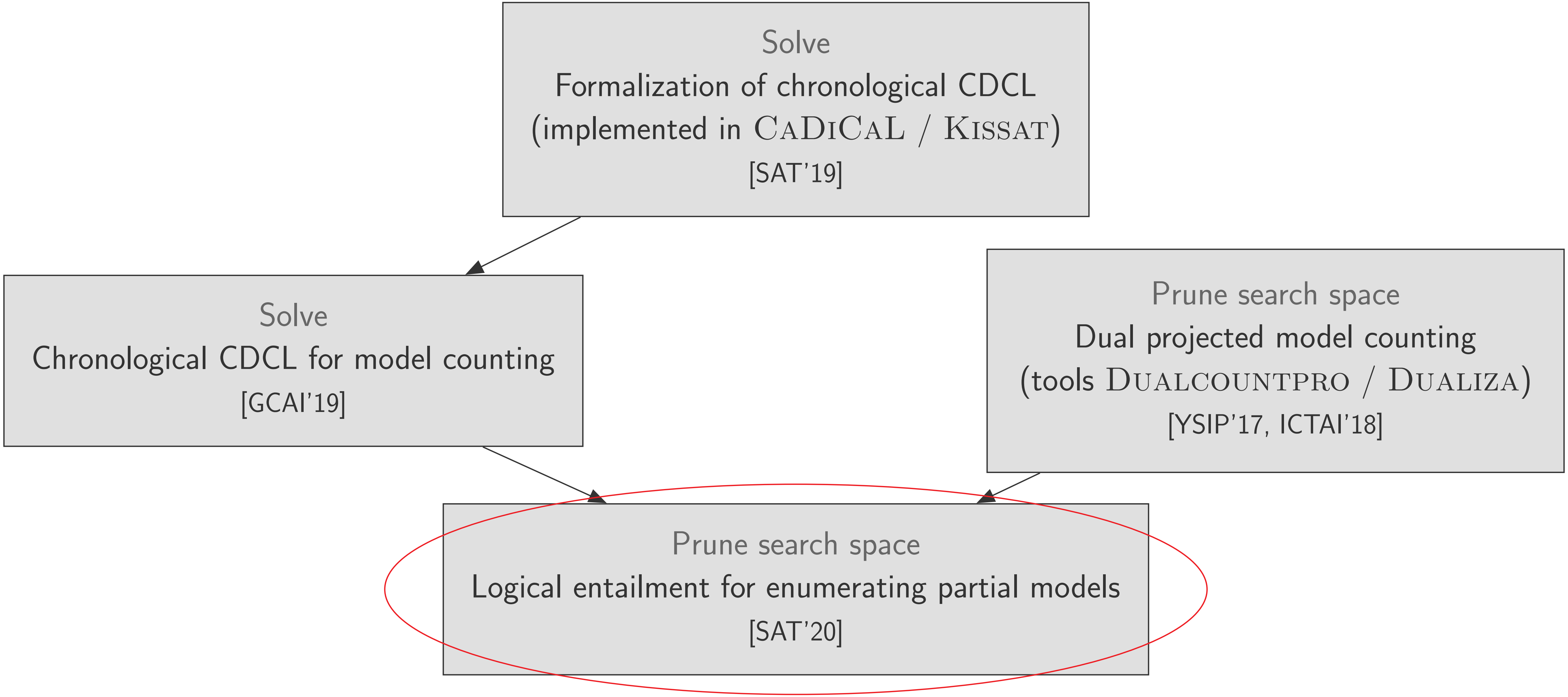
JOHANNES KEPLER UNIVERSITY LINZ

[2]Department of Information Engineering
and Computer Science

UNIVERSITY OF TRENTO

LogiCS Research Seminar

16 March 2021


Solve
Formalization of chronological CDCL
(implemented in CaDiCaL / Kissat)
[SAT'19]
Solve
Chronological CDCL for model counting
[GCAI'19]
Prune search space
Dual projected model counting
(tools DualCountPro / Dualiza)
[YSIP'17, ICTAI'18]
Prune search space
Logical entailment for enumerating partial models
[SAT'20]

Outline

Outline

Motivation

Model Enumeration has Various Applications

- lazy Satisfiability Modulo Theories (Sebastiani, JSAT, 2007)
- predicate abstraction (Lahiri, Nieuwenhuis and Oliveras, CAV'06)
- software product line engineering (Galindo et al., SPLC'16)
- model checking (Biere et al., TACAS'99; McMillan, CAV'02; Strichman, CAV'00)
- preimage computation (Li, Hsiao and Sheng, DATE'04; Sheng and Hsiao, DATE'03)
- weighted model counting (Sang, Beame and Kautz, AAAI'05; Chavira and Darwiche, Artif. Intell., 2008)
- weighted model integration (Morettin, Passerini and Sebastiani, IJCAI'17 and Artif. Intell., 2019)

Motivation

Model Enumeration has Various Applications

- lazy Satisfiability Modulo Theories (Sebastiani, JSAT, 2007)
- predicate abstraction (Lahiri, Nieuwenhuis and Oliveras, CAV'06)
- software product line engineering (Galindo et al., SPLC'16)
- model checking (Biere et al., TACAS'99; McMillan, CAV'02; Strichman, CAV'00)
- preimage computation (Li, Hsiao and Sheng, DATE'04; Sheng and Hsiao, DATE'03)
- weighted model counting (Sang, Beame and Kautz, AAAI'05; Chavira and Darwiche, Artif. Intell., 2008)
- weighted model integration (Morettin, Passerini and Sebastiani, IJCAI'17 and Artif. Intell., 2019)

no repetitions, please!

Design Choices

We need...

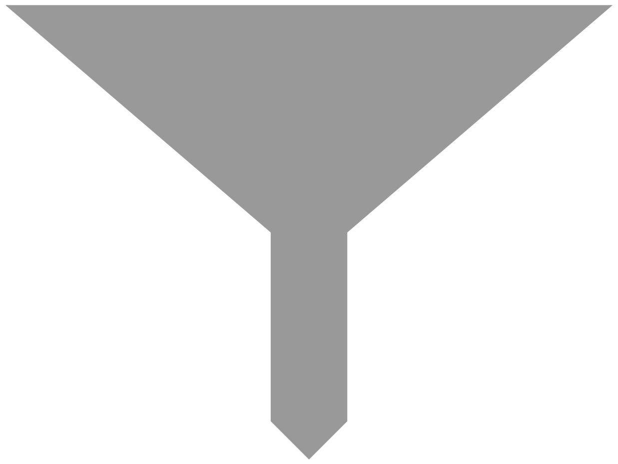
...short (partial) models

- model shrinking
  (Tibebu and Fey, DDECS'18)
- dual reasoning
  (M and Biere, ICTAI'18)
- logical entailment
  (Sebastiani, arXiv.org, 2020)

Example $F = (x \land y) \lor (x \land \neg y)$

$$F|_x = y \lor \neg y \neq 1$$

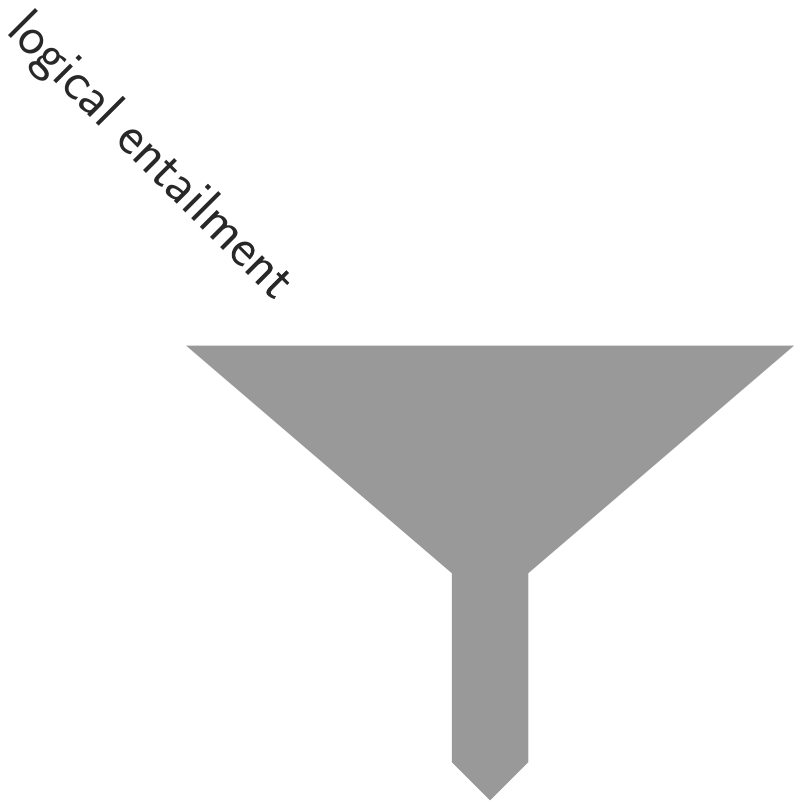
$$F|_{xy} = F|_{x\neg y} = 1 \implies x \models F$$



Design Choices

We need...

...short (partial) models



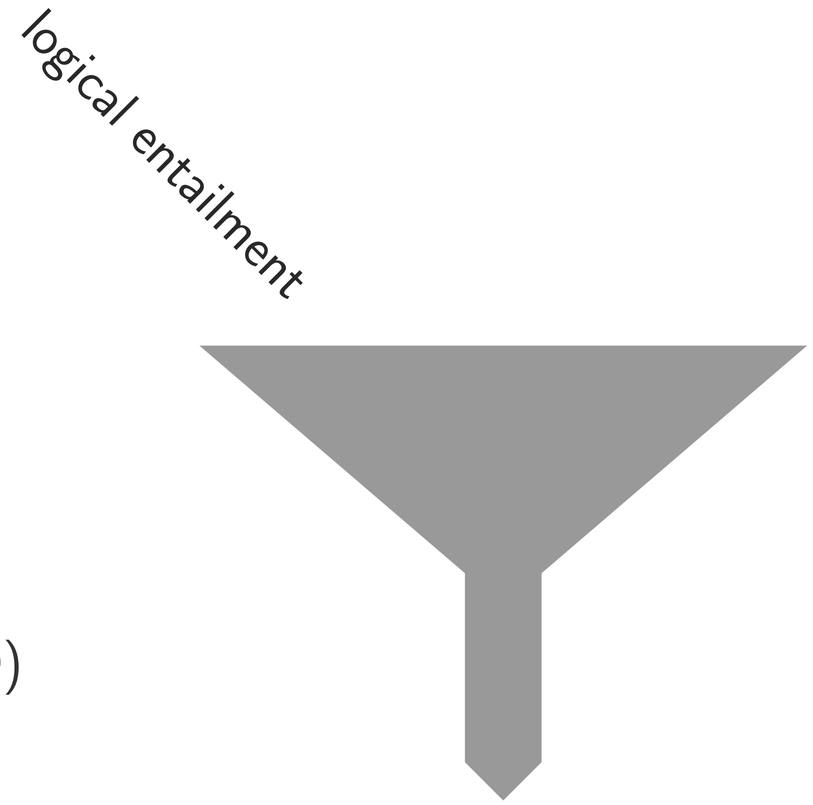
Design Choices

We need. . .

. . . short (partial) models

. . . pairwise disjoint models

- add the negated models as blocking clauses
- variant of conflict analysis
  (Toda and Soh, ACM J. Exp. Algorithmics, 2016)
- chronological CDCL
  (Nadel and Ryvchin, SAT'18; M and Biere, SAT'19)

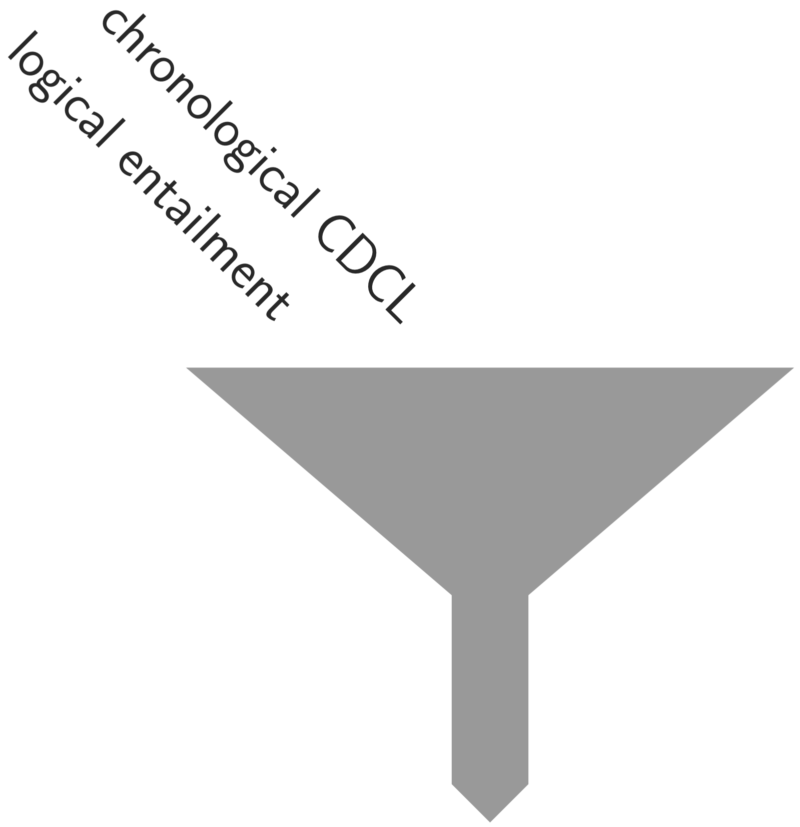


Design Choices

We need. . .

. . . short (partial) models

. . . pairwise disjoint models



Design Choices

We need. . .

. . . short (partial) models

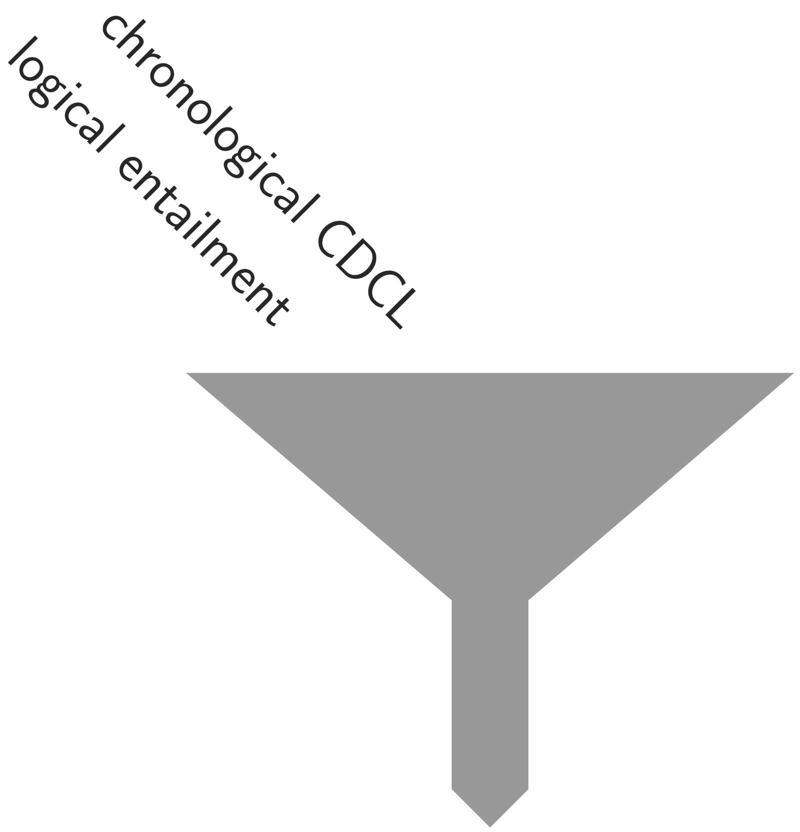
. . . pairwise disjoint models

. . . projection

$F(X, Y)$ where $X \cap Y = \emptyset$

X relevant variables
Y irrelevant variables

$\exists Y\, [\, F(X, Y)\,]$ project $F(X, Y)$ onto X



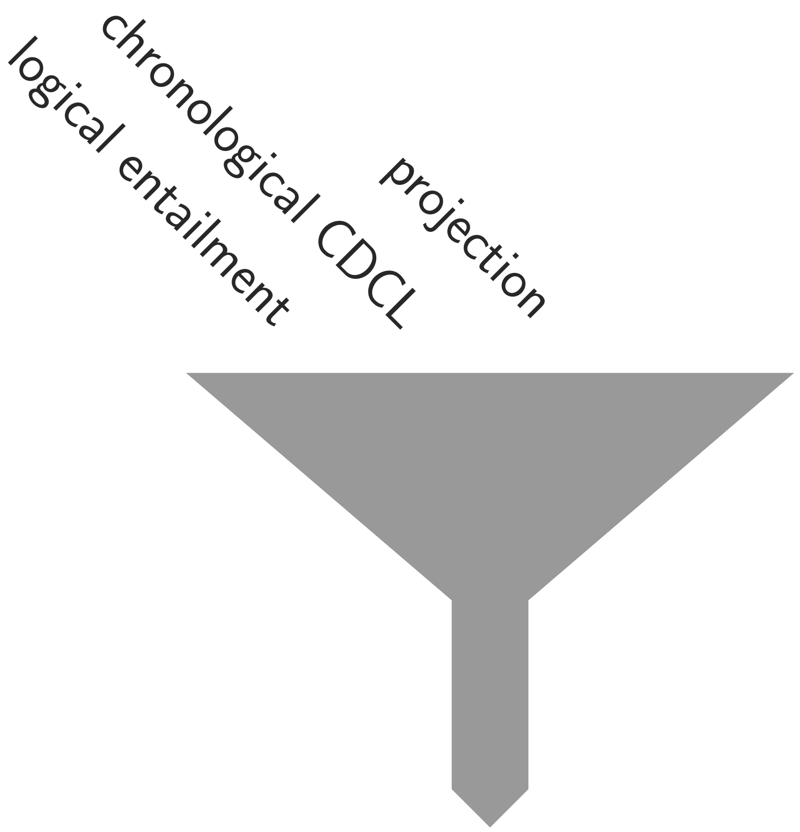
Design Choices

We need...

...short (partial) models

...pairwise disjoint models

...projection



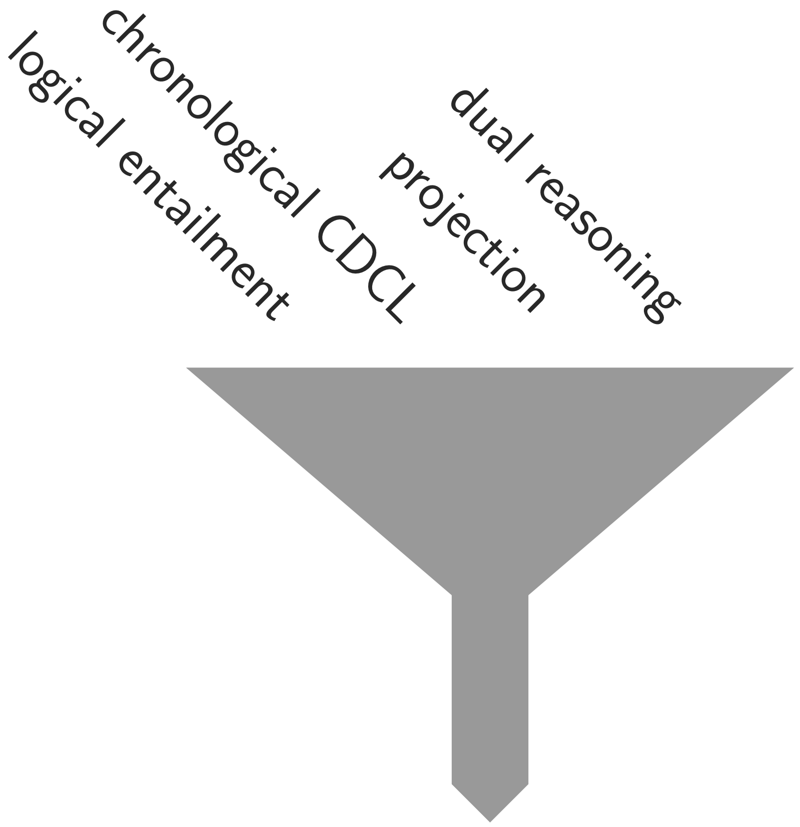
Design Choices

We need. . .

. . . short (partial) models

. . . pairwise disjoint models

. . . projection



Design Choices

We need. . .

. . . short (partial) models

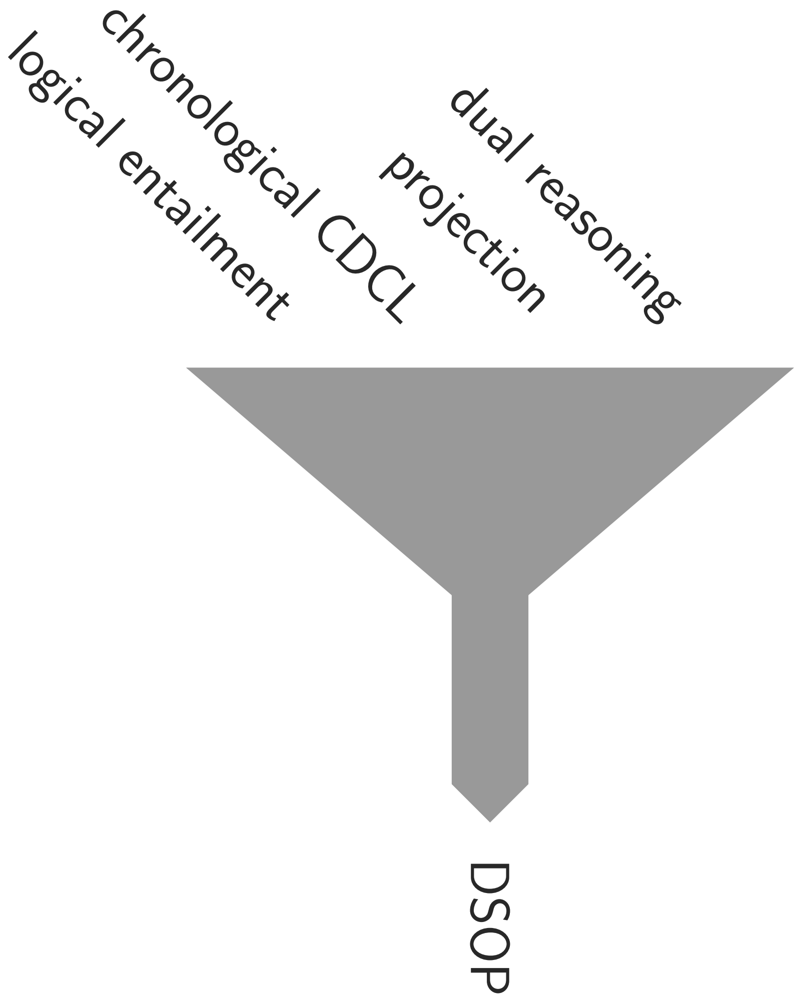
. . . pairwise disjoint models

. . . projection

We get. . .

. . . Disjoint Sum-of-Products (DSOP)

well known in circuit design



Outline

Logical Entailment Condition under Projection

Given: $F(X, Y)$ formula over a set of relevant variables X and a set of irrelevant variables Y

I trail over variables in $X \cup Y$

Entailment under projection onto X: $\forall X \exists Y \, [F|_I]$

Logical Entailment Condition under Projection

Given: $F(X,Y)$ formula over a set of relevant variables X and a set of irrelevant variables Y

I trail over variables in $X \cup Y$

Entailment under projection onto X: $\forall X \exists Y \, [F|_I]$

the unassigned variables are quantified

Logical Entailment Condition under Projection

Given: $F(X,Y)$ formula over a set of relevant variables X and a set of irrelevant variables Y

I trail over variables in $X \cup Y$

Entailment under projection onto X: $\forall X \exists Y \, [F|_I]$

Example: $F(X,Y) = x_1(x_2 \leftrightarrow y_2)$ $\quad X = \{x_1, x_2\}$ $\quad Y = \{y_2\}$

$$F|_{x_1} = (x_2 \leftrightarrow y_2)$$

$$F|_{x_1 x_2} = (1 \leftrightarrow y_2) \quad \text{and} \quad F|_{x_1 x_2 y_2} = 1$$

$$F|_{x_1 \overline{x_2}} = (0 \leftrightarrow y_2) \quad \text{and} \quad F|_{x_1 \overline{x_2}\,\overline{y_2}} = 1$$

$$\Longrightarrow \quad x_1 \models F$$

Logical Entailment Condition under Projection

Given: $F(X, Y)$ formula over a set of relevant variables X and a set of irrelevant variables Y

I trail over variables in $X \cup Y$

Entailment under projection onto X: $\forall X \exists Y \, [F|_I]$

Example: $F(X, Y) = x_1(x_2 \leftrightarrow y_2)$ $X = \{x_1, x_2\}$ $Y = \{y_2\}$

$F|_{x_1} = (x_2 \leftrightarrow y_2)$

Does for each J_X exist *one* J_Y such that $F|_{I'} = 1$ where $I' = I \cup J_X \cup J_Y$?

$F|_{x_1 x_2} = (1 \leftrightarrow y_2)$ and $F|_{x_1 x_2 y_2} = 1$

$F|_{x_1 \overline{x_2}} = (0 \leftrightarrow y_2)$ and $F|_{x_1 \overline{x_2}\, \overline{y_2}} = 1$

$\implies \quad x_1 \models F$

Logical Entailment Condition under Projection

Given: $F(X, Y)$ formula over a set of relevant variables X and a set of irrelevant variables Y

I trail over variables in $X \cup Y$

Entailment under projection onto X: $\forall X \exists Y \, [F|_I]$

Example: $F(X, Y) = x_1(x_2 \leftrightarrow y_2)$ $X = \{x_1, x_2\}$ $Y = \{y_2\}$

$F|_{x_1} = (x_2 \leftrightarrow y_2)$

$F|_{x_1 x_2} = (1 \leftrightarrow y_2)$ and $F|_{x_1 x_2 y_2} = 1$

$F|_{x_1 \overline{x_2}} = (0 \leftrightarrow y_2)$ and $F|_{x_1 \overline{x_2}\, \overline{y_2}} = 1$

$\implies$ $x_1 \models F$

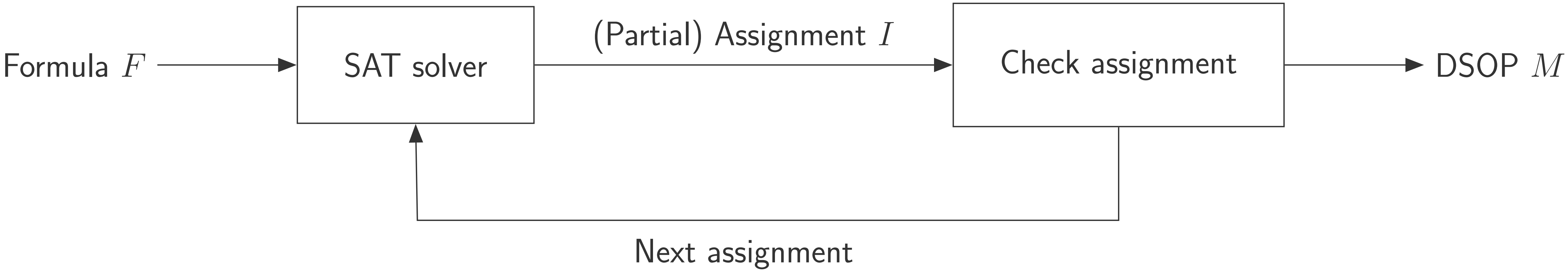
Does for each J_X exist *one* J_Y such that $F|_{I'} = 1$ where $I' = I \cup J_X \cup J_Y$?

$QBF(\varphi) = 1$ where $\varphi = \forall X \exists Y \, [F|_I] = 1$?

But this check is expensive ☹

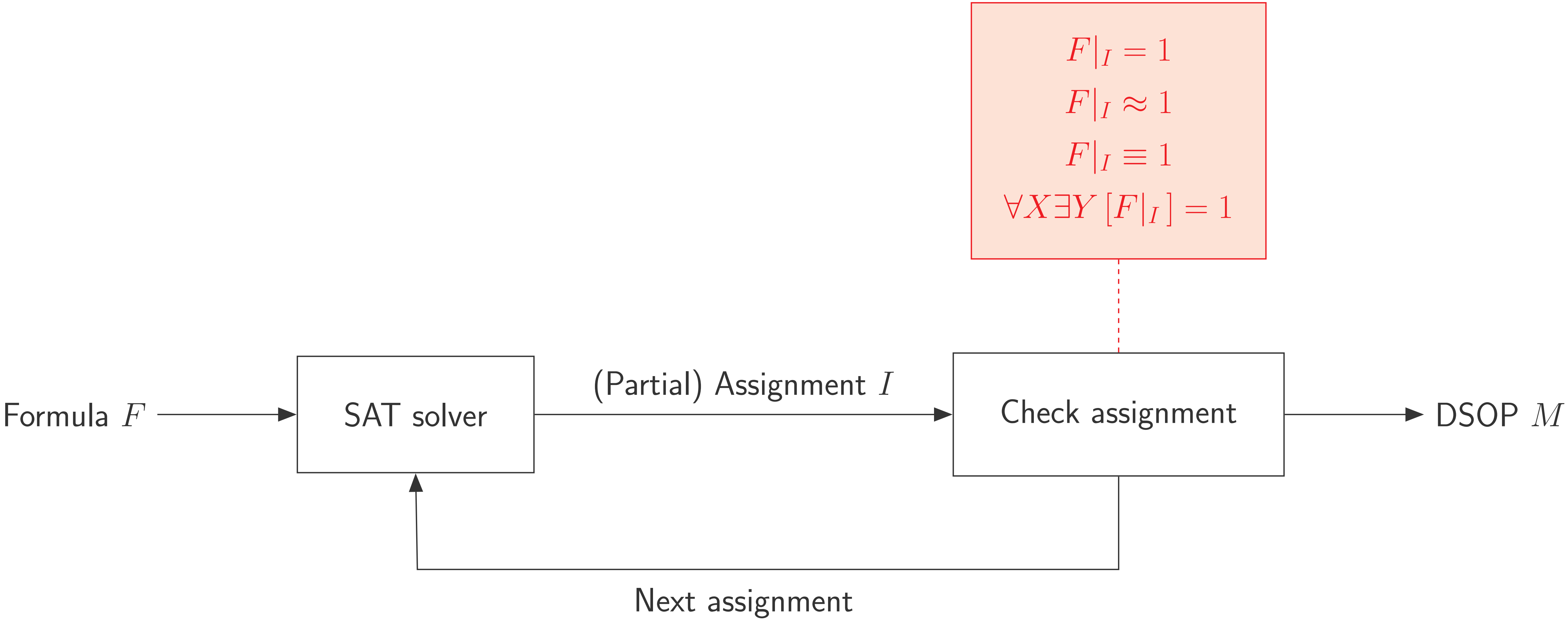
Outline

Main Idea



Formula F
SAT solver
(Partial) Assignment I
Check assignment
DSOP M
Next assignment

Our Contribution



Four Flavors of Logical Entailment under Projection

1) $F|_I = 1$ (*syntactic check*)

$F = (x_1 \vee y \vee x_2) \qquad X = \{x_1, x_2\} \qquad Y = \{y\}$

$I = x_1$: $\qquad F|_I = 1 \quad \Longrightarrow \quad I \models F$

Four Flavors of Logical Entailment under Projection

1) $F|_I = 1$ (*syntactic check*)

2) $F|_I \approx 1$ (*incomplete check in* $\mathbf{P}$)

$F = x_1 y \vee \bar{y} x_2 \qquad X = \{x_1, x_2\} \qquad Y = \{y\}$

$I = x_1 x_2$: $\qquad F|_I = y \vee \bar{y} \neq 1$ but is valid

$I = x_1 x_2 \bar{y}$: $\qquad 0 \in BCP(\neg F, I) \implies x_1 x_2 \models F$

Four Flavors of Logical Entailment under Projection

1) $F|_I = 1$ (*syntactic check*)

2) $F|_I \approx 1$ (*incomplete check in* $\mathbf{P}$)

3) $F|_I \equiv 1$ (*semantic check in* $\mathbf{coNP}$)

$F = x_1(\overline{x_2}\,\overline{y} \vee \overline{x_2}y \vee x_2\overline{y} \vee x_2y)$ $\quad X = \{x_1, x_2\}$ $\quad Y = \{y\}$

$I = x_1$: $\quad I(F) = \overline{x_2}\,\overline{y} \vee \overline{x_2}y \vee x_2\overline{y} \vee x_2y \neq 1$ but is valid

$P =$ CNF(F)
$N =$ CNF$(\neg F)$:

$P|_I$ and $N|_I$ are non-constant and contain no units

$N|_I = (x_2 \vee y)(x_2 \vee \overline{y})(\overline{x_2} \vee y)(\overline{x_2} \vee \overline{y})$: $\quad SAT(N \wedge I) = 0 \quad \Longrightarrow \quad I \models F$

Four Flavors of Logical Entailment under Projection

1) $F|_I = 1$ (*syntactic check*)

2) $F|_I \approx 1$ (*incomplete check in* $\mathbf{P}$)

3) $F|_I \equiv 1$ (*semantic check in* $\mathbf{coNP}$)

4) $\forall X \exists Y \, [F|_I] = 1$ (*check in* $\mathbf{\Pi}_2^P$)

$F = x_1(x_2 \leftrightarrow y_2) \qquad X = \{x_1, x_2\} \qquad Y = \{y_2\}$

$P = \mathsf{CNF}(F)$ and $N = \mathsf{CNF}(\neg F)$:

$P = (x_1)(s_1 \vee s_2)(\overline{s_1} \vee x_2)(\overline{s_1} \vee y_2)(\overline{s_2} \vee \overline{x_2})(\overline{s_2} \vee \overline{y_2})$ where $S = \{s_1, s_2\}$
$N = (\overline{x_1} \vee t_1 \vee t_2)(\overline{t_1} \vee x_2)(\overline{t_1} \vee \overline{y_2})(\overline{t_2} \vee \overline{x_2})(\overline{t_2} \vee y_2)$ where $T = \{t_1, t_2\}$

$I = x_1$: $P|_I$ and $N|_I$ are non-constant and contain no units

$I = x_1 x_2^d \overline{t_2} t_1 \overline{y_2}$: $N|_I = 1$

$\varphi = \forall X \exists Y \, [\, x_2 y_2 \vee \overline{x_2}\, \overline{y_2} \,]$: $QBF(\varphi) = 1 \quad \Longrightarrow \quad x_1 \models F$

Outline

Algorithm

Input: formula $F(X, Y)$ over variables $X \cup Y$ such that $X \cap Y = \emptyset$, trail I, decision level function δ
Output: M DSOP representation of F projected onto X

Count(F)

$I := \varepsilon$; $\delta := \infty$; $M := 0$
forever do
  $C :=$ PropagateUnits(F, I, δ)
  if $C \neq 0$ **then**
    $c := \delta(C)$
    if $c = 0$ **then return** M
    AnalyzeConflict(F, I, C, c)
  else if all variables in $X \cup Y$ are assigned **then**
    if $V(\text{decs}(I)) \cap X = \emptyset$ **then return** $M \vee \pi(I, X)$
    $M := M \vee \pi(I, X)$
    $b := \delta(\text{decs}(\pi(I, X)))$
    Backtrack(I, $b - 1$)

  else Decide(I, δ)

Algorithm

Input: formula $F(X, Y)$ over variables $X \cup Y$ such that $X \cap Y = \emptyset$, trail I, decision level function δ

Output: M DSOP representation of F projected onto X

Count(F)

$I := \varepsilon$; $\delta := \infty$; $M := 0$
forever do
    $C :=$ PropagateUnits(F, I, δ)
    if $C \neq 0$ **then**
        $c := \delta(C)$
        if $c = 0$ **then return** M
        AnalyzeConflict(F, I, C, c)
    else if all variables in $X \cup Y$ are assigned **then**
        if $V(\text{decs}(I)) \cap X = \emptyset$ **then return** $M \vee \pi(I, X)$
        $M := M \vee \pi(I, X)$
        $b := \delta(\text{decs}(\pi(I, X)))$
        Backtrack(I, $b - 1$)
    else if Entails(I, F) **then**
        if $V(\text{decs}(I)) \cap X = \emptyset$ **then return** $M \vee \pi(I, X)$
        $M := M \vee \pi(I, X)$
        $b := \delta(\text{decs}(\pi(I, X)))$
        Backtrack(I, $b - 1$)
    else Decide(I, δ)

Calculus

EndTrue: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{EndTrue}} M \vee m$ if $V(\mathsf{decs}(I)) \cap X = \emptyset$ and $m \stackrel{\text{def}}{=} \pi(I, X)$ and $\forall X \exists Y\, [\, F|_I\,] = 1$

EndFalse: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{EndFalse}} M$ if exists $C \in F$ and $C|_I = 0$ and $\delta(C) = 0$

Unit: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{Unit}} (F, I\ell, M, \delta[\ell \mapsto a])$ if $F|_I \neq 0$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a \stackrel{\text{def}}{=} \delta(C \setminus \{\ell\})$

BackTrue: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{BackTrue}} (F, UK\ell, M \vee m, \delta[L \mapsto \infty][\ell \mapsto b])$ if $UV \stackrel{\text{def}}{=} I$ and $D \stackrel{\text{def}}{=} \overline{\pi(\mathsf{decs}(I), X)}$ and $b + 1 \stackrel{\text{def}}{=} \delta(D) \leqslant \delta(I)$ and $\ell \in D$ and $b = \delta(D \setminus \{\ell\}) = \delta(U)$ and $m \stackrel{\text{def}}{=} \pi(I, X)$ and $K \stackrel{\text{def}}{=} V_{\leqslant b}$ and $L \stackrel{\text{def}}{=} V_{>b}$ and $\forall X \exists Y\, [\, F|_I\,] = 1$

BackFalse: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{BackFalse}} (F, UK\ell, M, \delta[L \mapsto \infty][\ell \mapsto j])$ if exists $C \in F$ and exists D with $UV \stackrel{\text{def}}{=} I$ and $C|_I = 0$ and $c \stackrel{\text{def}}{=} \delta(C) = \delta(D) > 0$ such that $\ell \in D$ and $\bar{\ell} \in \mathsf{decs}(I)$ and $\bar{\ell}|_V = 0$ and $F \wedge \overline{M} \models D$ and $j \stackrel{\text{def}}{=} \delta(D \setminus \{\ell\})$ and $b \stackrel{\text{def}}{=} \delta(U) = c - 1$ and $K \stackrel{\text{def}}{=} V_{\leqslant b}$ and $L \stackrel{\text{def}}{=} V_{>b}$

DecideX: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{DecideX}} (F, I\ell^d, M, \delta[\ell \mapsto d])$ if $F|_I \neq 0$ and $\mathsf{units}(F|_I) = \emptyset$ and $\delta(\ell) = \infty$ and $d \stackrel{\text{def}}{=} \delta(I) + 1$ and $V(\ell) \in X$

DecideY: $(F, I, M, \delta) \rightsquigarrow_{\mathsf{DecideY}} (F, I\ell^d, M, \delta[\ell \mapsto d])$ if $F|_I \neq 0$ and $\mathsf{units}(F|_I) = \emptyset$ and $\delta(\ell) = \infty$ and $d \stackrel{\text{def}}{=} \delta(I) + 1$ and $V(\ell) \in Y$ and $X - I = \emptyset$

Outline

Unit Propagation

Idea: Assign the propagated unit literal the decision level of its reason clause

Unit Propagation

Idea: Assign the propagated unit literal the decision level of its reason clause

Unit:

Unit Propagation

Idea: Assign the propagated unit literal the decision level of its reason clause

Unit: $(F,\ I,\ M,\ \delta)$

Unit Propagation

Idea: Assign the propagated unit literal the decision level of its reason clause

Unit: $(F, I, M, \delta) \rightsquigarrow_{\textsf{Unit}} (F, I\ell, M, \delta[\ell \mapsto a])$ if $F|_I \neq 0$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a \stackrel{\textsf{def}}{=} \delta(C \setminus \{\ell\})$

Unit Propagation

Idea: Assign the propagated unit literal the decision level of its reason clause

Unit: $(F,\ I,\ M,\ \delta) \leadsto_{\mathsf{Unit}} (F,\ I\ell,\ \ M,\ \delta[\ell \mapsto a])$ if $F|_I \neq 0$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a \stackrel{\text{def}}{=} \delta(C \setminus \{\ell\})$

Backtracking upon Model Found

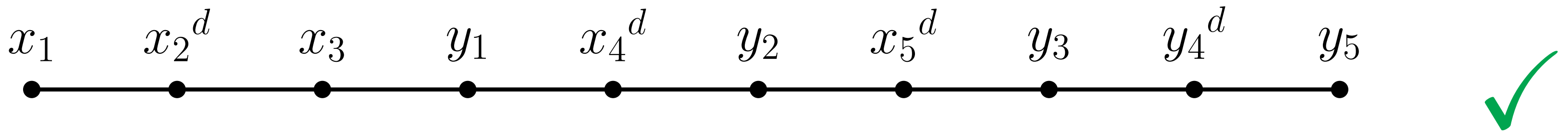
Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

Backtracking upon Model Found

Given: Formula $F(X,Y)$ over relevant variables X and irrelevant variables Y

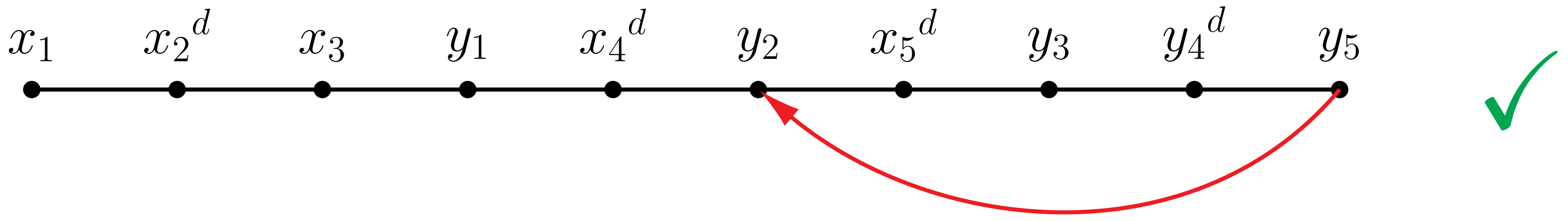
Idea: Flip the last relevant decision literal



Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

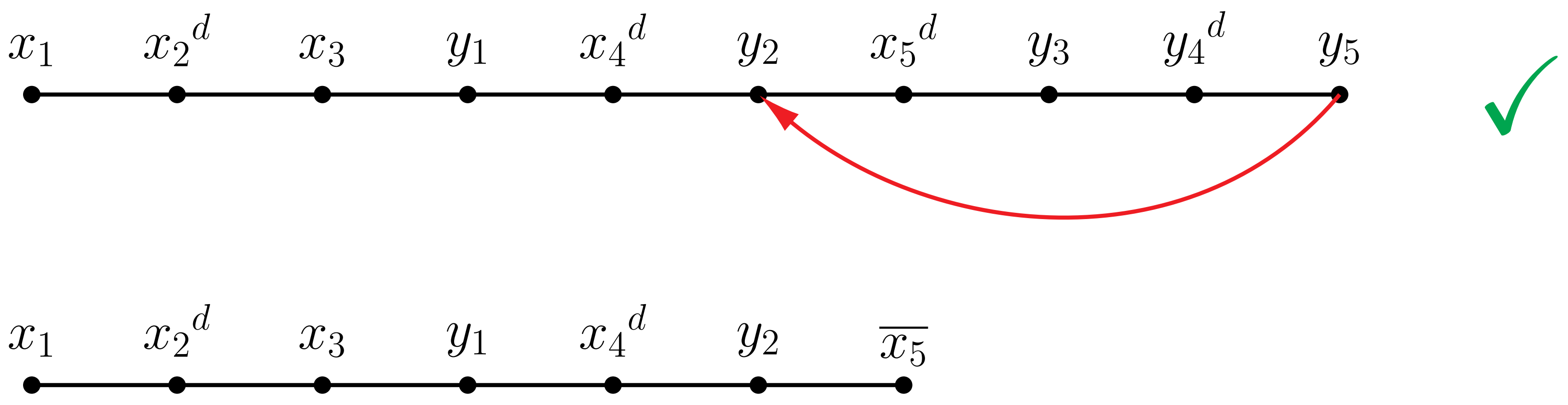
Idea: Flip the last relevant decision literal



Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

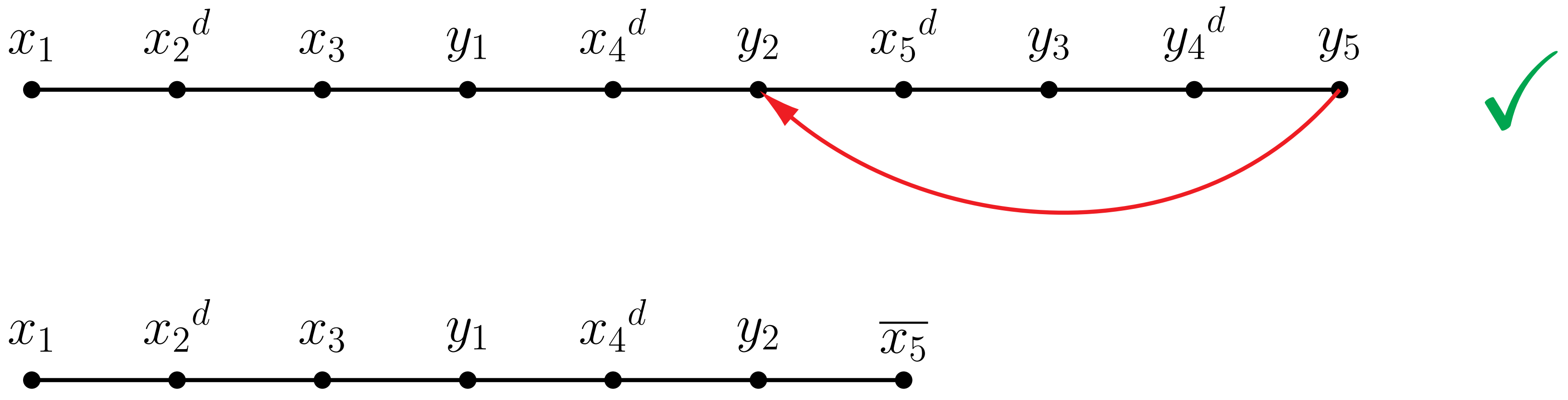
Idea: Flip the last relevant decision literal



Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

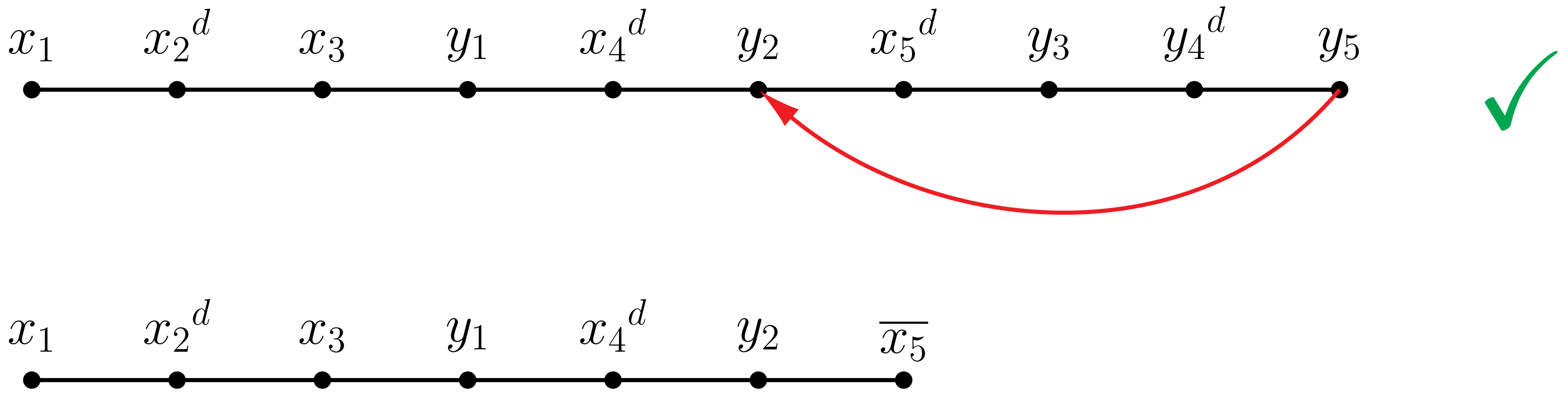


BackTrue:

Backtracking upon Model Found

Given: Formula $F(X,Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

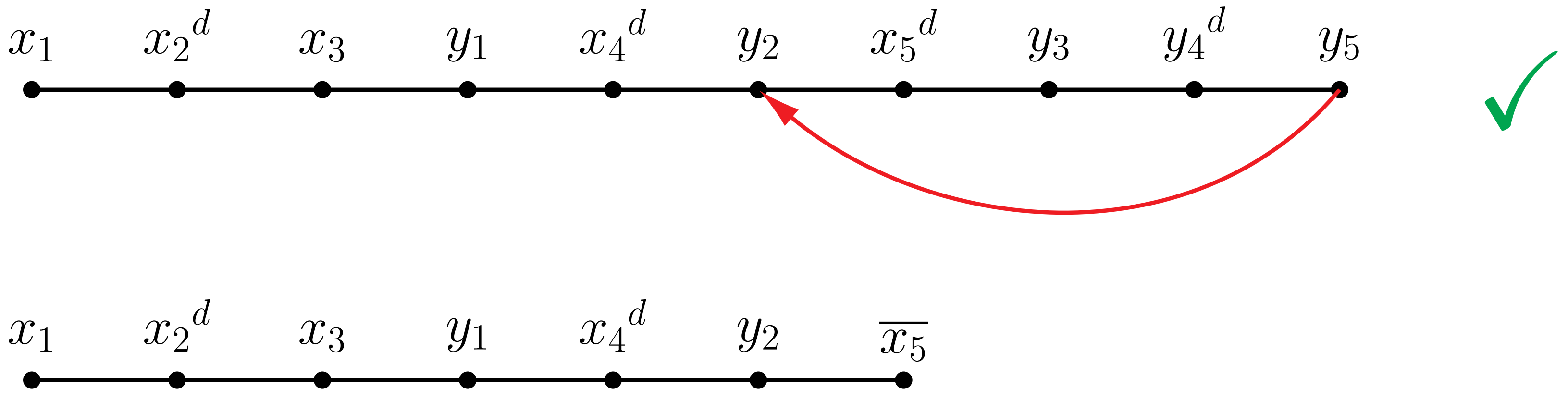


BackTrue: $(F,\ I,\ M,\ \delta)$

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

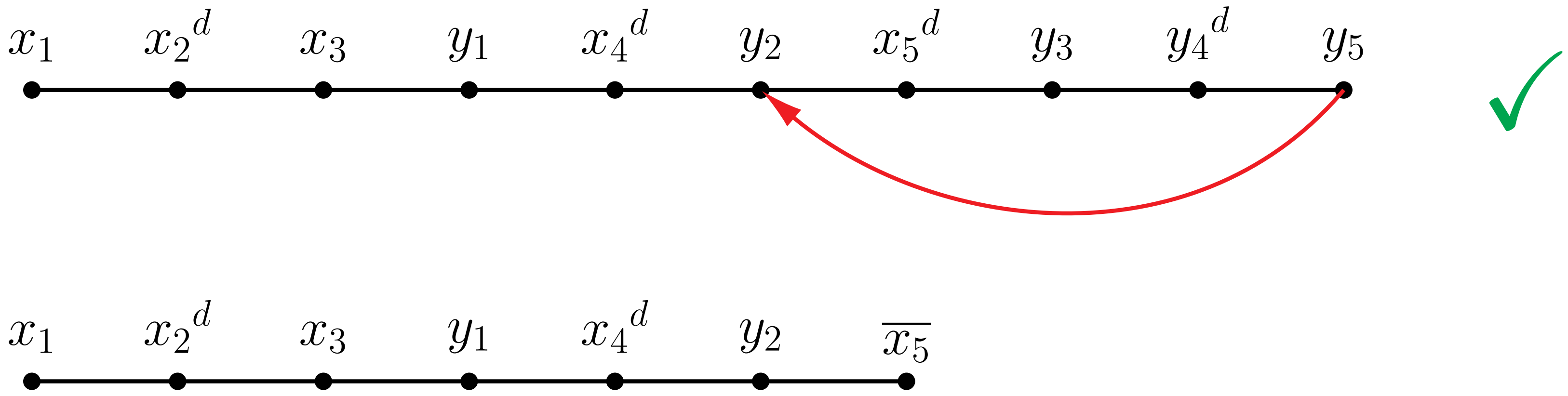


BackTrue: $(F,\ I,\ M,\ \delta)$ if $\forall X \exists Y\, [\, F|_I \,] = 1$

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

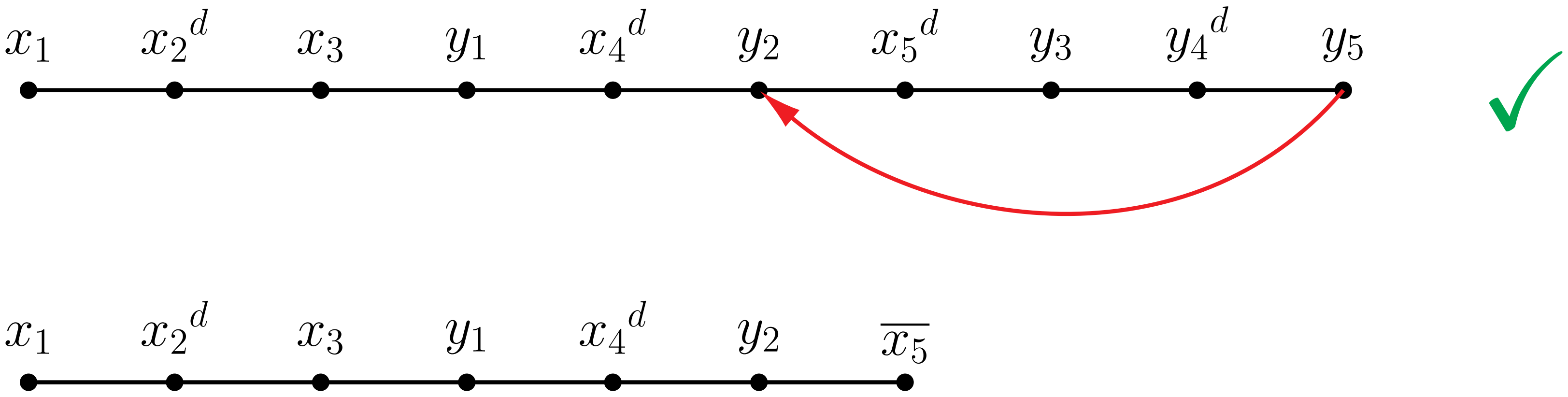


BackTrue: $(F,\ I,\ M,\ \delta) \rightsquigarrow_{\text{BackTrue}} (F,\qquad)$ if $\forall X \exists Y\, [\, F|_I \,] = 1$

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

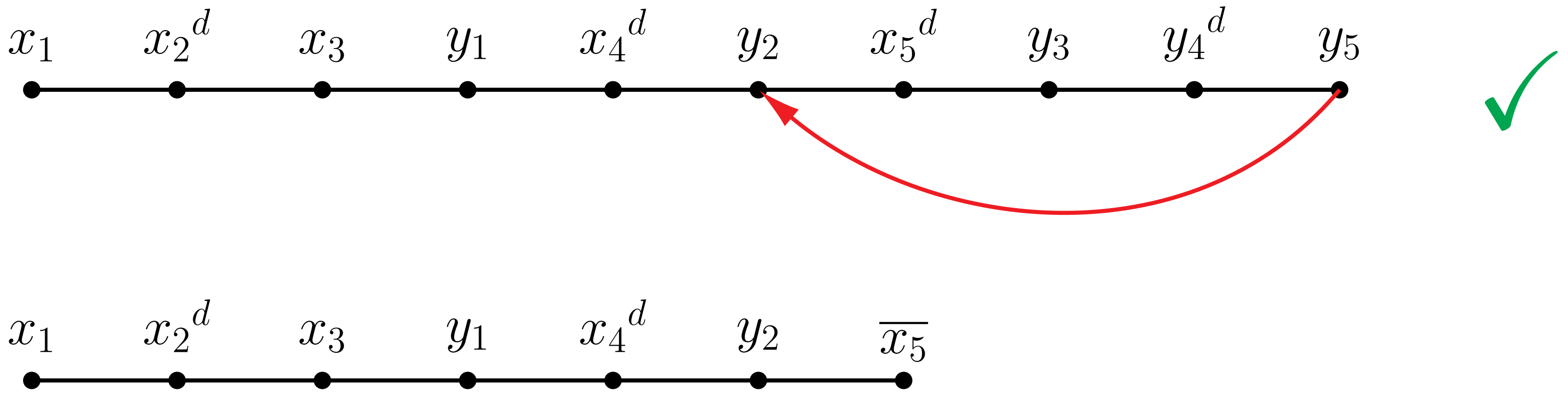


BackTrue: $(F,\ I,\ M,\ \delta) \rightsquigarrow_{\mathsf{BackTrue}} (F,\quad M \vee m, \quad)$ if $\forall X \exists Y\, [\, F|_I \,] = 1$ and $m \overset{\text{def}}{=} \pi(I, X)$ and

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

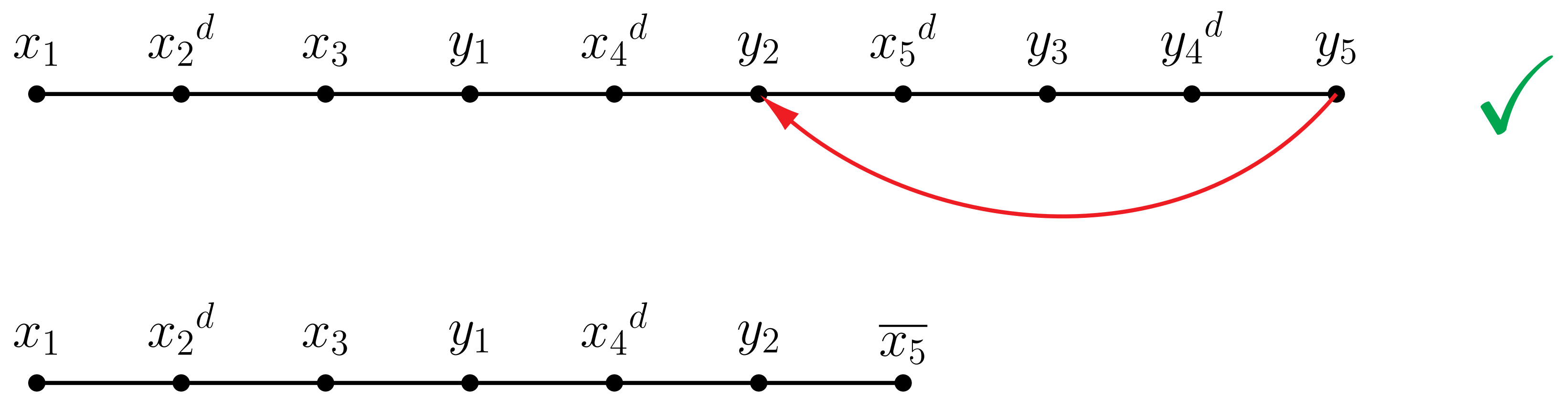


BackTrue: $(F,\ I,\ M,\ \delta) \leadsto_{\text{BackTrue}} (F,\ UK\ell,\ M + m,\qquad)$ if $\forall X \exists Y\,[\,F|_I\,] = 1$ and $m \overset{\text{def}}{=} \pi(I, X)$ and $D \overset{\text{def}}{=} \overline{\pi(\text{decs}(I), X)}$ and $\ell \in D$ and $UV \overset{\text{def}}{=} I$ and $K \overset{\text{def}}{=} V_{\leqslant b}$ and

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal

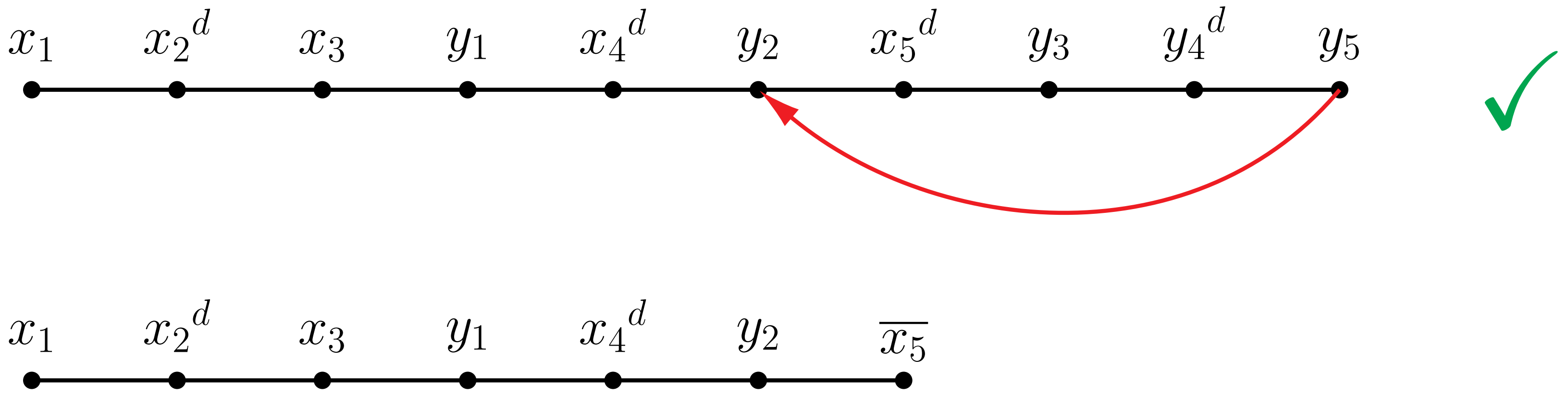


BackTrue: $(F,\, I,\, M,\, \delta) \rightsquigarrow_{\text{BackTrue}} (F,\, UK\ell,\, M + m,\, \delta[L \mapsto \infty][\ell \mapsto b])$ if $\forall X \exists Y\, [\, F|_I\,] = 1$ and $m \stackrel{\text{def}}{=} \pi(I, X)$ and $D \stackrel{\text{def}}{=} \overline{\pi(\text{decs}(I), X)}$ and $\ell \in D$ and $UV \stackrel{\text{def}}{=} I$ and $K \stackrel{\text{def}}{=} V_{\leqslant b}$ and $b = \delta(D \setminus \{\ell\}) = \delta(U)$ and $b + 1 \stackrel{\text{def}}{=} \delta(D) \leqslant \delta(I)$ and $L \stackrel{\text{def}}{=} V_{>b}$

Backtracking upon Model Found

Given: Formula $F(X, Y)$ over relevant variables X and irrelevant variables Y

Idea: Flip the last relevant decision literal



BackTrue: $(F,\, I,\, M,\, \delta) \rightsquigarrow_{\mathsf{BackTrue}} (F,\, UK\ell,\, M+m,\, \delta[L \mapsto \infty][\ell \mapsto b])$ if $\forall X \exists Y\, [\, F|_I\,] = 1$ and $m \stackrel{\text{def}}{=} \pi(I, X)$ and $D \stackrel{\text{def}}{=} \overline{\pi(\mathsf{decs}(I), X)}$ and $\ell \in D$ and $UV \stackrel{\text{def}}{=} I$ and $K \stackrel{\text{def}}{=} V_{\leqslant b}$ and $b = \delta(D \setminus \{\ell\}) = \delta(U)$ and $b + 1 \stackrel{\text{def}}{=} \delta(D) \leqslant \delta(I)$ and $L \stackrel{\text{def}}{=} V_{>b}$

Outline

Conclusion

Our Contribution

Method for computing partial assignments entailing the formula on the fly

- Inspired by the interaction of theory and SAT solvers in SMT
- Combines dual reasoning and chronological CDCL
- Algorithm (in the paper)
- Formalization (in the paper)

Entailment test in four flavors of increasing strength

- $F|_I = 1$ (syntactic check)
- $F|_I \approx 1$ (incomplete check in $\mathbf{P}$)
- $F|_I \equiv 1$ (semantic check in $\mathbf{coNP}$)
- $\forall X \exists Y \, [F|_I] = 1$ (check in $\mathbf{\Pi}_2^P$)

Conclusion

Our Contribution

Method for computing partial assignments entailing the formula on the fly

- Inspired by the interaction of theory and SAT solvers in SMT
- Combines dual reasoning and chronological CDCL
- Algorithm (in the paper)
- Formalization (in the paper)

Entailment test in four flavors of increasing strength

- $F|_I = 1$ (syntactic check)
- $F|_I \approx 1$ (incomplete check in $\mathbf{P}$)
- $F|_I \equiv 1$ (semantic check in $\mathbf{coNP}$)
- $\forall X \exists Y \, [F|_I\,] = 1$ (check in $\mathbf{\Pi}_2^P$)

Further Research

- Implement and validate our method
- Target weighted model integration and model counting with or without projection
- Investigate methods concerning the implementation of QBF oracles
  - Dependency schemes (Samer and Szeider, JAR, 2009)
  - Incremental QBF (Lonsing and Egly, CP'14)
- Combine with decomposition-based approaches and generate d-DNNF