Addressing Proposition Model Counting and Enumeration with and without Projection

Sibylle Möhle

Institute for Formal Models and Verification
LIT Secure and Correct Systems Lab

JOHANNES KEPLER
UNIVERSITY LINZ

July 16, 2020

Who Wants the Model Count Anyway?

Planning

Software Verification

Cryptography

Hardware Verification

Bayesian Networks

Probabilistic Reasoning

Product Configuration

Planning

Software Verification

Cryptography

Hardware Verification

???

Bayesian Networks

Probabilistic Reasoning

Product Configuration

Projected Propositional Model Counting

SAT: Propositional satisfiability problem

- Is the propositional formula F satisfiable?

  Example $F = x \vee \overline{x}y$ is satisfiable: $I = x\overline{y}$ is a model of F

Projected Propositional Model Counting

SAT: Propositional satisfiability problem

- Is the propositional formula F satisfiable?

  Example $F = x \vee \overline{x}y$ is satisfiable: $I = x\overline{y}$ is a model of F

#SAT: Counting problem associated with SAT

- How many total models has F?

  Example $F = x \vee \overline{x}y$ and $\#F = 3$: models$(F) = \{xy, x\overline{y}, \overline{x}y\}$

Projected Propositional Model Counting

SAT: Propositional satisfiability problem

- Is the propositional formula F satisfiable?

  Example $F = x \vee \overline{x}y$ is satisfiable: $I = x\overline{y}$ is a model of F

#SAT: Counting problem associated with SAT

- How many total models has F?

  Example $F = x \vee \overline{x}y$ and $\#F = 3$: models$(F) = \{xy, x\overline{y}, \overline{x}y\}$

#∃SAT: Projected propositional model counting

- How many models has F projected onto x?

  Example $F = x \vee \overline{x}y$ and $\#\exists y\,[F\,] = 2$: models$(\exists y\,[F\,]) = \{x, \overline{x}\}$

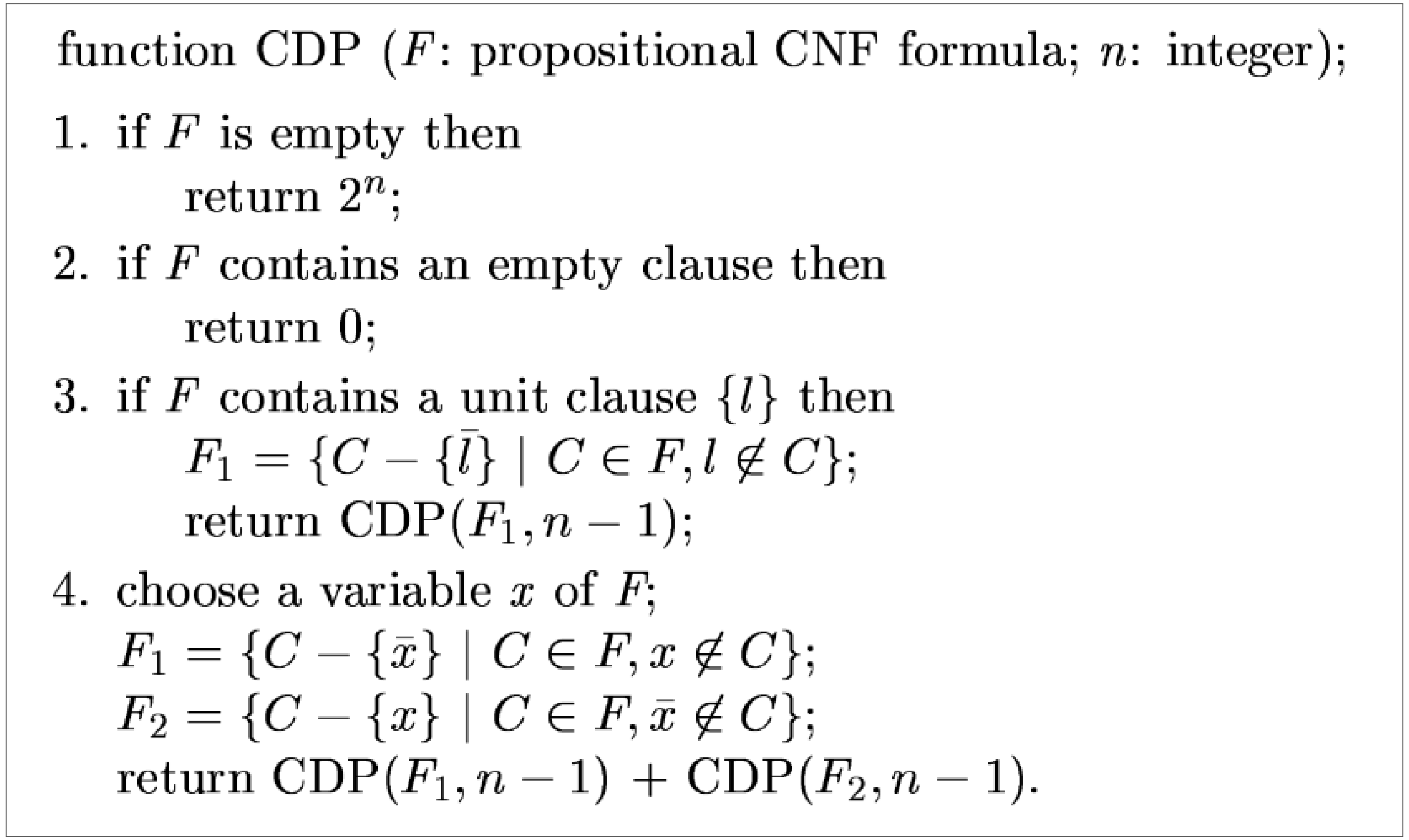
Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

Counting Davis-Putnam (CDP)[1]



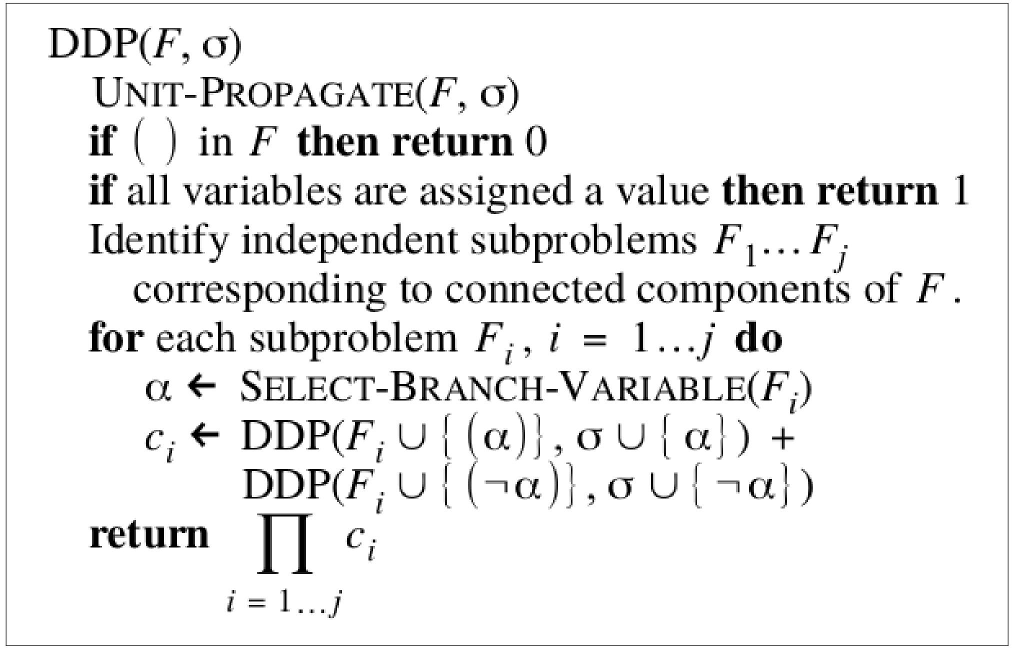
```
function CDP ($F$: propositional CNF formula; $n$: integer);
1. if $F$ is empty then
       return $2^n$;
2. if $F$ contains an empty clause then
       return 0;
3. if $F$ contains a unit clause $\{l\}$ then
       $F_1 = \{C - \{\bar{l}\} \mid C \in F, l \notin C\}$;
       return CDP$(F_1, n-1)$;
4. choose a variable $x$ of $F$;
   $F_1 = \{C - \{\bar{x}\} \mid C \in F, x \notin C\}$;
   $F_2 = \{C - \{x\} \mid C \in F, \bar{x} \notin C\}$;
   return CDP$(F_1, n-1)$ + CDP$(F_2, n-1)$.
```

(Source: [1])

---

[1] E. Birnbaum, E.L. Lozinskii, "The Good Old Davis-Putnam Procedure Helps Counting Models", JAIR, 1999.

Decomposing-Davis-Putnam [2]



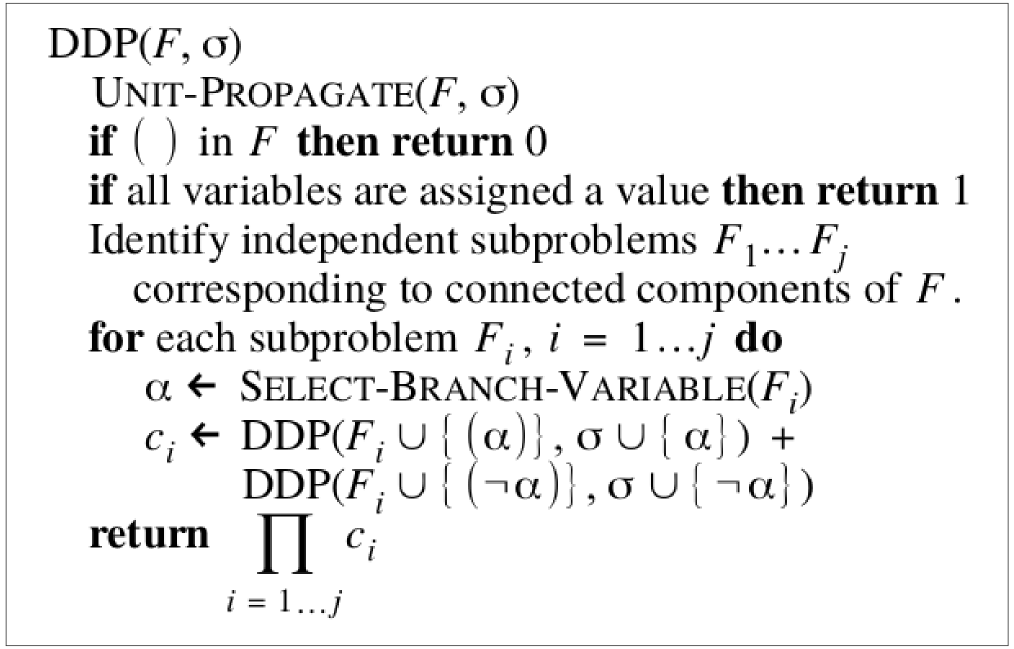
```
DDP($F$, $\sigma$)
    UNIT-PROPAGATE($F$, $\sigma$)
    if $(\ )$ in $F$ then return 0
    if all variables are assigned a value then return 1
    Identify independent subproblems $F_1 \ldots F_j$
        corresponding to connected components of $F$.
    for each subproblem $F_i$, $i = 1 \ldots j$ do
        $\alpha \leftarrow$ SELECT-BRANCH-VARIABLE($F_i$)
        $c_i \leftarrow$ DDP($F_i \cup \{(\alpha)\}, \sigma \cup \{\alpha\}$) +
                DDP($F_i \cup \{(\neg\alpha)\}, \sigma \cup \{\neg\alpha\}$)
    return $\prod_{i = 1 \ldots j} c_i$
```

(Source: [2])

---

[2] R.J. Bayardo, J.D. Pehoushek, "Counting Models Using Connected Components", AAAI'00.

Decomposing-Davis-Putnam [2]



```
DDP($F$, $\sigma$)
  UNIT-PROPAGATE($F$, $\sigma$)
  if ( ) in $F$ then return 0
  if all variables are assigned a value then return 1
  Identify independent subproblems $F_1 \ldots F_j$
    corresponding to connected components of $F$.
  for each subproblem $F_i$, $i = 1 \ldots j$ do
    $\alpha$ ← SELECT-BRANCH-VARIABLE($F_i$)
    $c_i$ ← DDP($F_i \cup \{(\alpha)\}$, $\sigma \cup \{\alpha\}$) +
            DDP($F_i \cup \{(\neg\alpha)\}$, $\sigma \cup \{\neg\alpha\}$)
  return $\prod_{i = 1 \ldots j} c_i$
```

(Source: [2])

Enhancements

- Component caching [3]
- Efficient binary constraint propagation (BCP) [4]
- Parallel version [5]
- Distributed version [6]

---

[2] R.J. Bayardo, J.D. Pehoushek, "Counting Models Using Connected Components", AAAI'00.

[3] T. Sang et al., "Combining Component Caching and Clause Learning for Effective Model Counting", SAT'04.

[4] M. Thurley, "sharpSAT – Counting Models with Advanced Component Caching and Implicit BCP", SAT'06.

[5] J. Burchard, T. Schubert, B. Becker, "Laissez-Faire Caching for Parallel #SAT Solving", SAT'15.

[6] J. Burchard, T. Schubert, B. Becker, "Distributed Parallel #SAT Solving", CLUSTER'16.

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

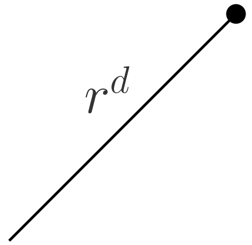
The Search Space Needs to be Explored Exhaustively

$F \quad = (\bar{p} \vee q) \wedge (p \vee q) \qquad M = 0$

$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

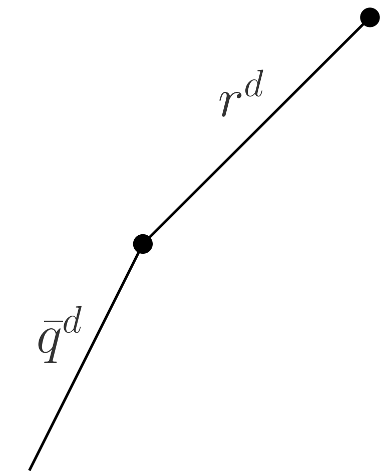
$$\begin{aligned} F &= (\bar{p} \vee q) \wedge (p \vee q) & M = 0 \\ F|_r &= (\bar{p} \vee q) \wedge (p \vee q) & M = 0 \end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

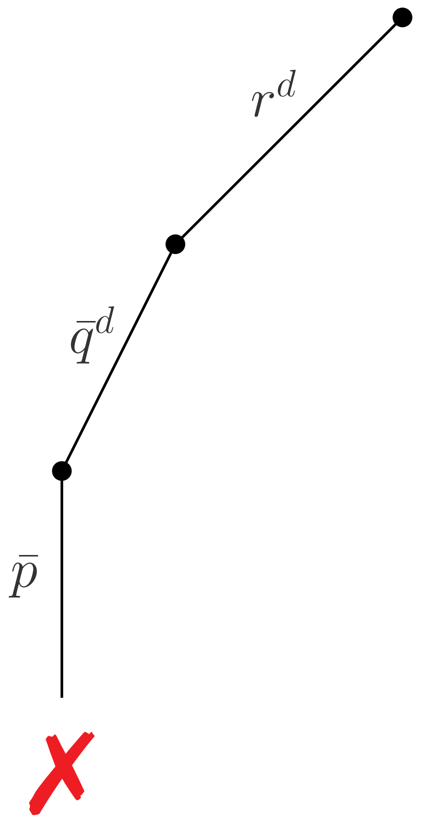
$$\begin{aligned} F &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\ F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\ F|_{r\bar{q}} &= (\bar{p}) \wedge (p) && M = 0 \end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

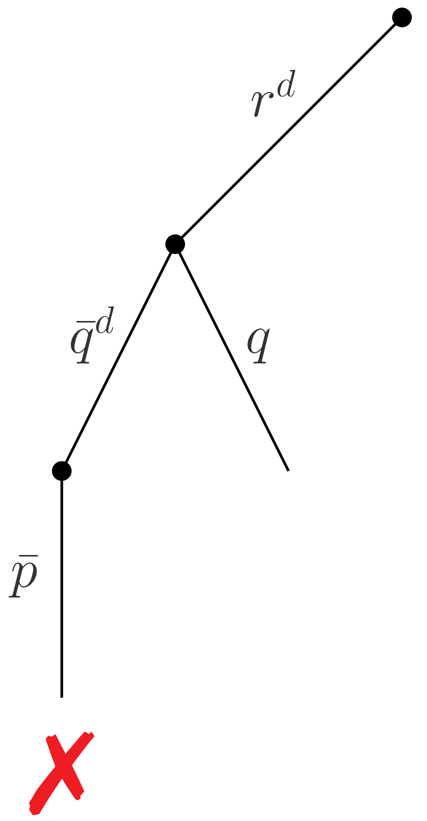
$$\begin{aligned} F &= (\bar{p} \lor q) \land (p \lor q) & M &= 0 \\ F|_r &= (\bar{p} \lor q) \land (p \lor q) & M &= 0 \\ F|_{r\bar{q}} &= (\bar{p}) \land (p) & M &= 0 \\ F|_{r\bar{q}\bar{p}} &= \bot & M &= 0 \end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

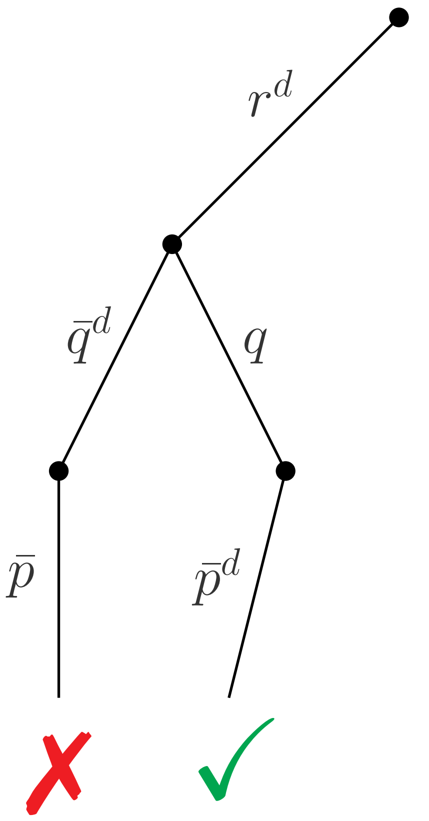
$$\begin{aligned}
F &= (\bar{p} \vee q) \wedge (p \vee q) & M &= 0 \\
F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) & M &= 0 \\
F|_{r\bar{q}} &= (\bar{p}) \wedge (p) & M &= 0 \\
F|_{r\bar{q}\bar{p}} &= \bot & M &= 0 \\
F|_{rq} &= \top & M &= 0
\end{aligned}$$



$$V = \{p, q, r\}$$

The Search Space Needs to be Explored Exhaustively

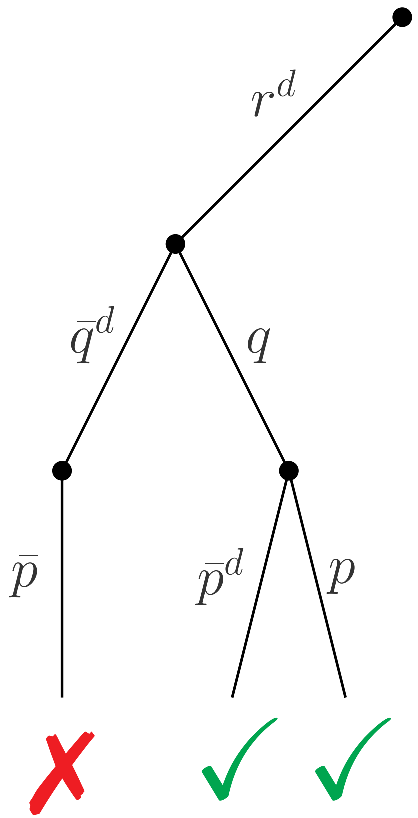
$$\begin{aligned}
F &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \wedge (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1
\end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

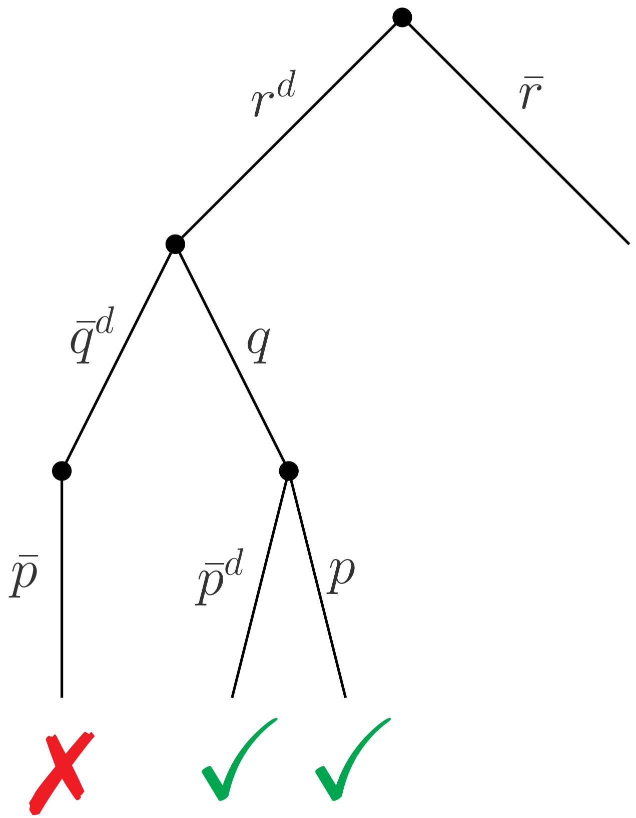
$$\begin{array}{llcl}
F & = (\bar{p} \vee q) \wedge (p \vee q) & \qquad & M = 0 \\
F|_{r} & = (\bar{p} \vee q) \wedge (p \vee q) & & M = 0 \\
F|_{r\bar{q}} & = (\bar{p}) \wedge (p) & & M = 0 \\
F|_{r\bar{q}\bar{p}} & = \bot & & M = 0 \\
F|_{rq} & = \top & & M = 0 \\
F|_{rq\bar{p}} & = \top & & M = 1 \\
F|_{rqp} & = \top & & M = 2
\end{array}$$



$$V = \{p, q, r\}$$

The Search Space Needs to be Explored Exhaustively

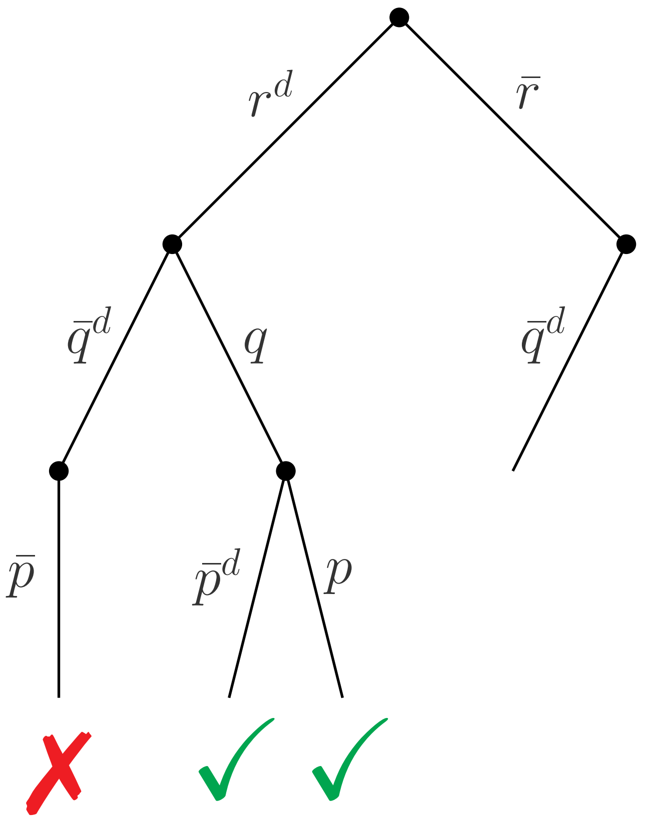
$$\begin{aligned}
F &= (\bar{p} \lor q) \land (p \lor q) && M = 0 \\
F|_{r} &= (\bar{p} \lor q) \land (p \lor q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \land (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1 \\
F|_{rqp} &= \top && M = 2 \\
F|_{\bar{r}} &= (\bar{p} \lor q) \land (p \lor q) && M = 2
\end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

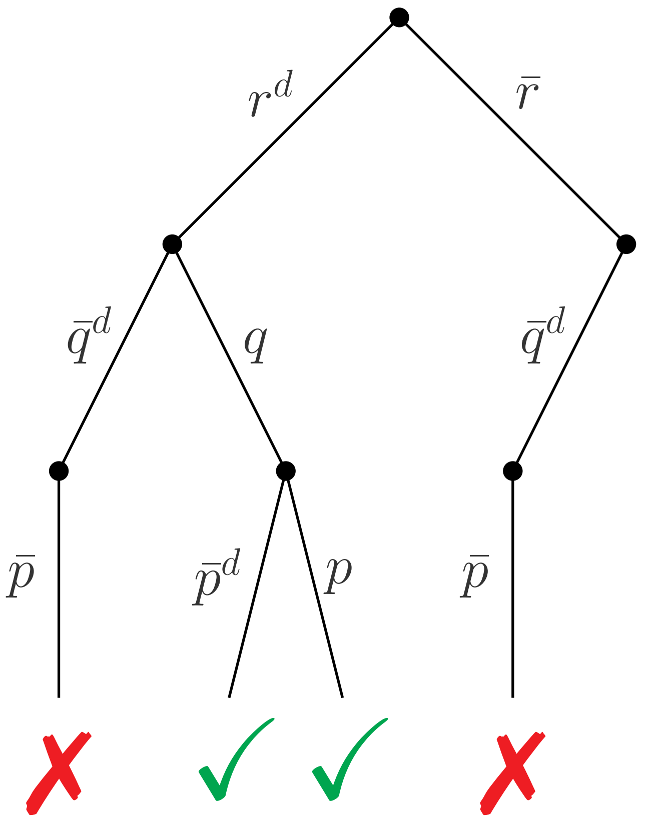
$$\begin{aligned}
F &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \wedge (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1 \\
F|_{rqp} &= \top && M = 2 \\
F|_{\bar{r}} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 2 \\
F|_{\bar{r}\bar{q}} &= (\bar{p}) \wedge (p) && M = 2
\end{aligned}$$



$$V = \{p, q, r\}$$

The Search Space Needs to be Explored Exhaustively

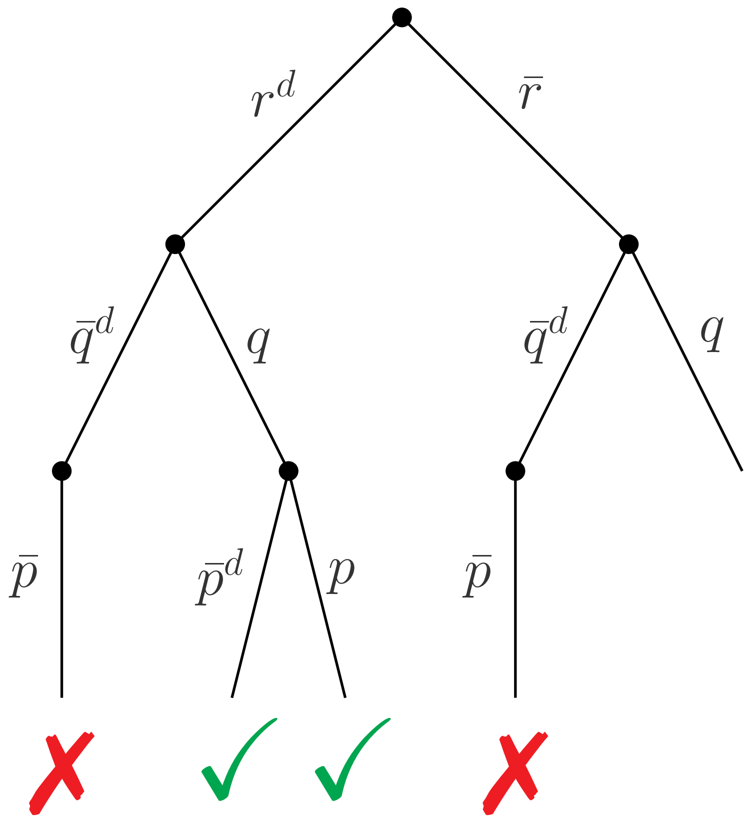
$$\begin{aligned}
F &= (\bar{p} \lor q) \land (p \lor q) && M = 0 \\
F|_{r} &= (\bar{p} \lor q) \land (p \lor q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \land (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1 \\
F|_{rqp} &= \top && M = 2 \\
F|_{\bar{r}} &= (\bar{p} \lor q) \land (p \lor q) && M = 2 \\
F|_{\bar{r}\bar{q}} &= (\bar{p}) \land (p) && M = 2 \\
F|_{\bar{r}\bar{q}\bar{p}} &= \bot && M = 2
\end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

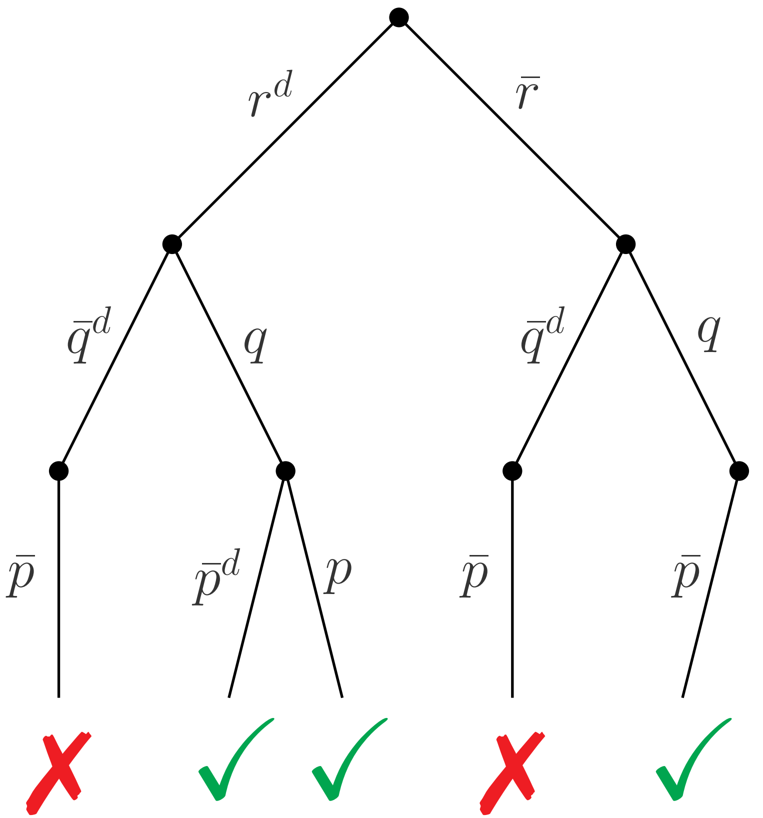
$$\begin{aligned}
F \quad &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0\\
F|_{r} \quad &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0\\
F|_{r\bar{q}} \quad &= (\bar{p}) \wedge (p) && M = 0\\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0\\
F|_{rq} \quad &= \top && M = 0\\
F|_{rq\bar{p}} &= \top && M = 1\\
F|_{rqp} &= \top && M = 2\\
F|_{\bar{r}} \quad &= (\bar{p} \vee q) \wedge (p \vee q) && M = 2\\
F|_{\bar{r}\bar{q}} \quad &= (\bar{p}) \wedge (p) && M = 2\\
F|_{\bar{r}\bar{q}\bar{p}} &= \bot && M = 2\\
F|_{\bar{r}q} \quad &= \top && M = 2
\end{aligned}$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

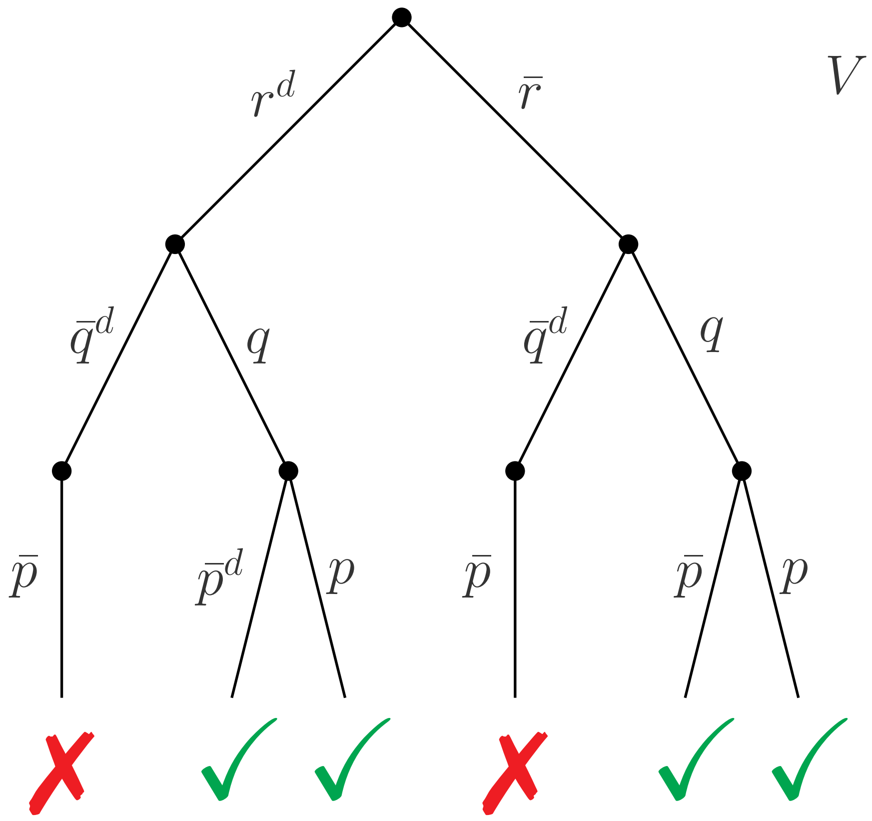
$$
\begin{aligned}
F &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \wedge (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1 \\
F|_{rqp} &= \top && M = 2 \\
F|_{\bar{r}} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 2 \\
F|_{\bar{r}\bar{q}} &= (\bar{p}) \wedge (p) && M = 2 \\
F|_{\bar{r}\bar{q}\bar{p}} &= \bot && M = 2 \\
F|_{\bar{r}q} &= \top && M = 2 \\
F|_{\bar{r}q\bar{p}} &= \top && M = 3
\end{aligned}
$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

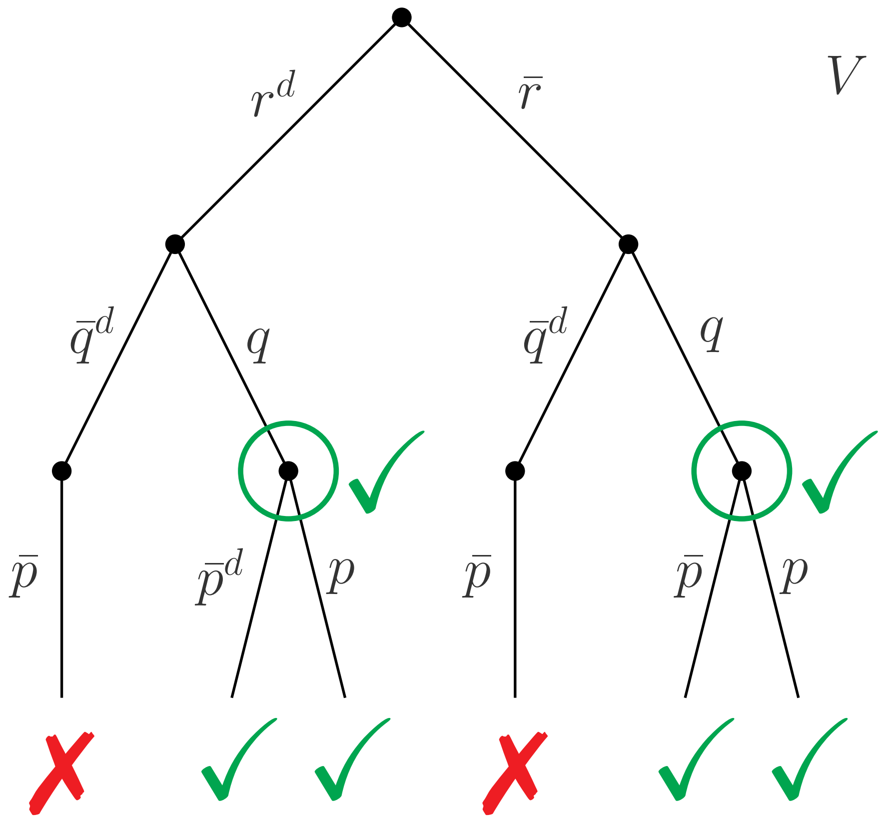
$$
\begin{aligned}
F &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 0 \\
F|_{r\bar{q}} &= (\bar{p}) \wedge (p) && M = 0 \\
F|_{r\bar{q}\bar{p}} &= \bot && M = 0 \\
F|_{rq} &= \top && M = 0 \\
F|_{rq\bar{p}} &= \top && M = 1 \\
F|_{rqp} &= \top && M = 2 \\
F|_{\bar{r}} &= (\bar{p} \vee q) \wedge (p \vee q) && M = 2 \\
F|_{\bar{r}\bar{q}} &= (\bar{p}) \wedge (p) && M = 2 \\
F|_{\bar{r}\bar{q}\bar{p}} &= \bot && M = 2 \\
F|_{\bar{r}q} &= \top && M = 2 \\
F|_{\bar{r}q\bar{p}} &= \top && M = 3 \\
F|_{\bar{r}qp} &= \top && M = 4
\end{aligned}
$$



$V = \{p, q, r\}$

The Search Space Needs to be Explored Exhaustively

$$\begin{array}{lll}
F & = (\bar{p} \vee q) \wedge (p \vee q) & M = 0 \\
F|_{r} & = (\bar{p} \vee q) \wedge (p \vee q) & M = 0 \\
F|_{r\bar{q}} & = (\bar{p}) \wedge (p) & M = 0 \\
F|_{r\bar{q}\bar{p}} & = \bot & M = 0 \\
F|_{rq} & = \top & M = 0 \\
F|_{rq\bar{p}} & = \top & M = 1 \\
F|_{rqp} & = \top & M = 2 \\
F|_{\bar{r}} & = (\bar{p} \vee q) \wedge (p \vee q) & M = 2 \\
F|_{\bar{r}\bar{q}} & = (\bar{p}) \wedge (p) & M = 2 \\
F|_{\bar{r}\bar{q}\bar{p}} & = \bot & M = 2 \\
F|_{\bar{r}q} & = \top & M = 2 \\
F|_{\bar{r}q\bar{p}} & = \top & M = 3 \\
F|_{\bar{r}qp} & = \top & M = 4
\end{array}$$

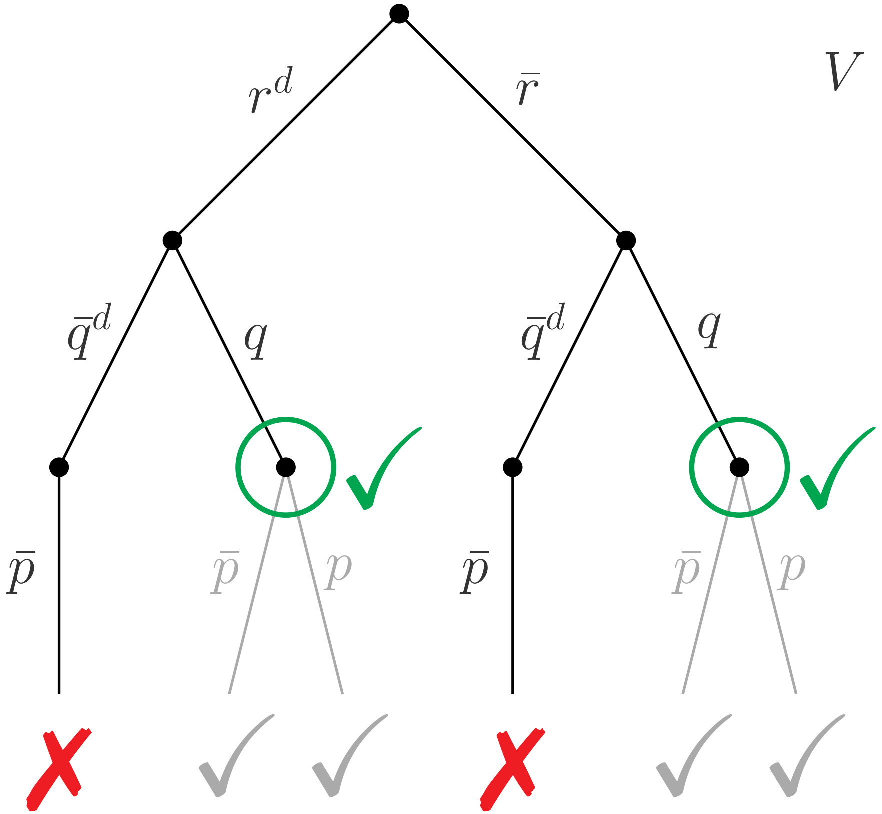


$V = \{p, q, r\}$

CDCL is biased towards conflicts!

The Search Space Needs to be Explored Exhaustively

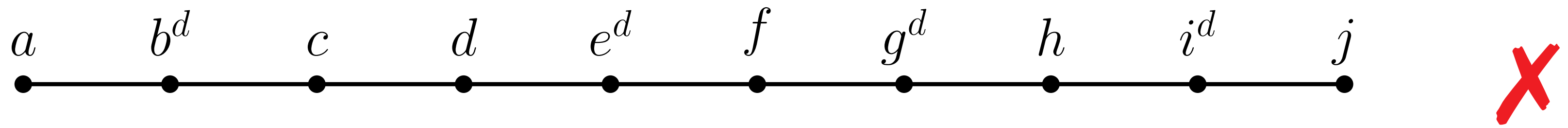
$$
\begin{aligned}
&F &&= (\bar{p} \vee q) \wedge (p \vee q) && M = 0\\
&F|_{r} &&= (\bar{p} \vee q) \wedge (p \vee q) && M = 0\\
&F|_{r\bar{q}} &&= (\bar{p}) \wedge (p) && M = 0\\
&F|_{r\bar{q}\bar{p}} &&= \bot && M = 0\\
&F|_{rq} &&= \top && M = 0\\
&F|_{rq\bar{p}} &&= \top && M = 1\\
&F|_{rqp} &&= \top && M = 2\\
&F|_{\bar{r}} &&= (\bar{p} \vee q) \wedge (p \vee q) && M = 2\\
&F|_{\bar{r}\bar{q}} &&= (\bar{p}) \wedge (p) && M = 2\\
&F|_{\bar{r}\bar{q}\bar{p}} &&= \bot && M = 2\\
&F|_{\bar{r}q} &&= \top && M = 2\\
&F|_{\bar{r}q\bar{p}} &&= \top && M = 3\\
&F|_{\bar{r}qp} &&= \top && M = 4
\end{aligned}
$$



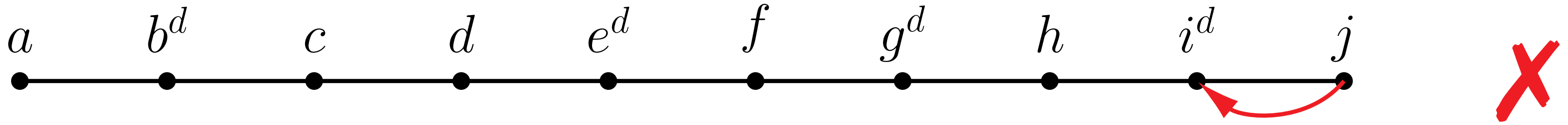
$V = \{p, q, r\}$

CDCL is biased towards conflicts!

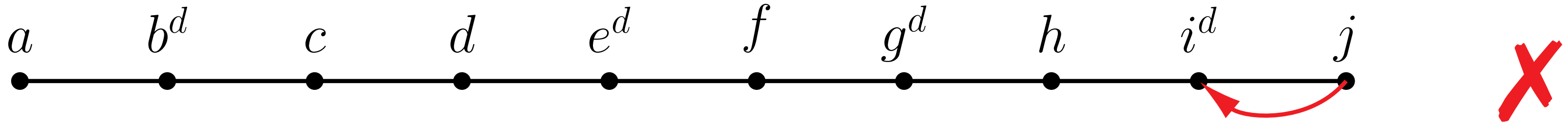
Counting by Means of the Davis-Putnam-Logemann-Loveland (DPLL) Algorithm

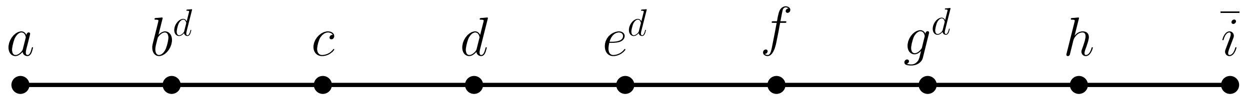


Counting by Means of the Davis-Putnam-Logemann-Loveland (DPLL) Algorithm

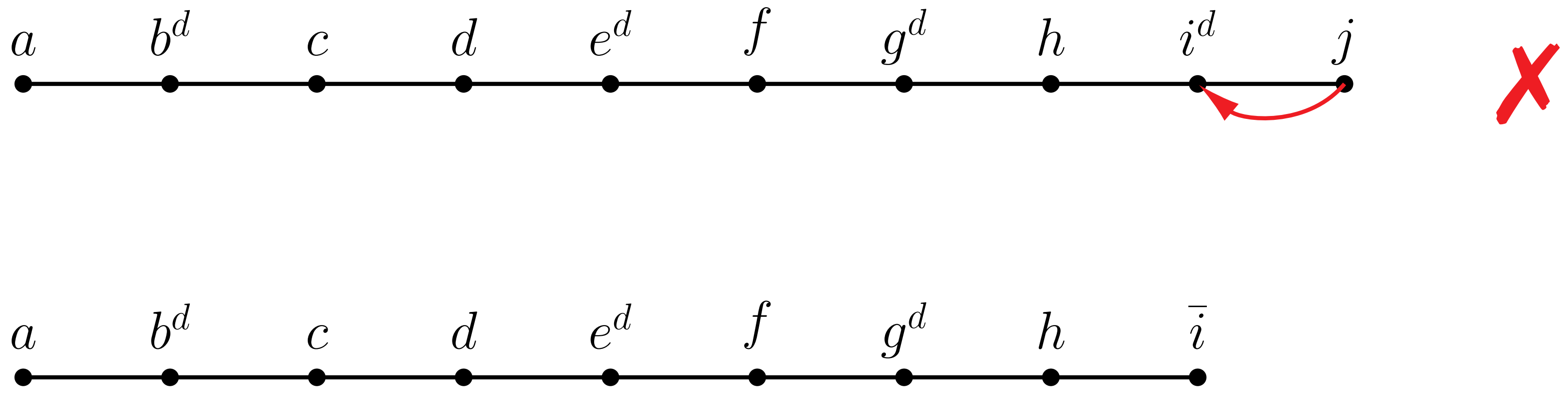


Counting by Means of the Davis-Putnam-Logemann-Loveland (DPLL) Algorithm





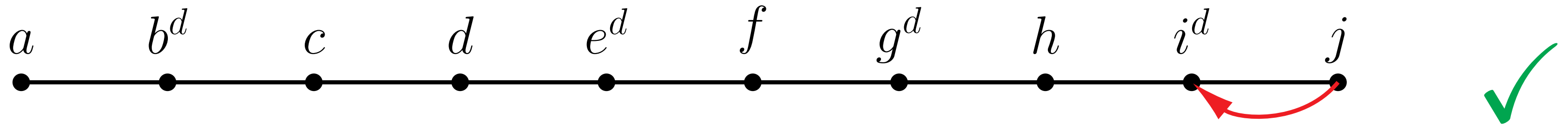
Counting by Means of the Davis-Putnam-Logemann-Loveland (DPLL) Algorithm



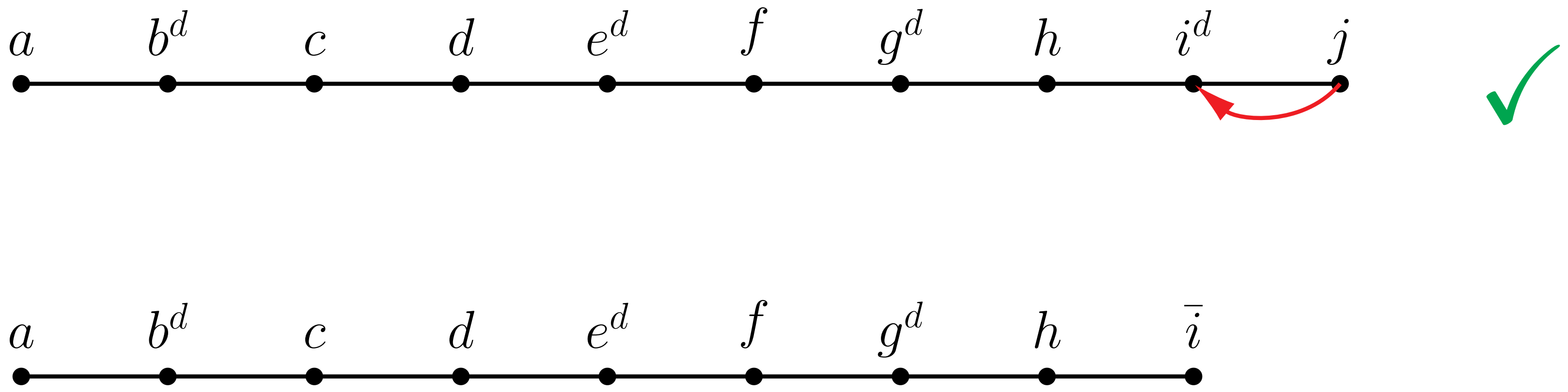
Suitability for #SAT

- \+ Search space is traversed in an ordered manner
- \+ The correct model count is returned
- – Regions of the search space without solution can not be escaped easily
- – Less efficient than with learning

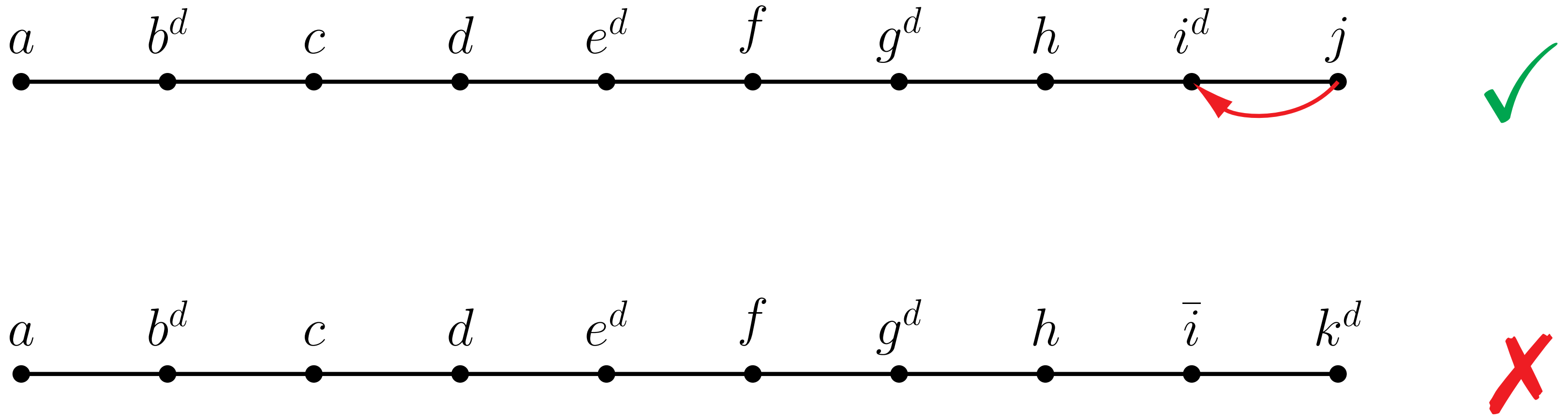
Non-Chronological Backtracking with Conflict-Driven Clause Learning (CDCL)



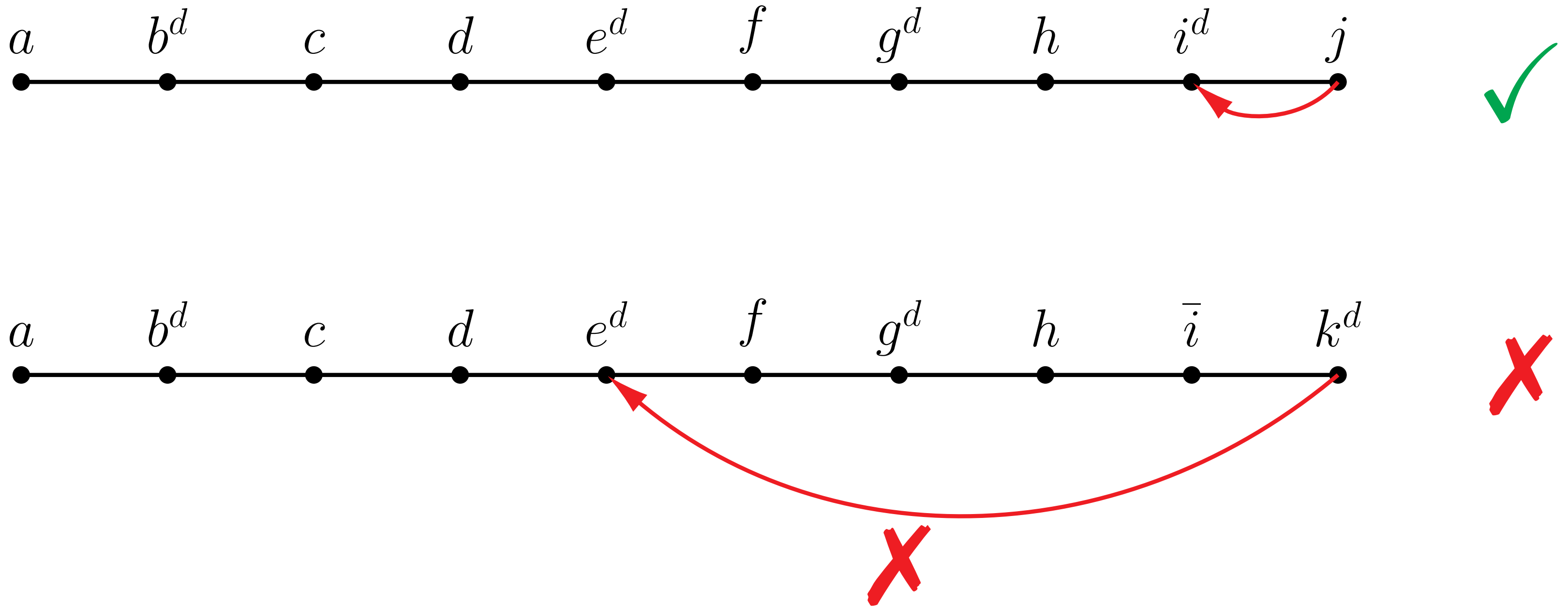
Non-Chronological Backtracking with Conflict-Driven Clause Learning (CDCL)



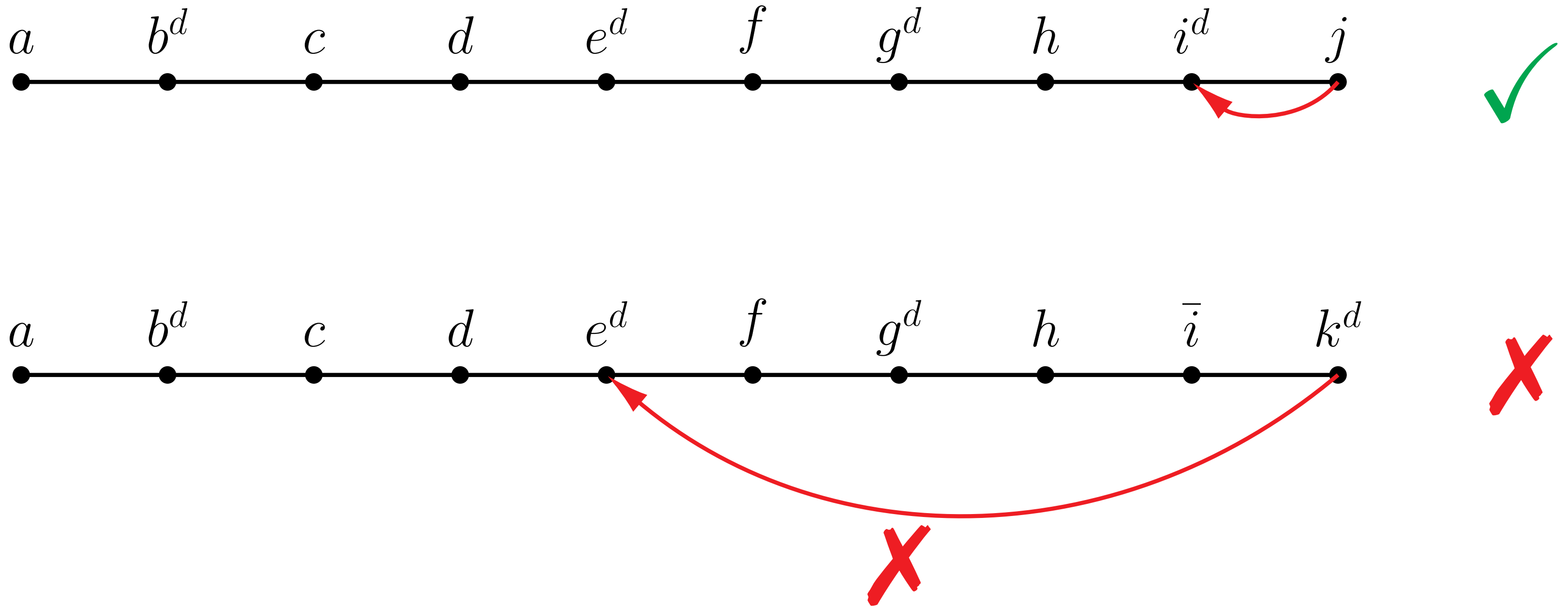
Non-Chronological Backtracking with Conflict-Driven Clause Learning (CDCL)



Non-Chronological Backtracking with Conflict-Driven Clause Learning (CDCL)



Non-Chronological Backtracking with Conflict-Driven Clause Learning (CDCL)



Suitability for #SAT

- \+ Enables the solver to escape regions of the search space with no solution
- \+ Gain in performance (for SAT)
- – Might result in a wrong model count
- – Might lead to redundant work

Chronological Conflict-Driven Clause Learning (Chronological CDCL)

Suitability for #SAT

- + Enables the solver to escape regions of the search space with no solution
- + Returns the correct model count
- + Avoids (at least some) redundant work
- + Does not degrade solver performance of state-of-the-art SAT solvers

Challenges and Solutions

Challenge	Addressed by		
	Dual reasoning [7,8]	Chronological CDCL [9,10,11]	Logical entailment [12]
No expensive satisfiability checks	✓		(✓)
No exponential learning	(✓)	✓	✓
Good learning	✓		(✓)
Early model detection	✓		✓
Pruning of search space	✓		✓

[7] A. Biere, S. Hölldobler, S. Möhle, "An Abstract Dual Propositional Model Counter", YSIP'17.

[8] S. Möhle, A. Biere, "Dualizing Projected Model Counting", ICTAI'18.

[9] S. Möhle, A. Biere, "Combining Conflict-Driven Clause Learning and Chronological Backtracking for Propositional Model Counting", GCAI'19.

[10] A. Nadel, V. Ryvchin, "Chronological Backtracking", SAT'18.

[11] S. Möhle, A. Biere, "Backing Backtracking", SAT'19.

[12] S. Möhle, R. Sebastiani, A. Biere, "Four Flavors of Logical Entailment", SAT'20.

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

Projected Model Counting

$F(X, Y)$ (arbitrary) propositional formula over sets of variables X and Y, where

X	*relevant input* variables		
Y	*irrelevant input* variables	and	$X \cap Y = \emptyset$

Projected Model Counting

$F(X, Y)$ (arbitrary) propositional formula over sets of variables X and Y, where

X *relevant input* variables
Y *irrelevant input* variables and $X \cap Y = \emptyset$

We are interested in the number of models projected onto X: $\#\exists Y\, [\, F(X, Y)\,]$

Projected Model Counting

$F(X, Y)$ (arbitrary) propositional formula over sets of variables X and Y, where

X *relevant input* variables
Y *irrelevant input* variables and $X \cap Y = \emptyset$

We are interested in the number of models projected onto X: $\#\exists Y\,[\,F(X, Y)\,]$

Example $F(X, Y) = x \vee y$

$X = \{x, y\}$ $Y = \emptyset$ models$(\exists Y\,[\,F(X, Y)\,]) = \{xy, x\bar{y}, \bar{x}y\}$ $\#\exists Y\,[\,F(X, Y)\,] = 3 = \#F(X, Y)$

Projected Model Counting

$F(X,Y)$ (arbitrary) propositional formula over sets of variables X and Y, where

X *relevant input* variables
Y *irrelevant input* variables and $X \cap Y = \emptyset$

We are interested in the number of models projected onto X: $\#\exists Y\,[\,F(X,Y)\,]$

Example $F(X,Y) = x \vee y$

$X = \{x,y\}$	$Y = \emptyset$	models$(\exists Y\,[\,F(X,Y)\,]) = \{xy, x\overline{y}, \overline{x}y\}$	$\#\exists Y\,[\,F(X,Y)\,] = 3 = \#F(X,Y)$
$X = \{x\}$	$Y = \{y\}$	models$(\exists Y\,[\,F(X,Y)\,]) = \{x, \overline{x}\}$	$\#\exists Y\,[\,F(X,Y)\,] = 2$

Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$

$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$

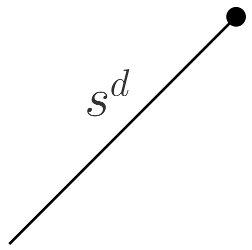
$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0

Example

$$F(X,Y) = p \vee q \vee r \vee s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$
$$\neg F(X,Y) = \bar{p} \wedge \bar{q} \wedge \bar{r} \wedge \bar{s}$$

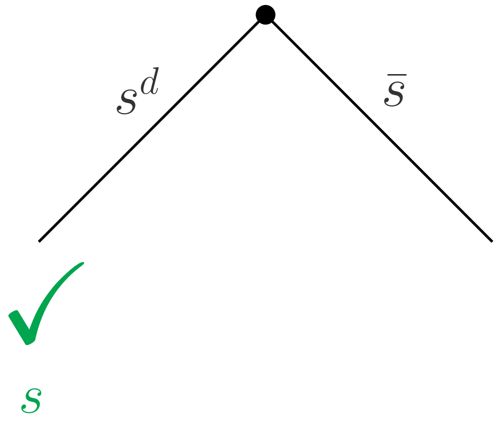
I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0



Example

$$F(X,Y) = p \vee q \vee r \vee s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$
$$\neg F(X,Y) = \bar{p} \wedge \bar{q} \wedge \bar{r} \wedge \bar{s}$$

I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \vee q \vee r$	$\bar{p} \wedge \bar{q} \wedge \bar{r}$	4

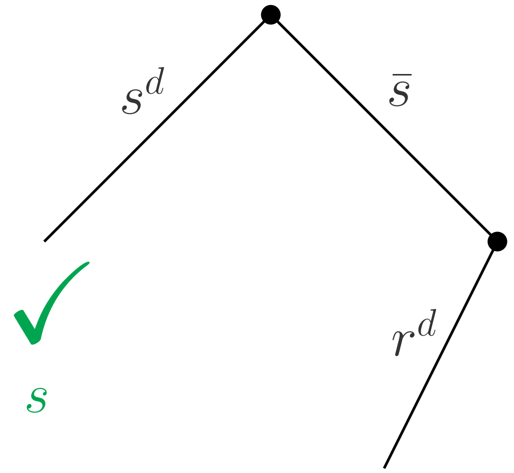


Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$

$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

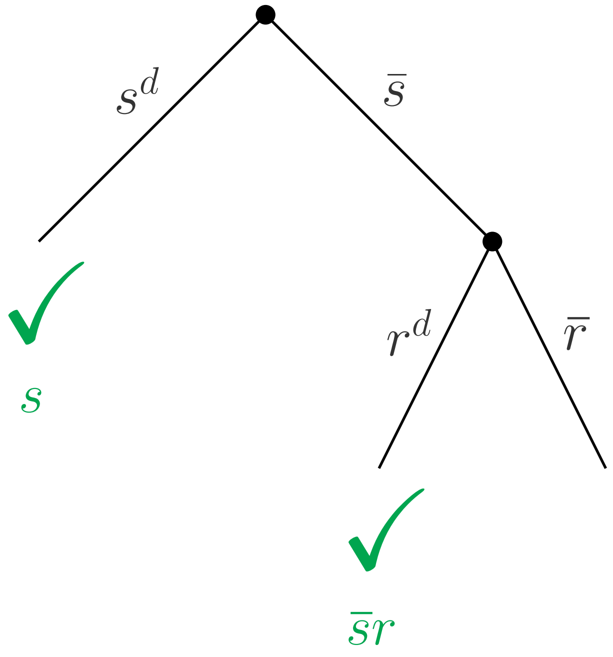
I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \lor q \lor r$	$\bar{p} \land \bar{q} \land \bar{r}$	4
$\bar{s}r^d$	$\top$	$\bot$	4



Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$
$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \lor q \lor r$	$\bar{p} \land \bar{q} \land \bar{r}$	4
$\bar{s}r^d$	$\top$	$\bot$	4
$\bar{s}\bar{r}$	$p \lor q$	$\bar{p} \land \bar{q}$	6

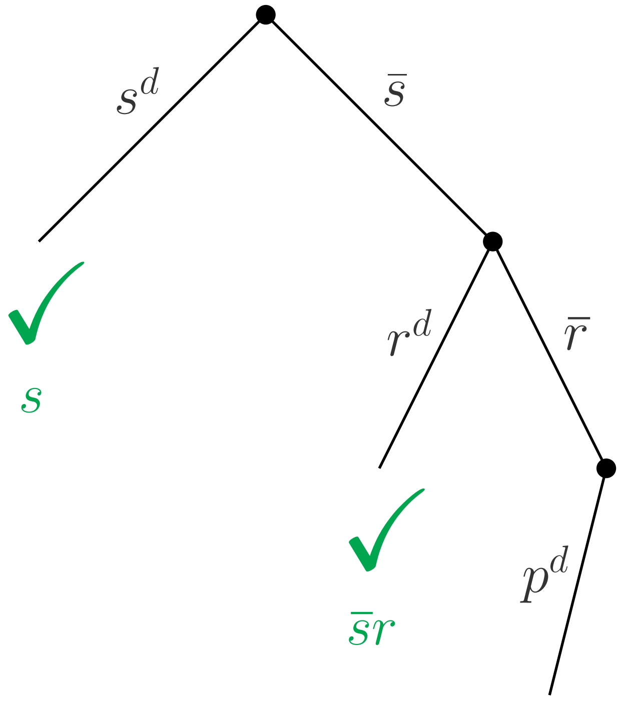


Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$

$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \lor q \lor r$	$\bar{p} \land \bar{q} \land \bar{r}$	4
$\bar{s}r^d$	$\top$	$\bot$	4
$\bar{s}\bar{r}$	$p \lor q$	$\bar{p} \land \bar{q}$	6
$\bar{s}\bar{r}p^d$	$\top$	$\bot$	6

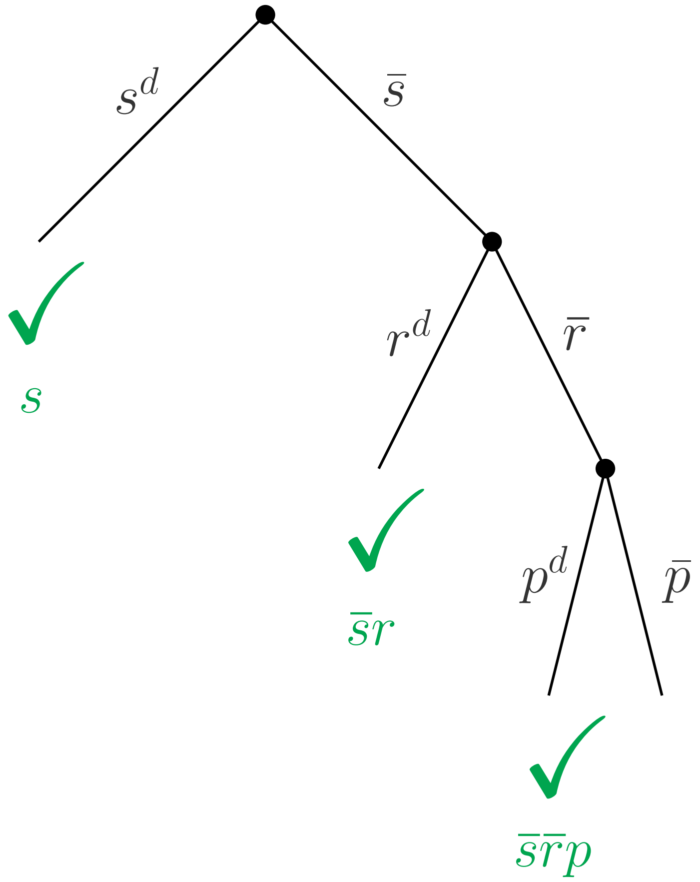


Example

$$F(X,Y) = p \lor q \lor r \lor s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$

$$\neg F(X,Y) = \bar{p} \land \bar{q} \land \bar{r} \land \bar{s}$$

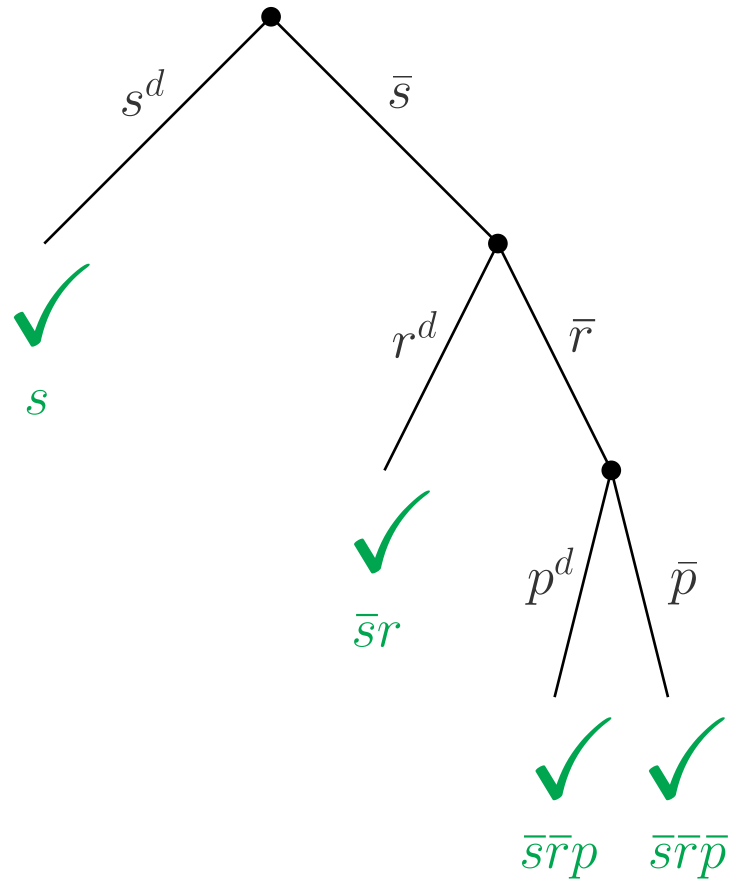
I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \lor q \lor r$	$\bar{p} \land \bar{q} \land \bar{r}$	4
$\bar{s}r^d$	$\top$	$\bot$	4
$\bar{s}\bar{r}$	$p \lor q$	$\bar{p} \land \bar{q}$	6
$\bar{s}\bar{r}p^d$	$\top$	$\bot$	6
$\bar{s}\bar{r}\bar{p}$	$\top$	$\bot$	7



Example

$$F(X,Y) = p \vee q \vee r \vee s \qquad X = \{p, r, s\} \qquad Y = \{q\}$$
$$\neg F(X,Y) = \bar{p} \wedge \bar{q} \wedge \bar{r} \wedge \bar{s}$$

I	$F\|_I$	$\neg F\|_I$	M
ε	F	$\neg F$	0
s^d	$\top$	$\bot$	0
$\bar{s}$	$p \vee q \vee r$	$\bar{p} \wedge \bar{q} \wedge \bar{r}$	4
$\bar{s}r^d$	$\top$	$\bot$	4
$\bar{s}\bar{r}$	$p \vee q$	$\bar{p} \wedge \bar{q}$	6
$\bar{s}\bar{r}p^d$	$\top$	$\bot$	6
$\bar{s}\bar{r}p$	$\top$	$\bot$	7
$\bar{s}\bar{r}\bar{p}$	$\top$	$\bot$	8



Our Contribution — the First Dual Calculus for Exact Projected Model Counting

EP0: $(P, N, I, M) \rightsquigarrow_{\mathsf{EP0}} M$ if $\emptyset \in P|_I$ and $\mathrm{decs}(I) = \emptyset$

EP1: $(P, N, I, M) \rightsquigarrow_{\mathsf{EP1}} M + 2^{|X-I|}$ if $P|_I = \emptyset$ and $\mathrm{var}(\mathrm{decs}(I)) \cap X = \emptyset$

EN0: $(P, N, I, M) \rightsquigarrow_{\mathsf{EN0}} M + 2^{|X-I|}$ if $\emptyset \in N|_I$ and $\mathrm{var}(\mathrm{decs}(I)) \cap X = \emptyset$

BP0F: $(P, N, I\ell^d I', M) \rightsquigarrow_{\mathsf{BP0F}} (P, N, I\overline{\ell}^{f(m')}, M)$ if $\emptyset \in P|_{I\ell I'}$ and $\mathrm{var}(\mathrm{decs}(I')) = \emptyset$ and
$m' = \sum \{m \mid \ell^{f(m)} \in I'\}$

JP0: $(P, N, II', M) \rightsquigarrow_{\mathsf{JP0}} (P \wedge C^r, N, I\ell', M - m')$ if $\emptyset \in P|_{II'}$ and $P \models C$ and $C|_I = \{\, \ell' \,\}$ and
$m' = \sum \{m \mid \ell^{f(m)} \in I'\}$

BP1F: $(P, N, I\ell^d I', M) \rightsquigarrow_{\mathsf{BP1F}} (P, N, I\overline{\ell}^{f(m'+m'')}, M + m'')$ if $P|_{I\ell I'} = \emptyset$ and $\mathrm{var}(\ell) \in X$ and
$\mathrm{var}(\mathrm{decs}(I')) \cap X = \emptyset$ and $m' = \sum \{m \mid \ell^{f(m)} \in I'\}$ and $m'' = 2^{|X-I\ell I'|}$

BP1L: $(P, N, I\ell^d I', M) \rightsquigarrow_{\mathsf{BP1L}} (P \wedge D, N, I\overline{\ell}, M + m'')$ if $P|_{I\ell I'} = \emptyset$ and $\mathrm{var}(\ell) \in X$ and
$\mathrm{var}(\mathrm{decs}(I')) \cap X = \emptyset$ and $m'' = 2^{|X-I\ell I'|}$ and $D = \pi(\neg\mathrm{decs}(I\ell), X)$

Our Contribution — the First Dual Calculus for Exact Projected Model Counting

BN0F: $(P, N, I\ell^d I', M) \rightsquigarrow_{\text{BN0F}} (P, N, I\overline{\ell}^{f(m'+m'')}, M + m'')$ if $\emptyset \in N|_{I\ell I'}$ and $\mathrm{var}(\ell) \in X$ and
$\mathrm{var}(\mathrm{decs}(I')) \cap X = \emptyset$ and $m' = \sum \{m \mid \ell^{f(m)} \in I'\}$ and $m'' = 2^{|X - I\ell I'|}$

BN0L: $(P, N, I\ell^d I', M) \rightsquigarrow_{\text{BN0L}} (P \wedge D, N, I\overline{\ell}, M + m'')$ if $\emptyset \in N|_{I\ell I'}$ and $\mathrm{var}(\ell) \in X$ and
$\mathrm{var}(\mathrm{decs}(I')) \cap X = \emptyset$ and $m'' = 2^{|X - I\ell I'|}$ and $D = \pi(\neg \mathrm{decs}(I\ell), X)$

DX: $(P, N, I, M) \rightsquigarrow_{\text{DX}} (P, N, I\ell^d, M)$ if $\emptyset \notin (P \wedge N)|_I$ and $\mathrm{units}((P \wedge N)|_I) = \emptyset$ and
$\mathrm{var}(\ell) \in X - I$

DYS: $(P, N, I, M) \rightsquigarrow_{\text{DYS}} (P, N, I\ell^d, M)$ if $\emptyset \notin (P \wedge N)|_I$ and $\mathrm{units}((P \wedge N)|_I) = \emptyset$ and
$\mathrm{var}(\ell) \in (Y \cup S) - I$ and $X - I = \emptyset$

UP: $(P, N, I, M) \rightsquigarrow_{\text{UP}} (P, N, I\ell, M)$ if $\{\ell\} \in P|_I$

UNXY: $(P, N, I, M) \rightsquigarrow_{\text{UNXY}} (P, N, I\overline{\ell}^d, M)$ if $\{\ell\} \in N|_I$ and $\mathrm{var}(\ell) \in X \cup Y$ and $\emptyset \notin P|_I$ and $\mathrm{units}(P|_I) = \emptyset$

UNT: $(P, N, I, M) \rightsquigarrow_{\text{UNT}} (P, N, I\ell, M)$ if $\{\ell\} \in N|_I$ and $\mathrm{var}(\ell) \in T$ and $\emptyset \notin P|_I$ and $\mathrm{units}(P|_I) = \emptyset$

FP: $(P \wedge C^r, N, I, M) \rightsquigarrow_{\text{FP}} (P, N, I, M)$ if $\emptyset \notin P|_I$

Our Dual Approach Facilitates the Detection of Partial Models

```
$ cat clause.form
p | q | r | s
$ dualiza -e -r p,r,s clause.form
ALL SATISFYING ASSIGNMENTS
s
r !s
!r !s
$ dualiza -r p,r,s clause.form
NUMBER SATISFYING ASSIGNMENTS
8
```

```
$ dualiza -r p,r,s clause.form -l | grep RULE
c LOG 1 RULE UNX 1  -4
c LOG 1 RULE UNX 2  -4
c LOG 1 RULE BNOF 1 -4
c LOG 2 RULE UNX 3  -3
c LOG 2 RULE BNOF 2 -3
c LOG 3 RULE UNY 1  -2
c LOG 3 RULE ENO 1
```

Can We Compete with State-of-the-Art #SAT Solvers?

```
$ cat clause4.form
(x1 | x2 | x3 | x4)
```

Can We Compete with State-of-the-Art #SAT Solvers?

```
$ cat clause4.form
(x1 | x2 | x3 | x4)
```

n	Mode	sharpSAT [s]	Dualiza [s]
10	dual	$< 1 \cdot 10^{-2}$	$< 1 \cdot 10^{-2}$
	block	$< 1 \cdot 10^{-2}$	$2 \cdot 10^{-2}$
	flip	$< 1 \cdot 10^{-2}$	$< 1 \cdot 10^{-2}$
20	block	$1 \cdot 10^{-2}$	$9 \cdot 10^{-1}$
	flip	$1 \cdot 10^{-2}$	$2 \cdot 10^{-1}$
30	block	$1 \cdot 10^{-2}$	$4 \cdot 10^{4}$
	flip	$1 \cdot 10^{-2}$	$2 \cdot 10^{2}$
100	dual	$< 1 \cdot 10^{-2}$	$< 1 \cdot 10^{-2}$
1000	dual	$8 \cdot 10^{-2}$	$2 \cdot 10^{-2}$
10000	dual	$1 \cdot 10^{1}$	$2 \cdot 10^{-1}$

Where Our Dual Approach Really Wins

```
$ cat nrp4.form
(x1 | x2 | x3 | x4) |
(x5 = x2 ^ x3 ^ x4) |
(x6 = x1 ^ x3 ^ x4) |
(x7 = x1 ^ x2 ^ x4) |
(x8 = x1 ^ x2 ^ x3)
```

Where Our Dual Approach Really Wins

```
$ cat nrp4.form
(x1 | x2 | x3 | x4) |
(x5 = x2 ^ x3 ^ x4) |
(x6 = x1 ^ x3 ^ x4) |
(x7 = x1 ^ x2 ^ x4) |
(x8 = x1 ^ x2 ^ x3)
```

n	Method	sharpSAT [s]	DUALIZA [s]
10	dual	$9 \cdot 10^{-2}$	$< 1 \cdot 10^{-2}$
20	dual	$7 \cdot 10^{2}$	$1 \cdot 10^{-2}$
21	dual	$2 \cdot 10^{3}$	$1 \cdot 10^{-2}$
22	dual	*	$1 \cdot 10^{-2}$
100	dual	*	$8 \cdot 10^{-2}$
1000	dual	*	$1 \cdot 10^{1}$
5000	dual	*	$2 \cdot 10^{2}$

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

The Main Idea

$F = (\bar{p} \vee q) \wedge (p \vee q)$

$V = \{p, q, r\}$

Rules

$M = (r \wedge q) \vee (\bar{r} \wedge p \wedge q) \vee (\bar{r} \wedge \bar{p} \wedge q) = C_1 \vee C_2 \vee C_3$

$M \equiv F$ and $\#M = \sum_{i=1}^{3} 2^{|V-C_i|} = 4 = \#F$

Generalizing, $\boxed{\#F = \sum_{C \in M} 2^{|V-C|}}$ and

M is a Disjoint-Sum-of-Products (DSOP) representation of F

- M is a disjunction of conjunctions of literals (cubes)
- The cubes in M are pairwise contradicting
- M is logically equivalent to F
- M is not unique

The Main Idea

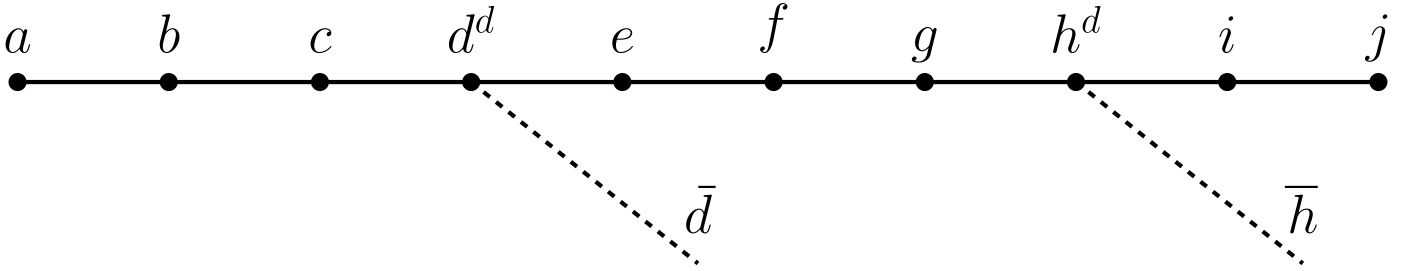
Assignment Trail I

$I = abcd^d efgh^d ij$

Pending Search Space $O(I)$

$O(I) = abc\bar{d} \vee abcdefg\bar{h} \vee I$

$O(I)$ is a DSOP



Pending Models of F $F \wedge O(I)$

Models of F found M

The Main Idea

During execution, we have that

$\boxed{O(I) \land F \lor M \equiv F}$ and $\boxed{\#F = \#(F \land O(I)) + \sum_{C \in M} 2^{|V-C|}}$

Upon termination, we have $O(I) = \bot$, hence

$\boxed{M \equiv F}$ and $\boxed{\#F = \sum_{C \in M} 2^{|V-C|}}$

Calculus

EndTrue: $(F, I, M, \delta) \rightsquigarrow_{\mathrm{EndTrue}} M \vee I$ if $F|_I = \top$ and $\mathrm{decs}(I) = \emptyset$

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

Unit: $(F, I, M, \delta) \rightsquigarrow_{\mathrm{Unit}} (F, I\ell, M, \delta[\ell \mapsto a])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a = \delta(C \setminus \{\ell\})$

BackTrue: $(F, I, M, \delta) \rightsquigarrow_{\mathrm{BackTrue}} (F, PK\ell, M \vee I, \delta[L \mapsto \infty][\ell \mapsto e])$ if $F|_I = \top$ and $PQ = I$ and $D = \overline{\mathrm{decs}(I)}$ and $e + 1 = \delta(D) = \delta(I)$ and $\ell \in D$ and $e = \delta(D \setminus \{\ell\}) = \delta(P)$ and $K = Q_{\leqslant e}$ and $L = Q_{>e}$

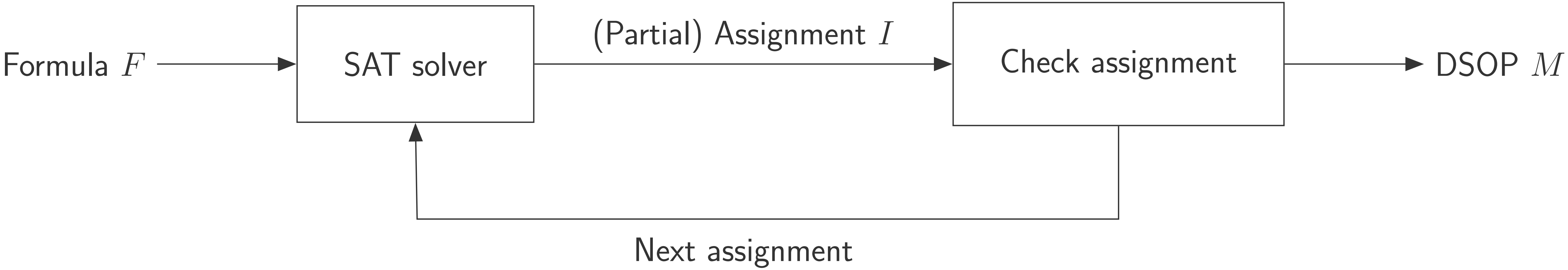
BackFalse: $(F, I, M, \delta) \rightsquigarrow_{\mathrm{BackFalse}} (F, PK\ell, M, \delta[L \mapsto \infty][\ell \mapsto j])$ if exists $C \in F$ and exists D with $PQ = I$ and $C|_I = \bot$ and $c = \delta(C) = \delta(D) > 0$ such that $\ell \in D$ and $\bar{\ell} \in \mathrm{decs}(I)$ and $\ell|_Q = \bot$ and $F \wedge \overline{M} \models D$ and $j = \delta(D \setminus \{\ell\})$ and $b = \delta(P) = c - 1$ and $K = Q_{\leqslant b}$ and $L = Q_{>b}$

Decide: $(F, I, M, \delta) \rightsquigarrow_{\mathrm{Decide}} (F, I\ell^d, M, \delta[\ell \mapsto d])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and $\mathrm{units}(F|_I) = \emptyset$ and $V(\ell) \in V$ and $\delta(\ell) = \infty$ and $d = \delta(I) + 1$

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

Main Idea



Formula F

SAT solver

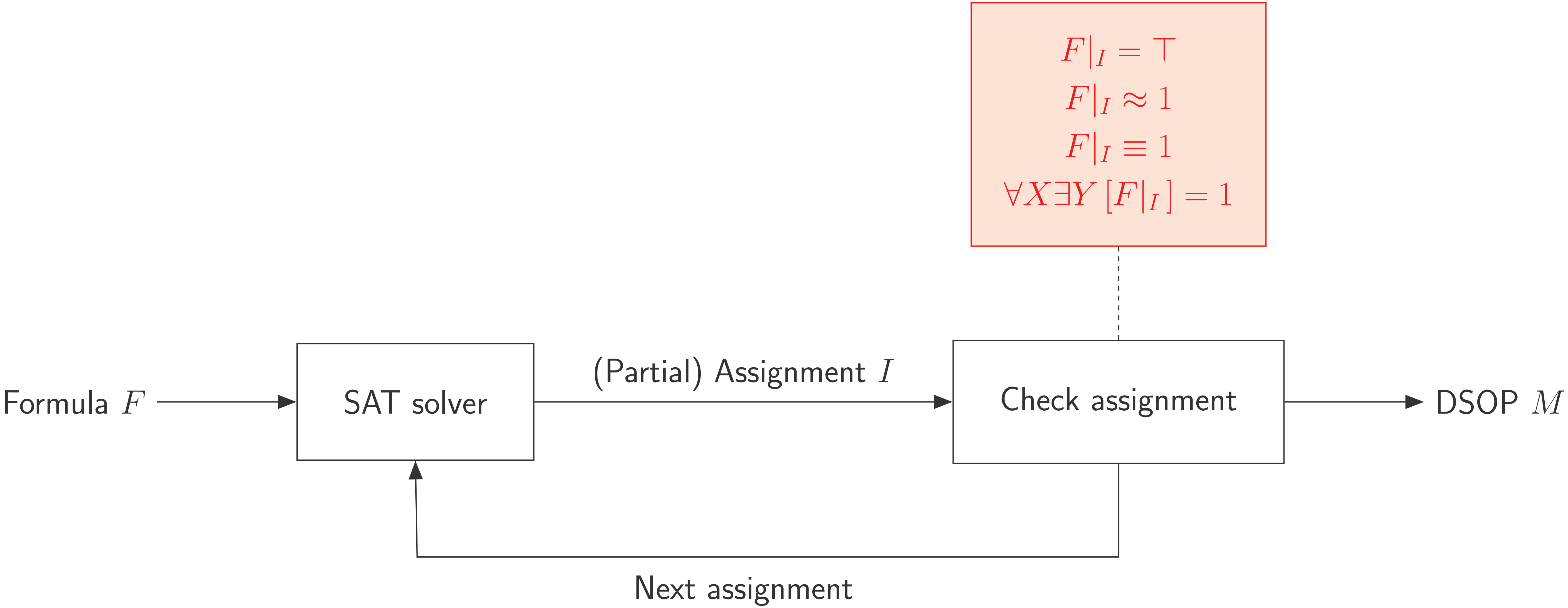
(Partial) Assignment I

Check assignment

DSOP M

Next assignment

Our Contribution



$F|_I = \top$

$F|_I \approx 1$

$F|_I \equiv 1$

$\forall X \exists Y \, [F|_I] = 1$

Formula F

SAT solver

(Partial) Assignment I

Check assignment

DSOP M

Next assignment

Logical Entailment Test under Projection

Given

F	formula over variables in $X \cup Y$
I	trail over variables in $X \cup Y$

Logical Entailment Test under Projection

Given

F	formula over variables in $X \cup Y$
I	trail over variables in $X \cup Y$

Quantified entailment condition

- In $\varphi = \forall X \forall Y \, [\, F|_I \,]$ the unassigned variables in $X \cup Y$ are quantified
- $\varphi = 1$: all possible total extensions of I satisfy F

Logical Entailment Test under Projection

Given

F formula over variables in $X \cup Y$
I trail over variables in $X \cup Y$

Quantified entailment condition

- In $\varphi = \forall X \forall Y \, [\, F|_I \,]$ the unassigned variables in $X \cup Y$ are quantified
- $\varphi = 1$: all possible total extensions of I satisfy F

Entailment under projection onto the set of variables X

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

Logical Entailment Test under Projection

Given

F formula over variables in $X \cup Y$

I trail over variables in $X \cup Y$

Quantified entailment condition

- In $\varphi = \forall X \forall Y \,[\, F|_I \,]$ the unassigned variables in $X \cup Y$ are quantified
- $\varphi = 1$: all possible total extensions of I satisfy F

Entailment under projection onto the set of variables X

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

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

Four Flavors of Logical Entailment under Projection

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

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

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

Four Flavors of Logical Entailment under Projection

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

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

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

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

$I = x_1 x_2 \bar{y}$: $\qquad \bot \in BCP(\overline{F}, I) \quad \Longrightarrow \quad x_1 x_2 \models F$

Four Flavors of Logical Entailment under Projection

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

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

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

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

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

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

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

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

Four Flavors of Logical Entailment under Projection

1) $F|_I = \top$ (*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 $N = \mathsf{CNF}(\overline{F})$:

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

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

$I = x_1 \bar{t_2} t_1 \bar{y_2}$: $N|_I = \top$

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

Algorithm

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

Output: DNF M consisting of models of F projected onto X

```
Enumerate( $F$ )
$I := \varepsilon$; $\delta := \infty$; $M := \bot$
forever do
  $C :=$ PropagateUnits( $F$, $I$, $\delta$ )
  if $C \neq \bot$ 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( $I$, $\delta$ )
```

Calculus

EndTrue: $(F,\, I,\, M,\, \delta) \rightsquigarrow_{\mathsf{EndTrue}} M \vee m$ if $V(\mathsf{decs}(I)) \cap X = \emptyset$ and $m \overset{\text{def}}{=} \pi(I, X)$ 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 $a \overset{\text{def}}{=} \delta(C \setminus \{\ell\})$

BackTrue: $(F,\, I,\, M,\, \delta) \rightsquigarrow_{\mathsf{BackTrue}} (F,\, UK\ell,\, M \vee m,\, \delta[L \mapsto \infty][\ell \mapsto b])$ if $UV \overset{\text{def}}{=} I$ and $D \overset{\text{def}}{=} \overline{\pi(\mathsf{decs}(I), X)}$ and $b + 1 \overset{\text{def}}{=} \delta(D) \leqslant \delta(I)$ and $\ell \in D$ and $b = \delta(D \setminus \{\ell\}) = \delta(U)$ and $m \overset{\text{def}}{=} \pi(I, X)$ and $K \overset{\text{def}}{=} V_{\leqslant b}$ and $L \overset{\text{def}}{=} V_{>b}$ 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 $UV \overset{\text{def}}{=} I$ and $C|_I = 0$ and $c \overset{\text{def}}{=} \delta(C) = \delta(D) > 0$ 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 $j \overset{\text{def}}{=} \delta(D \setminus \{\ell\})$ and $b \overset{\text{def}}{=} \delta(U) = c - 1$ and $K \overset{\text{def}}{=} V_{\leqslant b}$ and $L \overset{\text{def}}{=} V_{>b}$

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 $d \overset{\text{def}}{=} \delta(I) + 1$ 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 $d \overset{\text{def}}{=} \delta(I) + 1$ and $V(\ell) \in Y$ and $X - I = \emptyset$

Outline

- State of the Art in Exact Model Counting
- Challenges and Solutions
- **Solution 1: Dualizing Projected Model Counting**
- **Solution 2: Combining Conflict-Driven Clause Learning and Chronological Backtracking**
- **Solution 3: Exploiting Logical Entailment**
- Conclusion and Future Work

Conclusion

Our Contribution

First dual Calculus for exact projected model counting

- Search space pruning
- Good learning

Chronological CDCL for model counting

- Formal calculus and proof
- No exponential learning

Early Pruning

- Compute partial assignments entailing the formula on-the-fly
- Entailment tests in four flavors of different strength

Conclusion

Our Contribution

First dual Calculus for exact projected model counting

- Search space pruning
- Good learning

Chronological CDCL for model counting

- Formal calculus and proof
- No exponential learning

Early Pruning

- Compute partial assignments entailing the formula on-the-fly
- Entailment tests in four flavors of different strength

Future Work

- Implement and validate our method exploiting logical entailment
- Target weighted model integration and model counting with or without projection
- Investigate methods concerning the implementation of QBF oracles for exploiting logical entailment
- Combine with decomposition-based approaches and generate d-DNNF

Chronological CDCL[11]

---

[11] S. Möhle, A. Biere, “Backing Backtracking”, SAT’19.

CDCL Invariants

Trail: The assignment trail contains neither complementary pairs of literals nor duplicates.

ConflictLower: The assignment trail preceding the current decision level does not falsify the formula.

Propagation: On every decision level preceding the current decision level all unit clauses are propagated until completion.

LevelOrder: The literals are ordered on the assignment trail in ascending order with respect to their decision level.

ConflictingClause: At decision levels greater than zero the conflicting clause contains at least two literals with the current decision level.

CDCL Invariants

Trail:	The assignment trail contains neither complementary pairs of literals nor duplicates.
ConflictLower:	The assignment trail preceding the current decision level does not falsify the formula.
~~Propagation:~~	~~On every decision level preceding the current decision level all unit clauses are propagated until completion.~~
~~LevelOrder:~~	~~The literals are ordered on the assignment trail in ascending order with respect to their decision level.~~
~~ConflictingClause:~~	~~At decision levels greater than zero the conflicting clause contains at least two literals with the current decision level.~~

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	9	38	-23	17	44	-16
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I	$\cdots$	4	**5**	30	47	15	18	**6**	-7	-8	45	**9**	38	-23	17	44	-16
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6

decision literal

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I	$\cdots$	4	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	9	38	-23	17	44	-16
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6

$$\text{block}(I, 4)$$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I	$\cdots$	4	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	9	38	-23	17	44	-16
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6

slice$(I, 4)$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
I	$\cdots$	**4**	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	9	38	-23	17	44	-16
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6

$$I_{\leqslant 4}$$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	9	38	-23	17	44	-16	✗
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6	

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	9	38	-23	17	44	-16	✗
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6	

conflicting { -47, -17, -44 }

conflict level 6

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	9	38	-23	17	44	-16	✗
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6	

conflicting { -47, -17, -44 }

learned { -30, -47, -18, 23 }

jump level 4

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	9	38	-23	17	44	-16	✗
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	6	6	6	6	6	6	
conflicting		{			-47,										-17,	-44	}	
learned		{		-30,	-47,		-18,							23			}	

backtrack level 5

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45	**23**
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	**4**

out of order

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14
I	$\cdots$	4	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	23
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	4

block$(I, 4)$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14
I	$\cdots$	4	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	**23**
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	4

slice$(I, 4)$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13
I	$\cdots$	4	5	30	47	15	18	6	-7	-8	45
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14
I	$\cdots$	**4**	**5**	**30**	**47**	**15**	**18**	6	-7	-8	45	**23**
δ	$\cdots$	3	4	4	4	4	4	5	5	5	5	4

$I_{\leqslant 4}$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
I	$\cdots$	18	6	-7	-8	45	23	-38	-9	10	-11	13	16	-17	-25	42	12	-41
δ	$\cdots$	4	5	5	5	5	4	4	5	6	7	5	4	4	4	4	5	5

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
I	$\cdots$	18	6	-7	-8	45	23	-38	-9	10	-11	13	16	-17	-25	42	12	-41
δ	$\cdots$	4	5	5	5	5	4	4	5	6	7	5	4	4	4	4	5	5
																		✗

conflicting $\{$ 17, -42, -12 $\}$

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
I	$\cdots$	18	6	-7	-8	45	23	-38	-9	10	-11	13	16	-17	-25	42	12	-41
δ	$\cdots$	4	5	5	5	5	4	4	5	6	7	5	4	4	4	4	5	5
																		✗

conflicting { 17, -42, -12 }

backtrack level 4

Combining CDCL with Chronological Backtracking

τ	$\cdots$	4	5	6	7	8	9	10	11
I	$\cdots$	18	23	-38	16	-17	-25	42	-12
δ	$\cdots$	4	4	4	4	4	4	4	4

Calculus

True: $(F,\ I,\ \delta) \rightsquigarrow_{\text{True}}$ SAT if $F|_I = \top$

False: $(F,\ I,\ \delta) \rightsquigarrow_{\text{False}}$ UNSAT if exists $C \in F$ with $C|_I = \bot$ and $\delta(C) = 0$

Unit: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Unit}} (F,\ I\ell,\ \delta[\ell \mapsto a])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a = \delta(C \setminus \{\ell\})$

Jump: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Jump}} (F \wedge D,\ PK\ell,\ \delta[L \mapsto \infty][\ell \mapsto j])$ if exists $C \in F$ with $PQ = I$ and $C|_I = \bot$ such that $c = \delta(C) = \delta(D) > 0$ and $\ell \in D$ and $\ell|_Q = \bot$ and $F \models D$ and $j = \delta(D \setminus \{\ell\})$ and $b = \delta(P)$ and $\boxed{j \leqslant b < c}$ and $K = Q_{\leqslant b}$ and $L = Q_{>b}$

Decide: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Decide}} (F,\ I\ell,\ \delta[\ell \mapsto d])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and units$(F|_I) = \emptyset$ and $V(\ell) \in V$ and $\delta(\ell) = \infty$ and $d = \delta(I) + 1$

Calculus

True: $(F,\ I,\ \delta) \rightsquigarrow_{\text{True}} \text{SAT}$ if $F|_I = \top$

False: $(F,\ I,\ \delta) \rightsquigarrow_{\text{False}} \text{UNSAT}$ if exists $C \in F$ with $C|_I = \bot$ and $\delta(C) = 0$

Unit: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Unit}} (F,\ I\ell,\ \delta[\ell \mapsto a])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and exists $C \in F$ with $\{\ell\} = C|_I$ and $a = \delta(C \setminus \{\ell\})$

Jump: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Jump}} (F \wedge D,\ PK\ell,\ \delta[L \mapsto \infty][\ell \mapsto j])$ if exists $C \in F$ with $PQ = I$ and $C|_I = \bot$ such that $c = \delta(C) = \delta(D) > 0$ and $\ell \in D$ and $\ell|_Q = \bot$ and $F \models D$ and $j = \delta(D \setminus \{\ell\})$ and $b = \delta(P)$ and $\boxed{b = c - 1}$ and $K = Q_{\leqslant b}$ and $L = Q_{>b}$

Decide: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Decide}} (F,\ I\ell,\ \delta[\ell \mapsto d])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and units$(F|_I) = \emptyset$ and $V(\ell) \in V$ and $\delta(\ell) = \infty$ and $d = \delta(I) + 1$

Calculus

True: $(F,\ I,\ \delta) \rightsquigarrow_{\text{True}} \text{SAT}$ if $F|_I = \top$

False: $(F,\ I,\ \delta) \rightsquigarrow_{\text{False}} \text{UNSAT}$ if exists $C \in F$ with $C|_I = \bot$ and $\delta(C) = 0$

Unit: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Unit}} (F,\ I\ell,\ \delta[\ell \mapsto a])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and
exists $C \in F$ with $\{\ell\} = C|_I$ and $a = \delta(C \setminus \{\ell\})$

Jump: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Jump}} (F \wedge D,\ PK\ell,\ \delta[L \mapsto \infty][\ell \mapsto j])$ if exists $C \in F$ with
$PQ = I$ and $C|_I = \bot$ such that $c = \delta(C) = \delta(D) > 0$ and $\ell \in D$ and
$\ell|_Q = \bot$ and $F \models D$ and $j = \delta(D \setminus \{\ell\})$ and $b = \delta(P)$ and
$\boxed{b = j}$ and $K = Q_{\leqslant b}$ and $L = Q_{>b}$

Decide: $(F,\ I,\ \delta) \rightsquigarrow_{\text{Decide}} (F,\ I\ell,\ \delta[\ell \mapsto d])$ if $F|_I \neq \top$ and $\bot \notin F|_I$ and
$\text{units}(F|_I) = \emptyset$ and $V(\ell) \in V$ and $\delta(\ell) = \infty$ and $d = \delta(I) + 1$

Invariants

Trail: The assignment trail contains neither complementary pairs of literals nor duplicates.

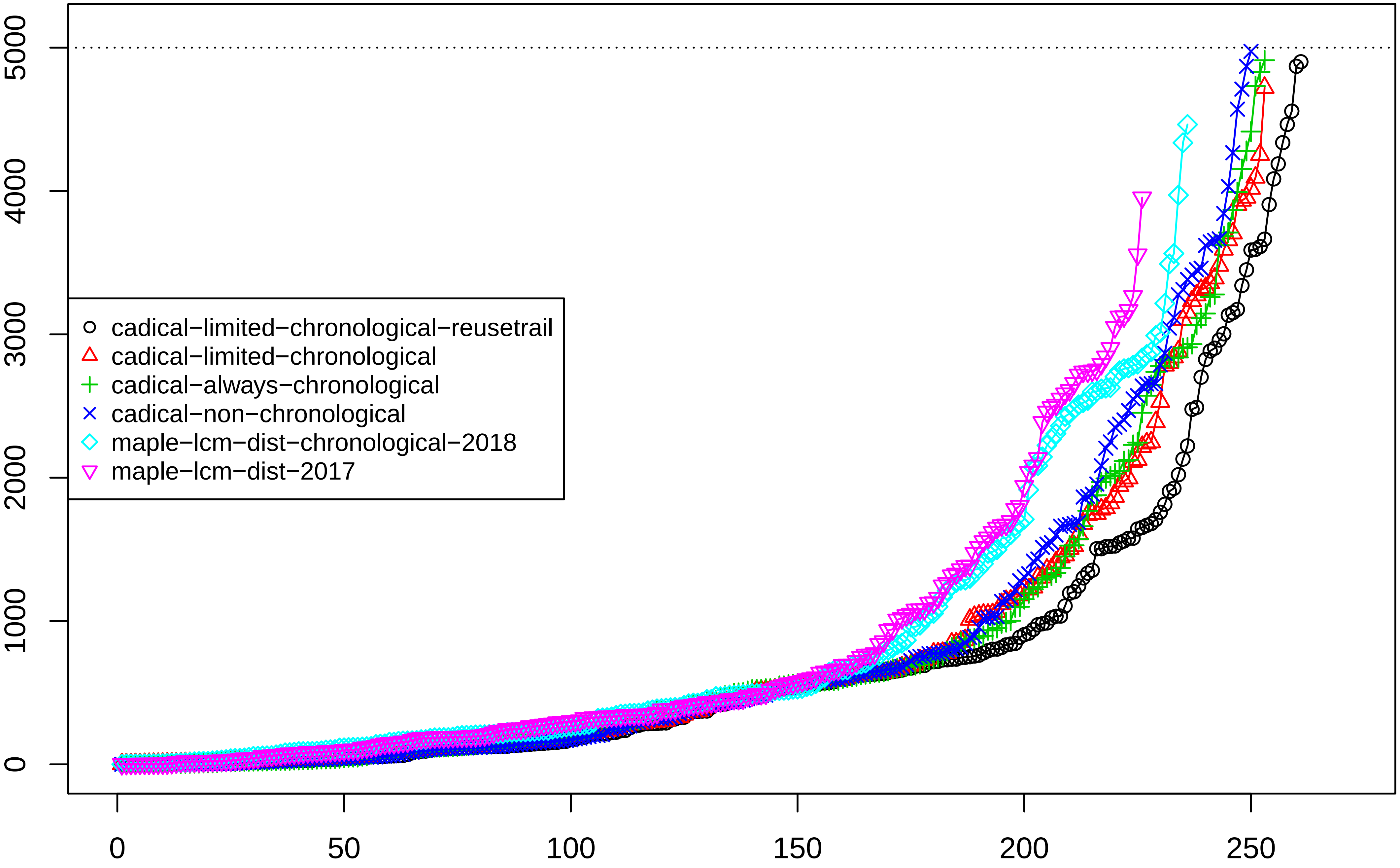
ConflictLower: The assignment trail preceding the current decision level does not falsify the formula.

(1): $\forall k, \ell \in \mathsf{decs}(I) \,.\, \tau(I, k) < \tau(I, \ell) \implies \delta(k) < \delta(\ell)$

(2): $\delta(\mathsf{decs}(I)) = \{1, \ldots, \delta(I)\}$

(3): $\forall n \in \mathbb{N} \,.\, F \wedge \mathsf{decs}_{\leqslant n}(I) \models I_{\leqslant n}$

Experiments — Main Track of SAT Competition 2018



Experiments

solver configurations	solved instances		
	total	SAT	UNSAT
cadical-limited-chronological-reusetrail	261	155	106
cadical-limited-chronological	253	147	106
cadical-always-chronological	253	148	105
cadical-non-chronological	250	144	106
maple-lcm-dist-chronological-2018	236	134	102
maple-lcm-dist-2017	226	126	100