Dualizing Projected Model Counting

Sibylle Möhle and Armin Biere

Institute for Formal Models and Verification
Johannes Kepler University Linz

30th International Conference on Tools with Artificial Intelligence
November 7, 2018

x	y	$F(x, y)$
0	0	0
0	1	1
1	0	1
1	1	1

x	y	$F(x, y)$
0	0	0
0	1	1
1	0	1
1	1	1

$$\mathcal{M}(F(x, y)) = \{\neg xy, x\neg y, xy\}$$

x	y	$F(x, y)$	$\exists y.F(x, y)$

<table>
<tr><th>x</th><th>y</th><th>$F(x, y)$</th><th>$\exists y . F(x, y)$</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td rowspan="2">1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td rowspan="2">1</td></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
</table>

x	y	$F(x,y)$	$\exists y.F(x,y)$
0	0	0	1
0	1	1	
1	0	1	1
1	1	1	

$$\mathcal{M}(\exists y.F(x,y)) = \{\neg x, x\}$$

Projected Model Counting Generalized

$F(X, Y)$ (arbitrary) propositional formula over variables X and Y with $X \cap Y = \emptyset$

X *relevant* input variables
Y *irrelevant* input variables

Projected Model Counting Generalized

$F(X, Y)$ (arbitrary) propositional formula over variables X and Y with $X \cap Y = \emptyset$

X *relevant* input variables
Y *irrelevant* input variables

We are interested in the number of models projected onto X:

$$\#\exists Y.F(X, Y)$$

Projected Model Counting Generalized

$F(X, Y)$ (arbitrary) propositional formula over variables X and Y with $X \cap Y = \emptyset$

X *relevant* input variables
Y *irrelevant* input variables

We are interested in the number of models projected onto X:

$$\#\exists Y.F(X, Y)$$

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

$X = \{x\}$ $Y = \{y\}$ $\mathcal{M}(\exists Y.F(X, Y)) = \{x, \neg x\}$ $\#\exists Y.F(X, Y) = 2$

Projected Model Counting Generalized

$F(X, Y)$ (arbitrary) propositional formula over variables X and Y with $X \cap Y = \emptyset$

X *relevant* input variables
Y *irrelevant* input variables

We are interested in the number of models projected onto X:

$$\#\exists Y.F(X, Y)$$

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

$X = \{x\}$	$Y = \{y\}$	$\mathcal{M}(\exists Y.F(X, Y)) = \{x, \neg x\}$	$\#\exists Y.F(X, Y) = 2$
$X = \{x, y\}$	$Y = \emptyset$	$\mathcal{M}(\exists Y.F(X, Y)) = \{xy, x\neg y, \neg xy\}$	$\#\exists Y.F(X, Y) = 3 = \#F(X, Y)$

Particular Challenges in #SAT

Particular Challenges in #SAT

- The search space needs to be traversed exhaustively.

Particular Challenges in #SAT

- The search space needs to be traversed exhaustively.
- CDCL is biased towards conflicts.

Particular Challenges in #SAT

- The search space needs to be traversed exhaustively.
- CDCL is biased towards conflicts.
- Blocking clauses blow up the formula.

Particular Challenges in #SAT

- The search space needs to be traversed exhaustively.
- CDCL is biased towards conflicts.
- Blocking clauses blow up the formula.
- The addition of large clauses slows down the solver.

Particular Challenges in #SAT

- The search space needs to be traversed exhaustively.
- CDCL is biased towards conflicts.
- Blocking clauses blow up the formula.
- The addition of large clauses slows down the solver.
- Satisfiability checks and clause watching mechanisms are expensive.

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
```

Dual Representation of $F(X, Y)$

$$\boxed{P(X, Y)} \qquad\qquad \boxed{N(X, Y)}$$

$$\equiv \qquad\qquad \equiv$$

$$F(X, Y) \qquad\qquad \neg F(X, Y)$$

Dual Representation of $F(X, Y)$

$$\boxed{\exists S. P(X, Y, S)} \qquad\qquad \boxed{\exists T. N(X, Y, T)}$$

$$\equiv \qquad\qquad \equiv$$

$$F(X, Y) \qquad\qquad \neg F(X, Y)$$

The General Case — Duality with Projection onto Relevant Input Variables

$$\boxed{\exists Y, S. P(X, Y, S)} \qquad \boxed{\exists Y, T. N(X, Y, T)}$$

$$\equiv \qquad \equiv$$

$$\exists Y. F(X, Y) \qquad \exists Y. \neg F(X, Y)$$

A First Example

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

$P(X, Y, S) = (p \vee q \vee r \vee s)$ $S = \emptyset$
$N(X, Y, T) = (\neg p) \wedge (\neg q) \wedge (\neg r) \wedge (\neg s)$ $T = \emptyset$

A First Example

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

$P(X, Y, S) = (p \vee q \vee r \vee s)$ $\qquad S = \emptyset$

$N(X, Y, T) = (\neg p) \wedge (\neg q) \wedge (\neg r) \wedge (\neg s)$ $\qquad T = \emptyset$

Step	Rule	I	$P\|_I$	$N\|_I$	M	Found
0		$()$	$(p \vee q \vee r \vee s)$	$(\neg p) \wedge (\neg q) \wedge (\neg r) \wedge (\neg s)$	0	
1	UNXY	s	$\emptyset$	$(\neg p) \wedge (\neg q) \wedge (\neg r) \wedge ()$	0	
2	BN0F	$\neg s$	$(p \vee q \vee r)$	$(\neg p) \wedge (\neg q) \wedge (\neg r)$	4	s
3	UNXY	$\neg s r$	$\emptyset$	$(\neg p) \wedge (\neg q) \wedge ()$	4	
4	BN0F	$\neg s \neg r$	$(p \vee q)$	$(\neg p) \wedge (\neg q)$	6	$\neg s r$
5	UNXY	$\neg s \neg r q$	$\emptyset$	$(\neg p) \wedge ()$	6	
6	EN0	$\neg s \neg r q$	$\emptyset$	$(\neg p) \wedge ()$	8	$\neg s \neg r$

With the Non-Dual Approach Only Total Models Are Detected

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

$P(X, Y, S) = (p \vee q \vee r \vee s)$ $\qquad S = \emptyset$

With the Non-Dual Approach Only Total Models Are Detected

$F(X, Y) = (p \vee q \vee r \vee s)$ $\quad X = \{p, r, s\}$ $\quad Y = \{q\}$

$P(X, Y, S) = (p \vee q \vee r \vee s)$ $\quad S = \emptyset$

Step	Rule	I	$P\|_I$	M	Found
0		$()$	$(p \vee q \vee r \vee s)$	0	
1	DX	s	$\emptyset$	0	
2	DX	sr	$\emptyset$	0	
3	DX	srp	$\emptyset$	0	
4	DYS	$srpq$	$\emptyset$	0	
5	BP1F	$sr\neg p$	$\emptyset$	1	srp
6	DYS	$sr\neg pq$	$\emptyset$	1	
7	BP1F	$s\neg r$	$\emptyset$	2	$sr\overline{p}$
8	DX	$s\neg rp$	$\emptyset$	2	
9	DYS	$s\neg rpq$	$\emptyset$	2	
10	BP1F	$s\neg r\neg p$	$\emptyset$	3	$s\neg rp$
11	DYS	$s\neg r\neg pq$	$\emptyset$	3	
12	BP1F	$\neg s$	$(p \vee q \vee r)$	4	$s\neg r\neg p$
$\vdots$					

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	*	$2 \cdot 10^{2}$

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

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

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

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

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

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

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

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

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

BN0F: $(P, N, I\ell^d I', M) \rightsquigarrow_{\text{BN0F}} (P, N, I\bar{\ell}^{f(m'+m'')}, M + m'')$ 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: $(P, N, I\ell^d I', M) \rightsquigarrow_{\text{BN0L}} (P \wedge D, N, I\bar{\ell}, M + m'')$ 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: $(P, N, I, M) \rightsquigarrow_{\text{UNXY}} (P, N, I\bar{\ell}^d, M)$ 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 Contribution — the First Dual Calculus for Exact Projected Model Counting

- dual representation of the formula enabling the detection of partial models and subsequent pruning of the search space
- good learning mechanism exempt from satisfiability checks and clause watching mechanisms
- significant performance gain compared to non-dual variant

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

- dual representation of the formula enabling the detection of partial models and subsequent pruning of the search space
- good learning mechanism exempt from satisfiability checks and clause watching mechanisms
- significant performance gain compared to non-dual variant
- accepts arbitrary formulae and circuits as argument

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

- dual representation of the formula enabling the detection of partial models and subsequent pruning of the search space
- good learning mechanism exempt from satisfiability checks and clause watching mechanisms
- significant performance gain compared to non-dual variant
- accepts arbitrary formulae and circuits as argument
- novel techniques for preventing multiple model counts: *flipping* and *discounting*
- models state-of-the-art techniques: conflict analysis and conflict-driven backjumping

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

- dual representation of the formula enabling the detection of partial models and subsequent pruning of the search space
- good learning mechanism exempt from satisfiability checks and clause watching mechanisms
- significant performance gain compared to non-dual variant
- accepts arbitrary formulae and circuits as argument
- novel techniques for preventing multiple model counts: *flipping* and *discounting*
- models state-of-the-art techniques: conflict analysis and conflict-driven backjumping
- robust and carefully tested implementation: DUALIZA

Conclusion and Future Work

We are on the right track

- DUALIZA is competitive on some CNF formulae and
- outperforms state-of-the-art #SAT solvers on another class of formulae.

Conclusion and Future Work

We are on the right track

- DUALIZA is competitive on some CNF formulae and
- outperforms state-of-the-art #SAT solvers on another class of formulae.

In the near future, we plan to

- incorporate dual conflict analysis with backjumping and redundant clause learning,
- drop decision restrictions,
- capture component reasoning and
- weighted projected model counting for Bayesian reasoning,
- optimize circuit representation to improve CNF encoding, and
- explore dual preprocessing techniques.