DRAT Proofs for XOR Reasoning

Tobias Philipp, Adrián Rebola-Pardo

TU Dresden, TU Wien

JELIA 2016
Larnaca, Cyprus
November 9th, 2016

Funded by FWF project W1255-N23

XOR reasoning for SAT solving

$$x \oplus y \oplus z \oplus t = 1$$

XOR reasoning for SAT solving

$$x \oplus y \oplus z \oplus t = 1$$

$x \vee y \vee z \vee t$ $\quad$ $\overline{x} \vee \overline{y} \vee z \vee t$ $\quad$ $\overline{x} \vee y \vee \overline{z} \vee t$

$\overline{x} \vee y \vee z \vee \overline{t}$ $\quad$ $x \vee \overline{y} \vee \overline{z} \vee t$

$x \vee \overline{y} \vee z \vee \overline{t}$ $\quad$ $x \vee y \vee \overline{z} \vee \overline{t}$ $\quad$ $\overline{x} \vee \overline{y} \vee \overline{z} \vee \overline{t}$

XOR reasoning for SAT solving

$$x \oplus y \oplus z \oplus t = 1$$

$$x \vee y \vee z \vee t \qquad \overline{x} \vee \overline{y} \vee z \vee t \qquad \overline{x} \vee y \vee \overline{z} \vee t$$

$$\overline{x} \vee y \vee z \vee \overline{t} \qquad x \vee \overline{y} \vee \overline{z} \vee t$$

$$x \vee \overline{y} \vee z \vee \overline{t} \qquad x \vee y \vee \overline{z} \vee \overline{t} \qquad \overline{x} \vee \overline{y} \vee \overline{z} \vee \overline{t}$$

Problem 3 variables must be assigned before unit propagation

XOR reasoning for SAT solving

$$x \oplus y \oplus z \oplus t = 1$$

$x \lor y \lor z \lor t$ $\quad$ $\overline{x} \lor \overline{y} \lor z \lor t$ $\quad$ $\overline{x} \lor y \lor \overline{z} \lor t$

$\overline{x} \lor y \lor z \lor \overline{t}$ $\quad$ $x \lor \overline{y} \lor \overline{z} \lor t$

$x \lor \overline{y} \lor z \lor \overline{t}$ $\quad$ $x \lor y \lor \overline{z} \lor \overline{t}$ $\quad$ $\overline{x} \lor \overline{y} \lor \overline{z} \lor \overline{t}$

Problem 3 variables must be assigned before unit propagation

CDCL is **not polynomially bounded** in the presence of encoded XOR constraints.
Urquhart (1987), Beame et al. (2004)

XOR constraints naturally occur in **cryptography**.
Massacci et al. (2000)

Polynomial procedures for XOR reasoning can be **integrated** in SAT solvers.
Soos et al. (2009), Laitinen et al. (2014)

Unsatisfiability proofs for SAT solving

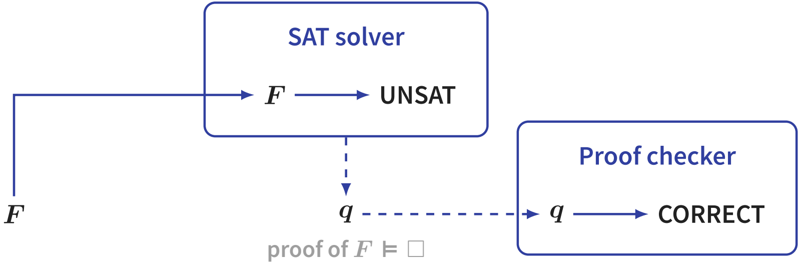
SAT solvers' architectures are complex, and bugs are hard to detect.

- false positives   partial interpretations as witnesses
- false negatives   unsatisfiability proofs are required

Unless $P = coNP$, validating unsatisfiability results is intractable.

The DRAT proof standard provides certificates for most techniques.
Heule et al. (2013, 2015), Philipp et al. (2014)

Unsatisfiability proofs for SAT solving



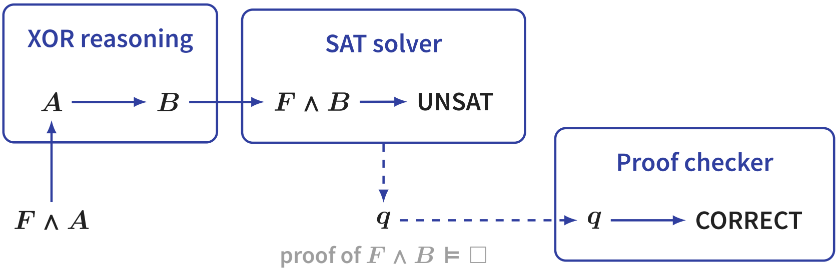
SAT solvers' architectures are complex, and bugs are hard to detect.

- false positives partial interpretations as witnesses
- false negatives unsatisfiability proofs are required

Unless $P = coNP$, validating unsatisfiability results is intractable.

The DRAT proof standard provides certificates for most techniques.
Heule et al. (2013, 2015), Philipp et al. (2014)

Unsatisfiability proofs for SAT solving



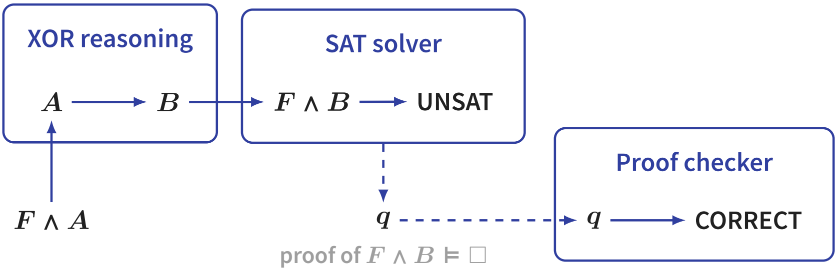
SAT solvers' architectures are complex, and bugs are hard to detect.

- false positives — partial interpretations as witnesses
- false negatives — unsatisfiability proofs are required

Unless $P = coNP$, validating unsatisfiability results is intractable.

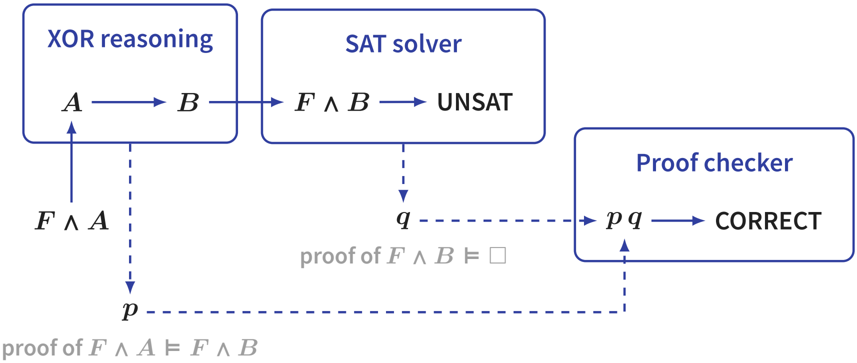
The DRAT proof standard provides certificates for most techniques.
Heule et al. (2013, 2015), Philipp et al. (2014)

Unsatisfiability proofs for SAT solving



Problem generating unsatisfiability proofs for XOR reasoning techniques.
Biere et al. (2006, 2015)

XOR reasoning is currently **disabled** when unsatisfiability proofs are required.



Problem generating unsatisfiability proofs for XOR reasoning techniques.

Biere et al. (2006, 2015)

XOR reasoning is currently **disabled** when unsatisfiability proofs are required.

The DRAT proof system

The DRAT proof system

Reverse unit propagation (RUP) in F

Blocked clauses

The DRAT proof system

Reverse unit propagation (RUP) in F

$$
\dfrac{\dfrac{\dfrac{C_0 \quad \in F}{D_1}\,\text{SUB} \qquad C_1 \quad \in F}{\ddots}\,\text{RES} \quad\; \begin{matrix} \\ D_{n-1} \qquad C_{n-1} \quad \in F \end{matrix}}{D_n}\,\text{RES} \qquad \text{RUP in } F
$$

Blocked clauses

The DRAT proof system

Reverse unit propagation (RUP) in F

$$
\begin{array}{l}
\dfrac{C_0 \quad \in F}{D_1} \text{ SUB} \\[1em]
\dfrac{D_1 \qquad C_1 \quad \in F}{\ddots} \text{ RES} \\[1em]
\dfrac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n \qquad \text{RUP in } F} \text{ RES}
\end{array}
$$

Blocked clauses

$$C \vee x$$

The DRAT proof system

Reverse unit propagation (RUP) in F

$$\dfrac{\dfrac{\dfrac{C_0 \quad \in F}{D_1}\,\text{SUB} \qquad C_1 \quad \in F}{\ddots}\,\text{RES}}{}$$

$$\dfrac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n}\,\text{RES} \qquad \text{RUP in } F$$

Blocked clauses

for all in F

$C \vee x \qquad D \vee \overline{x}$

The DRAT proof system

Reverse unit propagation (RUP) in F

$$\mathsf{SUB}\,\frac{C_0 \quad \in F}{D_1} \qquad D_1 \quad C_1 \quad \in F$$

$$\mathsf{RES}\,\frac{D_1 \quad C_1}{\ddots}$$

$$\mathsf{RES}\,\frac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n \qquad \text{RUP in } F}$$

Blocked clauses

for all in F

$$\mathsf{RES}\,\frac{C \lor x \qquad D \lor \overline{x}}{C \lor D}$$

tautology

The DRAT proof system

Reverse unit propagation (RUP) in F

$$\dfrac{\dfrac{\dfrac{C_0 \quad \in F}{D_1}\,\text{SUB} \qquad C_1 \quad \in F}{\ddots}\,\text{RES}}{}$$

$$\dfrac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n}\,\text{RES} \qquad \text{RUP in } F$$

Blocked clauses

blocked upon x for all in F

$$\dfrac{C \vee x \qquad D \vee \overline{x}}{C \vee D}\,\text{RES}$$

tautology

The DRAT proof system

Reverse unit propagation (RUP) in F

$$
\dfrac{\dfrac{\dfrac{C_0}{D_1}\,\text{SUB} \qquad C_1}{\ddots}\,\text{RES} \qquad \in F}{}
$$

$$C_0 \quad \in F$$
$$\text{SUB}\ \dfrac{C_0}{D_1} \qquad C_1 \quad \in F$$
$$\text{RES}\ \dfrac{D_1 \qquad C_1}{\ddots}$$
$$\text{RES}\ \dfrac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n} \qquad \text{RUP in } F$$

Blocked clauses

RAT upon x

$\underline{\cup}$

blocked upon x

for all in F

$$\text{RES}\ \dfrac{C \lor x \qquad D \lor \overline{x}}{C \lor D}$$

tautology

The DRAT proof system

Reverse unit propagation (RUP) in F

$C_0 \in F$

SUB

$D_1 \quad C_1 \in F$

RES

$\ddots$

$D_{n-1} \quad C_{n-1} \in F$

RES

D_n RUP in F

Blocked clauses

RAT upon x

$\cup$

blocked upon x

for all in F

$$\frac{C \lor x \qquad D \lor \overline{x}}{C \lor D}\ \text{RES}$$

tautology

Delete Resolution Asymmetric Tautology (DRAT) proof system

$$F \Rightarrow_{\text{DRAT}} G$$

The DRAT proof system

Reverse unit propagation (RUP) in F

$$\cfrac{\cfrac{\cfrac{C_0 \quad \in F}{D_1}\,\text{SUB} \qquad C_1 \quad \in F}{\ddots}\,\text{RES} \qquad \begin{array}{c} D_{n-1} \qquad C_{n-1} \quad \in F \end{array}}{D_n}\,\text{RES} \qquad \text{RUP in } F$$

Blocked clauses

RAT upon x

$\cup$I

blocked upon x

$$\cfrac{C \vee x \qquad D \vee \overline{x} \quad \text{for all in } F}{C \vee D \quad \text{tautology}}\,\text{RES}$$

Delete Resolution Asymmetric Tautology (DRAT) proof system

$$F \Rightarrow_{\text{DRAT}} G \quad \begin{cases} G = F \cup \{C\} \\ \\ G = F \setminus \{C\} \end{cases}$$

The DRAT proof system

Reverse unit propagation (RUP) in F

$$\dfrac{\dfrac{\dfrac{C_0 \quad \in F}{D_1}\,\text{SUB} \qquad C_1 \quad \in F}{\ddots}\,\text{RES} \qquad \dfrac{D_{n-1} \qquad C_{n-1} \quad \in F}{D_n}\,\text{RES}}{}$$

D_n RUP in F

Blocked clauses

RAT upon x

$\cup\!\!\!\text{I}$

blocked upon x

for all in F

$$\text{RES}\,\dfrac{C \lor x \qquad D \lor \overline{x}}{C \lor D}$$

tautology

Delete Resolution Asymmetric Tautology (DRAT) proof system

$$F \Rightarrow_{\text{DRAT}} G \quad \begin{cases} G = F \cup \{C\} & \begin{cases} C \text{ is a RUP in } F \\ C \text{ is a RAT in } F \end{cases} \\ G = F \setminus \{C\} \end{cases}$$

A DRAT proof of G from F is a sequence of DRAT inferences:

$$F = F_0 \Rightarrow_{\text{DRAT}} F_1 \Rightarrow_{\text{DRAT}} \cdots \Rightarrow_{\text{DRAT}} F_{n-1} \Rightarrow_{\text{DRAT}} F_n = G$$

XOR reasoning

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i *evaluated to true is* k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{array}{lllll} x & & \oplus\, t & & = 0 \\ x \oplus y & & \oplus\, t & \oplus\, w & = 1 \\ x \oplus y \oplus z & & & \oplus\, w & = 0 \end{array}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{aligned} x \quad\quad\quad\quad & \oplus t \quad\quad\ = 0 \\ x \oplus y \quad\quad & \oplus t \oplus w = 1 \\ x \oplus y \oplus z & \quad\ \ \oplus w = 0 \end{aligned}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{array}{lllllllll}
x & & & & & \oplus & t & & = 0 \\
 & & y & & & & & \oplus\, w & = 1 \\
x & \oplus & y & \oplus & z & & & \oplus\, w & = 0
\end{array}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0,1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{array}{rcccccccl} x & & & & & \oplus t & & & = 0 \\ & & y & & & & & \oplus w & = 1 \\ & & y & \oplus z & & \oplus t & & \oplus w & = 0 \end{array}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{aligned} x \qquad\qquad \oplus\, t \qquad &= 0 \\ y \qquad\qquad \oplus\, w &= 1 \\ y \oplus z \oplus t \oplus w &= 0 \end{aligned}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{aligned}
x \quad\quad\quad\quad\quad\quad \oplus t \quad\quad &= 0 \\
y \quad\quad\quad\quad\quad \oplus w &= 1 \\
z \oplus t \quad\quad &= 1
\end{aligned}$$

XOR reasoning

XOR constraints expressions of the form $x_1 \oplus \cdots \oplus x_n = k$ with $k \in \{0, 1\}$

true iff the parity of the number of x_i evaluated to true is k

XOR inferences

XOR constraint addition sum modulo 2 of the parity constraints

$$\text{ADD}\ \frac{x \oplus y \oplus t = 1 \qquad x \oplus z \oplus t = 1}{y \oplus z = 0}$$

XOR definition define a fresh variable x as a XOR constraint

$$\text{DEF}\ \frac{}{x \oplus y_1 \oplus \ldots \oplus y_n = k}$$

Gaussian elimination

$$\begin{array}{cccccccl} x & & & & \oplus\, t & & = 0 \\ & y & & & & \oplus\, w & = 1 \\ & & & z & \oplus\, t & & = 1 \end{array}$$

Direct encoding of XOR constraints

smallest CNF formula $D(X)$ semantically equivalent to X

$$X = (x \oplus y \oplus z = 1)$$

$$D(X) = (x \vee y \vee z) \wedge (\overline{x} \vee \overline{y} \vee z) \wedge (\overline{x} \vee y \vee \overline{z}) \wedge (x \vee \overline{y} \vee \overline{z})$$

$D(X)$ is exponentially sized on the size of X

XOR reasoning in SAT solving

Direct encoding of XOR constraints

smallest CNF formula $D(X)$ semantically equivalent to X

$$X = (x \oplus y \oplus z = 1)$$

$$D(X) = (x \vee y \vee z) \wedge (\overline{x} \vee \overline{y} \vee z) \wedge (\overline{x} \vee y \vee \overline{z}) \wedge (x \vee \overline{y} \vee \overline{z})$$

$D(X)$ is exponentially sized on the size of X

Gaussian elimination in SAT solving

Soos et al. (2009)

$$\begin{tikzcd} F \arrow[r, "\text{detection}"] & D(X_1, \dots, X_n) \wedge F' \arrow[d, "\text{Gaussian elimination}"] \\ \square & D(Y_1, \dots, Y_n) \wedge F' \arrow[l, "\text{SAT solver}"] \end{tikzcd}$$

XOR reasoning in SAT solving

Direct encoding of XOR constraints

smallest CNF formula $D(X)$ semantically equivalent to X

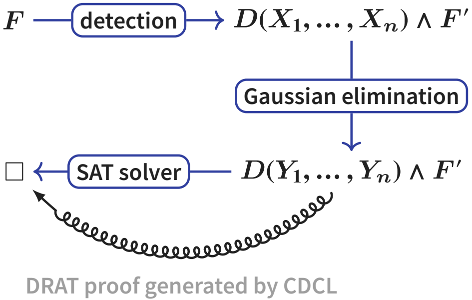
$$X = (x \oplus y \oplus z = 1)$$

$$D(X) = (x \lor y \lor z) \land (\overline{x} \lor \overline{y} \lor z) \land (\overline{x} \lor y \lor \overline{z}) \land (x \lor \overline{y} \lor \overline{z})$$

$D(X)$ is exponentially sized on the size of X

Gaussian elimination in SAT solving

Soos et al. (2009)



XOR reasoning in SAT solving

Direct encoding of XOR constraints

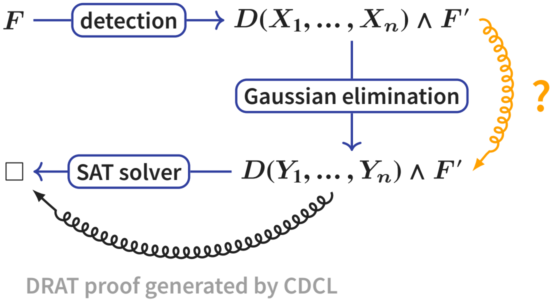
smallest CNF formula $D(X)$ semantically equivalent to X

$$X = (x \oplus y \oplus z = 1)$$

$$D(X) = (x \lor y \lor z) \land (\overline{x} \lor \overline{y} \lor z) \land (\overline{x} \lor y \lor \overline{z}) \land (x \lor \overline{y} \lor \overline{z})$$

$D(X)$ is exponentially sized on the size of X

Gaussian elimination in SAT solving *Soos et al. (2009)*



Problem finding a DRAT proof for the Gaussian elimination part

Unsatisfiability proofs for XOR reasoning

Problem finding a DRAT proof for the Gaussian elimination part

Unsatisfiability proofs for XOR reasoning

Problem finding a DRAT proof for the Gaussian elimination part

Observation Gaussian elimination only uses XOR constraint additions

Unsatisfiability proofs for XOR reasoning

$$\begin{array}{c} \{X_1, \dots, X_n\} \\ \rightsquigarrow \\ \{Y_1, \dots, Y_n\} \end{array}$$

XOR proof by
Gaussian elimination

Problem finding a DRAT proof for the Gaussian elimination part

Observation Gaussian elimination only uses XOR constraint additions

Unsatisfiability proofs for XOR reasoning

$$\{X_1, \dots, X_n\} \rightsquigarrow \{Y_1, \dots, Y_n\}$$

XOR proof by
Gaussian elimination

Problem finding a DRAT proof for the Gaussian elimination part

Observation Gaussian elimination only uses XOR constraint additions

Solution translate the XOR proof of Gaussian elimination into a DRAT proof
every addition inference must be translated

Unsatisfiability proofs for XOR reasoning

$$\begin{tikzcd}[column sep=huge]
\{X_1, \dots, X_n\} \arrow[d, squiggly] & D(X_1, \dots, X_n) \arrow[d, squiggly] \\
\{Y_1, \dots, Y_n\} & D(Y_1, \dots, Y_n)
\end{tikzcd}$$

XOR proof by
Gaussian elimination

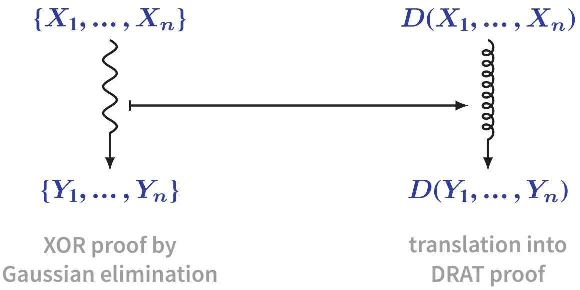
translation into
DRAT proof

Problem finding a DRAT proof for the Gaussian elimination part

Observation Gaussian elimination only uses XOR constraint additions

Solution translate the XOR proof of Gaussian elimination into a DRAT proof
every addition inference must be translated

Unsatisfiability proofs for XOR reasoning



Problem finding a DRAT proof for the Gaussian elimination part

Observation Gaussian elimination only uses XOR constraint additions

Solution translate the XOR proof of Gaussian elimination into a DRAT proof
every addition inference must be translated

Contribution two translation methods

- **Direct translation** exponential in $|D(X_1, \dots, X_n)| + |D(Y_1, \dots, Y_n)|$
- **T-translation** polynomial in $|D(X_1, \dots, X_n)| + |D(Y_1, \dots, Y_n)|$

Direct translation

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \;+\; (y \oplus z \oplus t \oplus w = 1) \;=\; (x \oplus w = 1)$$

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \; + \; (y \oplus z \oplus t \oplus w = 1) \; = \; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \ + \ (y \oplus z \oplus t \oplus w = 1) \ = \ (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \; + \; (y \oplus z \oplus t \oplus w = 1) \; = \; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \lor w) \land (\overline{x} \lor \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \ + \ (y \oplus z \oplus t \oplus w = 1) \ = \ (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.

$$\overline{x} \vee \overline{w}$$

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \; + \; (y \oplus z \oplus t \oplus w = 1) \; = \; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.
- Derive every clause as a resolvent upon an eliminated variable.

$$\text{RUP}\;\frac{\overline{x} \vee \overline{w} \vee y \qquad \overline{x} \vee \overline{w} \vee \overline{y}}{\overline{x} \vee \overline{w}}$$

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \;+\; (y \oplus z \oplus t \oplus w = 1) \;=\; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.
- Derive every clause as a resolvent upon an eliminated variable.
- When $n - 1$ eliminated variables are eliminated, top-level clauses are RUPs in $D(X) \wedge D(Y)$.

$$\dfrac{\text{RUP}\;\dfrac{\overline{x} \vee \overline{w} \vee y \vee z \qquad \overline{x} \vee \overline{w} \vee y \vee \overline{z}}{\overline{x} \vee \overline{w} \vee y} \qquad \dfrac{\overline{x} \vee \overline{w} \vee \overline{y} \vee z \qquad \overline{x} \vee \overline{w} \vee \overline{y} \vee \overline{z}}{\overline{x} \vee \overline{w} \vee \overline{y}}\;\text{RUP}}{\overline{x} \vee \overline{w}}\;\text{RUP}$$

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \; + \; (y \oplus z \oplus t \oplus w = 1) \; = \; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.
- Derive every clause as a resolvent upon an eliminated variable.
- When $n - 1$ eliminated variables are eliminated, top-level clauses are RUPs in $D(X) \wedge D(Y)$.

$$\cfrac{\cfrac{\cfrac{D(X) \wedge D(Y)}{\overline{x} \vee \overline{w} \vee y \vee z \qquad \overline{x} \vee \overline{w} \vee y \vee \overline{z}}\,\text{RUP}}{\overline{x} \vee \overline{w} \vee y}\,\text{RUP} \qquad \cfrac{\cfrac{D(X) \wedge D(Y)}{\overline{x} \vee \overline{w} \vee \overline{y} \vee z \qquad \overline{x} \vee \overline{w} \vee \overline{y} \vee \overline{z}}\,\text{RUP}}{\overline{x} \vee \overline{w} \vee \overline{y}}\,\text{RUP}}{\overline{x} \vee \overline{w}}\,\text{RUP}$$

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = \mathbf{0}) \;+\; (y \oplus z \oplus t \oplus w = \mathbf{1}) \;=\; (x \oplus w = \mathbf{1})$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.
- Derive every clause as a resolvent upon an eliminated variable.
- When $n - 1$ eliminated variables are eliminated, top-level clauses are RUPs in $D(X) \wedge D(Y)$.

Problem proofs are exponential in the size of the XOR constraints

Direct translation

Example XOR addition of the form $X + Y = Z$

$$(x \oplus y \oplus z \oplus t = 0) \;+\; (y \oplus z \oplus t \oplus w = 1) \;=\; (x \oplus w = 1)$$

Direct translation of addition inferences

- The clauses $D(Z) = (x \vee w) \wedge (\overline{x} \vee \overline{w})$ must be derived.
- Variables y, z, t were eliminated in the addition inference.
- Start with a clause in $D(Z)$, and proceed bottom-up.
- Derive every clause as a resolvent upon an eliminated variable.
- When $n - 1$ eliminated variables are eliminated, top-level clauses are RUPs in $D(X) \wedge D(Y)$.

Problem proofs are exponential in the size of the XOR constraints

Solution bound the size of XOR constraints using Tseitin variables

T-translation

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$x_1 \oplus x_2 \quad \oplus x_3 \quad \oplus x_4 \oplus x_5 = 1$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{array}{rcl} x_1 \oplus x_2 \oplus s_1 & = & 0 \\ s_1 \oplus x_3 \oplus s_2 & = & 0 \\ s_2 \oplus x_4 \oplus x_5 & = & 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{rl}
\text{matrix} & \left.\begin{array}{rcl} x_1 \oplus x_2 \oplus s_1 & & = 0 \\ & s_1 \oplus x_3 \oplus s_2 & = 0 \\ & & s_2 \oplus x_4 \oplus x_5 = 1 \end{array}\right\} S(X) \\
\text{independent constraint} & \\
& x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1
\end{array}
$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{rl}
\text{matrix} & \left.\begin{array}{l} x_1 \oplus x_2 \oplus s_1 = 0 \\ s_1 \oplus x_3 \oplus s_2 = 0 \\ s_2 \oplus x_4 \oplus x_5 = 1 \end{array}\right\} S(X) \\
\text{independent constraint} & \\
& x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1
\end{array}
$$

T-translation of addition inferences   translating $X + Y = Z$

Intermediate XOR translation

Lift to a DRAT proof

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{array}{rlllll} \text{matrix} & x_1 \oplus x_2 \oplus s_1 & & & = 0 \\ & & s_1 \oplus x_3 \oplus s_2 & & = 0 \\ \text{independent constraint} & & & s_2 \oplus x_4 \oplus x_5 & = 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

T-translation of addition inferences translating $X + Y = Z$

Intermediate XOR translation

- matrix(Z) introduced by XOR definitions

Lift to a DRAT proof

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

matrix

independent constraint

$$\left.\begin{array}{lllllllll} x_1 \oplus x_2 \oplus s_1 & & & & & & & = & 0 \\ & & s_1 \oplus x_3 \oplus s_2 & & & & & = & 0 \\ & & & & s_2 \oplus x_4 \oplus x_5 & & & = & 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

T-translation of addition inferences translating $X + Y = Z$

Intermediate XOR translation

- matrix(Z) introduced by XOR definitions
- indep(Z) $= \sum S(X) + \sum S(Y) + \sum$ matrix(Z)

Lift to a DRAT proof

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{array}{rl} \text{matrix} & \begin{array}{lllr} x_1 \oplus x_2 \oplus s_1 & & & = 0 \\ & s_1 \oplus x_3 \oplus s_2 & & = 0 \end{array} \\ \text{independent constraint} & \begin{array}{lllr} & & s_2 \oplus x_4 \oplus x_5 & = 1 \end{array} \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

T-translation of addition inferences

translating $X + Y = Z$

Intermediate XOR translation

- matrix(Z) introduced by XOR definitions
- indep(Z) $= \sum S(X) + \sum S(Y) + \sum$ matrix(Z)

Lift to a DRAT proof

- D(matrix(Z)) introduced as blocked clauses

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{rl}
\text{matrix} & \left.\begin{array}{l}
x_1 \oplus x_2 \oplus s_1 \qquad\qquad\qquad\qquad = 0 \\
\qquad\qquad s_1 \oplus x_3 \oplus s_2 \qquad\quad = 0 \\
\text{independent constraint} \qquad\qquad\qquad s_2 \oplus x_4 \oplus x_5 = 1
\end{array}\right\} S(X)
\end{array}
$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

T-translation of addition inferences translating $X + Y = Z$

Intermediate XOR translation

- matrix(Z) introduced by XOR definitions
- indep(Z) $= \sum S(X) + \sum S(Y) + \sum$ matrix(Z)

Lift to a DRAT proof

- D(matrix(Z)) introduced as blocked clauses
- D(indep(Z)) derived by direct translation

Dodging the exponential blowup

Direct translation

$$A \rightsquigarrow B$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Direct translation

$$\begin{tikzcd} A \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\Downarrow"] & B \\ D(A) \arrow[r, rightsquigarrow] & D(B) \end{tikzcd}$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd} A \arrow[rr, rightsquigarrow, ""{name=U, below}] & & B \\ D(A) \arrow[rr, rightsquigarrow, ""{name=L, above}] & & D(B) \arrow[from=U, to=L, mapsto] \end{tikzcd}$$

possibly long
XOR constraints

Remember

- $D(X)$ CNF formula: direct encoding of X

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd}[column sep=huge] A \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\downarrow"] & B \\ D(A) \arrow[r, rightsquigarrow] & D(B) \end{tikzcd}$$

possibly long XOR constraints

exponential blowup

Remember

- $D(X)$ CNF formula: direct encoding of X

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd} A \arrow[r, squiggly] & B \\ D(A) \arrow[r, squiggly] & D(B) \end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd} A \arrow[r, squiggly] & B \end{tikzcd}$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \big\downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \big\downarrow & \\ S(A) & \rightsquigarrow & S(B) \end{array}$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd}
A \arrow[rr, squiggly] & {} \arrow[d, mapsto] & B \\
D(A) \arrow[rr, squiggly] & {} & D(B)
\end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd}
A \arrow[rr, squiggly] & {} \arrow[d, mapsto] & B \\
S(A) \arrow[rr, squiggly] & {} \arrow[d, mapsto] & S(B) \\
D(S(A)) \arrow[rr, squiggly] & {} & D(S(B))
\end{tikzcd}$$

possibly long XOR constraints

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd} A \arrow[rr, squiggly] & {} \arrow[d, mapsto] & B \\ D(A) \arrow[rr, squiggly] & {} & D(B) \end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd} A \arrow[rr, squiggly] & {} \arrow[d, mapsto] & B \\ S(A) \arrow[rr, squiggly] & {} \arrow[d, mapsto] & S(B) \\ D(S(A)) \arrow[rr, squiggly] & {} & D(S(B)) \end{tikzcd}$$

possibly long XOR constraints

possibly long XORs, too!

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

possibly long XOR constraints

possibly long XORs, too!

exponential blowup

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

possibly long XOR constraints

possibly long XORs, too!

exponential blowup

$$X + Y = Z \quad \Rightarrow \quad \text{indep}(Z) = \sum S(X) + \sum S(Y) + \sum \text{matrix}(Z)$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \mapsto\!\!\downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \mapsto\!\!\downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \mapsto\!\!\downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

possibly long XOR constraints

$$X + Y = Z \quad \Rightarrow \quad \mathsf{indep}(Z) = \sum S(X) + \sum S(Y) + \sum \mathsf{matrix}(Z)$$

reorder!
$x_1 < \cdots < x_n$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

possibly long XOR constraints

XOR constraint size bounded by 6!

$X + Y = Z \quad \Rightarrow \quad \text{indep}(Z) = \sum S(X) + \sum S(Y) + \sum \text{matrix}(Z)$

reorder! $x_1 < \cdots < x_n$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ D(A) & \rightsquigarrow & D(B) \end{array}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{array}{ccc} A & \rightsquigarrow & B \\ & \downarrow & \\ S(A) & \rightsquigarrow & S(B) \\ & \downarrow & \\ D(S(A)) & \rightsquigarrow & D(S(B)) \end{array}$$

possibly long XOR constraints

XOR constraint size bounded by 6!

polynomial!

$X + Y = Z \quad \Rightarrow \quad \text{indep}(Z) = \sum S(X) + \sum S(Y) + \sum \text{matrix}(Z)$

reorder! $x_1 < \cdots < x_n$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd} A \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & B \\ D(A) \arrow[rr, rightsquigarrow] & {} & D(B) \end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd} & A \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & B \\ A \arrow[r, rightsquigarrow] & S(A) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(B) \\ & D(S(A)) \arrow[rr, rightsquigarrow] & {} & D(S(B)) \end{tikzcd}$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd}
A \arrow[r, squiggly, ""{name=U, below}] & B \\
D(A) \arrow[r, squiggly, ""{name=V}] & D(B)
\arrow[from=U, to=V, mapsto]
\end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd}
 & A \arrow[r, squiggly, ""{name=T, below}] & B \\
A \arrow[r, squiggly, ""{name=L, below}] & S(A) \arrow[r, squiggly, ""{name=M, below}] & S(B) \\
D(A) \arrow[r, squiggly, ""{name=P}] & D(S(A)) \arrow[r, squiggly, ""{name=Q}] & D(S(B))
\arrow[from=T, to=M, mapsto]
\arrow[from=L, to=P, mapsto]
\arrow[from=M, to=Q, mapsto]
\end{tikzcd}$$

$$\text{indep}(X) = X + \sum \text{matrix}(X)$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$
\begin{tikzcd}
A \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & B \\
D(A) \arrow[rr, rightsquigarrow] & {} & D(B)
\end{tikzcd}
$$

possibly long XOR constraints

exponential blowup

T-translation

$$
\begin{tikzcd}
 & & A \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & B \\
A \arrow[r, rightsquigarrow] & {} \arrow[d, mapsto] & S(A) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(B) \\
D(A) \arrow[rr, rightsquigarrow] & {} & D(S(A)) \arrow[rr, rightsquigarrow] & {} & D(S(B))
\end{tikzcd}
$$

bounded by $O(|D(A)|)$

$\text{indep}(X) = X + \sum \text{matrix}(X)$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation

$$\begin{tikzcd} A \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\downarrow"] & B \\ D(A) \arrow[r, rightsquigarrow] & D(B) \end{tikzcd}$$

possibly long XOR constraints

exponential blowup

T-translation

$$\begin{tikzcd} & A \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\downarrow"] & B & \\ A \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\downarrow"] & S(A) \arrow[r, rightsquigarrow] \arrow[dr, phantom, "\downarrow"] & S(B) & \\ D(A) \arrow[r, rightsquigarrow] & D(S(A)) \arrow[r, rightsquigarrow] & D(S(B)) \arrow[r, rightsquigarrow] & D(B) \end{tikzcd}$$

bounded by $O(|D(A)|)$

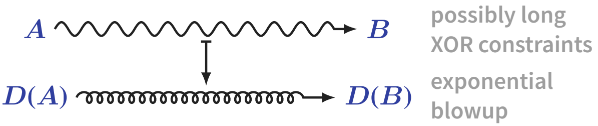
introduce by RUP

Remember

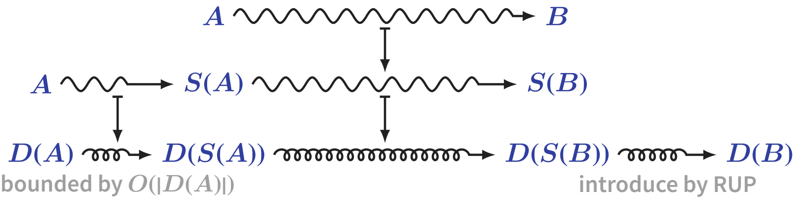
- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X
- $x_1 < \cdots < x_n$ arbitrary but fixed total order on variables

Dodging the exponential blowup

Direct translation



T-translation



Theorem direct translations are exponential in $|D(A)| + |D(B)|$

Theorem T-translations translations are polynomial in $|D(A)| + |D(B)|$

Experiments

Experimental setup

Main concern size of generated DRAT proofs

Experimental setup

Main concern size of generated DRAT proofs

Experimental setup

Experiments three proof generation methods implemented in Scala

- Direct translation
- T-translation
- BDD-based approach *Biere, Sinz (2006)*
  used as a baseline

Experimental setup

Main concern size of generated DRAT proofs

Experimental setup

Experiments three proof generation methods implemented in Scala

- Direct translation
- T-translation
- BDD-based approach *Biere, Sinz (2006)*
  used as a baseline

Benchmarks XOR reasoning records produced with CoProcessor

- 300 problems from SAT Competition 2014
  210 problems with nonempty XOR records
- average 36 000 XOR constraints, largest 350 000 XOR constraints

Experimental setup

Main concern size of generated DRAT proofs

Experimental setup

Experiments three proof generation methods implemented in Scala

- Direct translation
- T-translation
- BDD-based approach *Biere, Sinz (2006)*
  used as a baseline

Benchmarks XOR reasoning records produced with CoProcessor

- 300 problems from SAT Competition 2014
  210 problems with nonempty XOR records
- average 36 000 XOR constraints, largest 350 000 XOR constraints

Timeout 5 minutes

BDDs vs translations

Results

	BDD-based approach	direct translation	T-translation
timeouts	15%	13%	0%
shortest	0%	46%	54%

Termination with BDDs $\Rightarrow$ termination with translations

BDDs $<$ direct translation $\Rightarrow$ T-translation $<$ BDDs

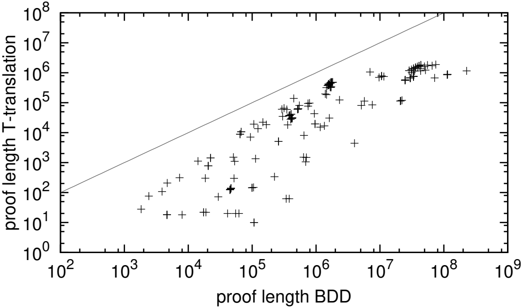
BDDs vs translations

Results

	BDD-based approach	direct translation	T-translation
timeouts	15%	13%	0%
shortest	0%	46%	54%

Termination with BDDs $\Rightarrow$ termination with translations

BDDs $<$ direct translation $\Rightarrow$ T-translation $<$ BDDs



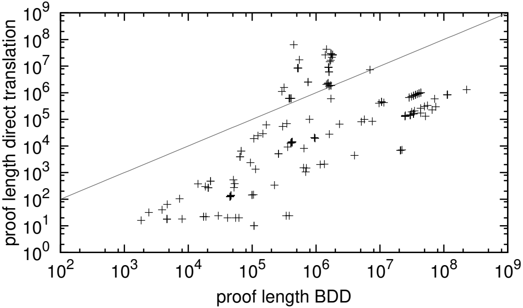
BDDs vs translations

Results

	BDD-based approach	direct translation	T-translation
timeouts	15%	13%	0%
shortest	0%	46%	54%

Termination with BDDs $\Rightarrow$ termination with translations

BDDs $<$ direct translation $\Rightarrow$ T-translation $<$ BDDs



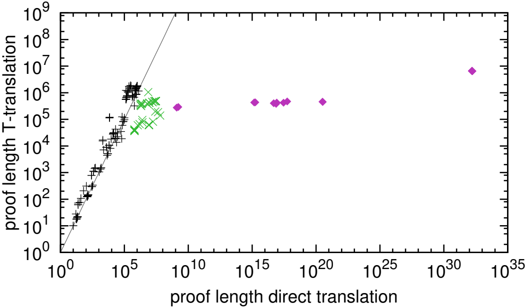
BDDs vs translations

Results

	BDD-based approach	direct translation	T-translation
timeouts	15%	13%	0%
shortest	0%	46%	54%

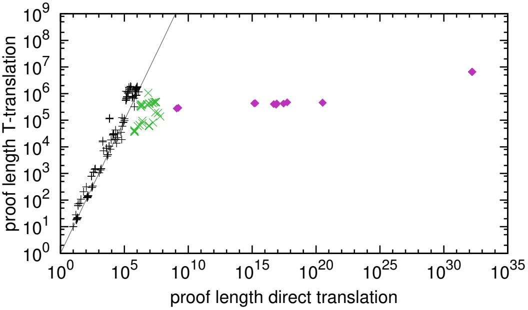
Termination with BDDs $\Rightarrow$ termination with translations

BDDs $<$ direct translation $\Rightarrow$ T-translation $<$ BDDs



Direct translations vs T-translations

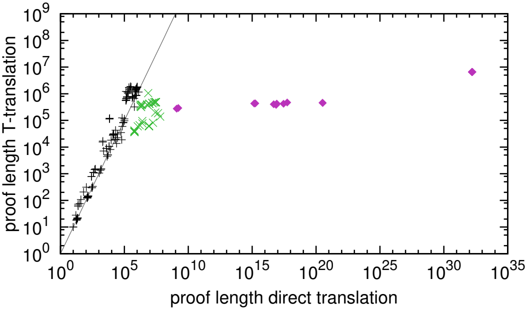
No clear *a priori* preference



Direct translations vs T-translations

No clear *a priori* preference

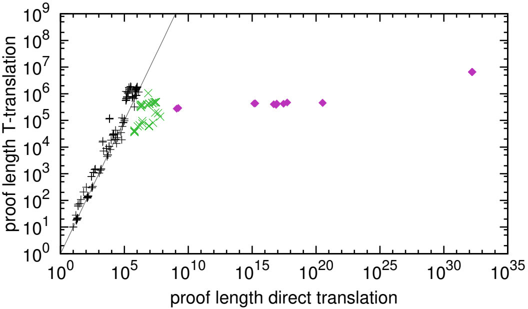
Direct translations are up to 300 times shorter



Direct translations vs T-translations

No clear *a priori* preference

Direct translations are up to 300 times shorter and up to 10^{25} times longer!

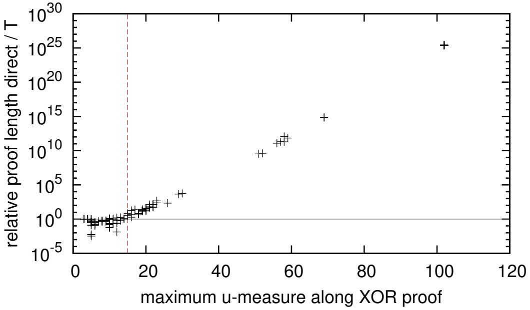


Direct translations vs T-translations

No clear *a priori* preference

Direct translations are up to 300 times shorter and up to 10^{25} times longer!

u-measure of $X + Y = Z$ total number of variables in X, Y, Z



Direct translations vs T-translations

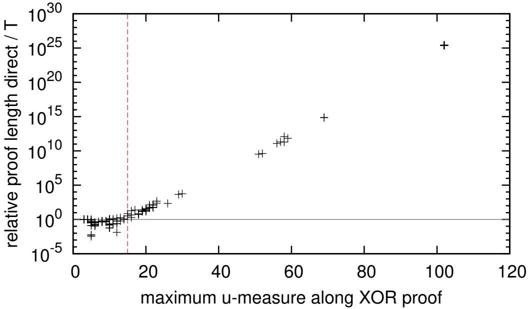
No clear *a priori* preference

Direct translations are up to 300 times shorter and up to 10^{25} times longer!

u-measure of $X + Y = Z$ total number of variables in X, Y, Z

Empirical criterion compute maximum u-measure along the input XOR proof

- $u < 15 \Rightarrow$ direct translation
- $u \geq 15 \Rightarrow$ T-translation



Conclusion

Conclusion

- XOR reasoning is essential for state-of-the-art SAT solvers.
- We enable XOR reasoning when unsatisfiability proofs are required.
- Translation methods outperform the BDD-based approach.
- An empirical criterion allows to decide for the shortest translation.

Backup slides

Reverse unit propagation

Proof systems for SAT solving

- simple to generate with minimum overhead

- efficient to check

Reverse unit propagation

Proof systems for SAT solving

- simple to generate with minimum overhead
- efficient to check

Self-subsuming resolution

$$\frac{C \vee D \vee x \qquad D \vee \overline{x}}{C \vee D}\,\text{SSR}$$

Subsumption

$$\frac{C}{C \vee D}\,\text{SUB}$$

Reverse unit propagation

Proof systems for SAT solving

- simple to generate with minimum overhead
- efficient to check

Self-subsuming resolution

$$\frac{C \lor D \lor x \qquad D \lor \overline{x}}{C \lor D}\,\text{SSR}$$

Subsumption

$$\frac{C}{C \lor D}\,\text{SUB}$$

Reverse unit propagation (RUP) in F

a.k.a. asymmetric tautologies in F

$$\text{SSR}\,\dfrac{\dfrac{D_{n-1} \qquad C_{n-1}}{}}{D_n}$$

$$\text{SSR}\,\dfrac{\text{SSR}\,\dfrac{\ddots\,\text{SSR}\,\dfrac{\text{SSR}\,\dfrac{\text{SUB}\,\dfrac{C_0}{D_1} \qquad C_1}{D_2} \qquad C_2}{\ddots}}{D_{n-1}} \qquad C_{n-1}}{D_n}$$

Reverse unit propagation

Proof systems for SAT solving

- simple to generate with minimum overhead
  learned clauses in SAT solvers are RUPs
- efficient to check
  RUPs can be checked by unit propagation

Self-subsuming resolution

$$\frac{C \vee D \vee x \qquad D \vee \overline{x}}{C \vee D}\,\text{SSR}$$

Subsumption

$$\frac{C}{C \vee D}\,\text{SUB}$$

Reverse unit propagation (RUP) in F

a.k.a. asymmetric tautologies in F

$$\text{SSR}\dfrac{\text{SSR}\dfrac{\begin{array}{c}\text{SSR}\dfrac{\text{SSR}\dfrac{\text{SUB}\dfrac{C_0}{D_1} \qquad C_1}{D_2} \qquad C_2}{\ddots} \\ \end{array}}{D_{n-1}} \qquad C_{n-1}}{D_n}$$

Unsatisfiability proofs for CDCL SAT solving

SAT problem deciding whether an input CNF formula is satisfiable.

CDCL SAT solving

- try to guess a satisfying assignment
- maintain arc-consistency by unit propagation
- learn implied clauses every a conflicting assignment is guessed
- inprocessing techniques replace the formula by an equisatisfiable one
- satisfiable if a satisfying interpretation is found
- unsatisfiable if the empty clause is learned

Unsatisfiability proofs for CDCL SAT solving

SAT problem deciding whether an input CNF formula is satisfiable.

CDCL SAT solving

- try to guess a satisfying assignment
- maintain arc-consistency by unit propagation
- learn implied clauses every a conflicting assignment is guessed
- inprocessing techniques replace the formula by an equisatisfiable one
- **satisfiable** if a satisfying interpretation is found
- **unsatisfiable** if the empty clause is learned

Question how to generate an unsatisfiability proof?

A (partial) solution *Goldberg, Novikov (2003)*

record the sequence of learned clauses

Theorem *Beame et al. (2004)*

learned clauses from a CNF formula F are linear resolvents from F

Unsatisfiability proofs for CDCL SAT solving

SAT problem deciding whether an input CNF formula is satisfiable.

CDCL SAT solving

- try to guess a satisfying assignment
- maintain arc-consistency by unit propagation
- learn implied clauses every a conflicting assignment is guessed
- inprocessing techniques replace the formula by an equisatisfiable one
- satisfiable if a satisfying interpretation is found
- unsatisfiable if the empty clause is learned

Question how to generate an unsatisfiability proof?

A (partial) solution *Goldberg, Novikov (2003)*
record the sequence of learned clauses

Theorem *Beame et al. (2004)*
learned clauses from a CNF formula F are linear resolvents from F

Problem inprocessing techniques, in particular XOR reasoning, are not covered

XOR reasoning and unsatisfiability proofs

CDCL SAT solving

branching heuristics

clause removal

symmetry breaking

cardinality resolution

learning schemas

XOR reasoning

unsatisfiability proofs

XOR reasoning and unsatisfiability proofs

CDCL SAT solving

	branching heuristics	clause removal
symmetry breaking	cardinality resolution	learning schemas
	XOR reasoning	**unsatisfiability proofs**

XOR reasoning and unsatisfiability proofs

CDCL SAT solving

branching heuristics

clause removal

symmetry breaking

cardinality resolution

learning schemas

XOR reasoning

unsatisfiability proofs

CDCL is not polynomially bounded in the presence of encoded XOR constraints.
Urquhart (1987), Beame et al. (2004)

XOR constraints often occur in cryptography and bit-vector arithmetic.
Massacci et al. (2000)

Polynomial procedures for XOR reasoning can be integrated in SAT solvers.
Soos et al. (2009), Laitinen et al. (2014)

XOR reasoning and unsatisfiability proofs

CDCL SAT solving

	branching heuristics	clause removal
symmetry breaking	cardinality resolution	learning schemas
	XOR reasoning	unsatisfiability proofs

SAT solvers' architectures are complex, and bugs are hard to detect.

- false positives — partial interpretations as witnesses
- false negatives — unsatisfiability proofs are required

Unless $P = coNP$, validating unsatisfiability results is intractable.

The DRAT proof standard provides certificates for most techniques.
Heule et al. (2013, 2015), Philipp et al. (2014)

XOR reasoning and unsatisfiability proofs

CDCL SAT solving

	branching heuristics	clause removal
symmetry breaking	cardinality resolution	learning schemas
	XOR reasoning	unsatisfiability proofs

Problem generating unsatisfiability proofs for XOR reasoning techniques.
Biere et al. (2006, 2015)

XOR reasoning is currently disabled when unsatisfiability proofs are required.

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$x_1 \oplus x_2 \qquad \oplus x_3 \qquad \oplus x_4 \oplus x_5 = 1$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{aligned} x_1 \oplus x_2 \oplus s_1 &= 0 \\ s_1 \oplus x_3 \oplus s_2 &= 0 \\ s_2 \oplus x_4 \oplus x_5 &= 1 \end{aligned}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

matrix

independent constraint

$$\left.\begin{array}{lllllllll} x_1 \oplus x_2 \oplus s_1 & & & & & & & = & \mathbf{0} \\ & & s_1 \oplus x_3 \oplus s_2 & & & & & = & \mathbf{0} \\ & & & & s_2 \oplus x_4 \oplus x_5 & & & = & \mathbf{1} \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = \mathbf{1}$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

matrix

independent constraint

$$\left.\begin{array}{rcl} x_1 \oplus x_2 \oplus s_1 & & = 0 \\ s_1 \oplus x_3 \oplus s_2 & & = 0 \\ s_2 \oplus x_4 \oplus x_5 & & = 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

Deriving the split representation

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{r l}
\text{matrix} & \left.\begin{array}{l} x_1 \oplus x_2 \oplus s_1 = 0 \\ s_1 \oplus x_3 \oplus s_2 = 0 \\ s_2 \oplus x_4 \oplus x_5 = 1 \end{array}\right\} S(X) \\
\text{independent constraint} & \\
& x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1
\end{array}
$$

Deriving the split representation

- Introduce the clauses in $D(X)$ for every XOR constraint X in the matrix.

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{array}{rlllllllll} \text{matrix} & x_1 \oplus x_2 \oplus s_1 & & & & & & & = 0 \\ & & & s_1 \oplus x_3 \oplus s_2 & & & & & = 0 \\ \text{independent constraint} & & & & & s_2 \oplus x_4 \oplus x_5 & & & = 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \quad \oplus x_3 \quad \oplus x_4 \oplus x_5 = 1$$

Deriving the split representation

- Introduce the clauses in $D(X)$ for every XOR constraint X in the matrix.

Proposition if a XOR constraint X contains a fresh variable, then the clauses in $D(X)$ can be introduced as blocked clauses

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{rl}
\text{matrix} & \left.\begin{array}{llllllllllll}
x_1 \oplus x_2 \oplus s_1 & & & & = 0 \\
& s_1 \oplus x_3 \oplus s_2 & & & = 0 \\
\text{independent constraint} & & s_2 \oplus x_4 \oplus x_5 & & = 1
\end{array}\right\} S(X) \\
& x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1
\end{array}
$$

Deriving the split representation

- Introduce the clauses in $D(X)$ for every XOR constraint X in the matrix.
- Derive the independent constraint by addition

Proposition if a XOR constraint X contains a fresh variable, then the clauses in $D(X)$ can be introduced as blocked clauses

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$
\begin{array}{rl}
\text{matrix} & \left.\begin{array}{l} x_1 \oplus x_2 \oplus s_1 = 0 \\ s_1 \oplus x_3 \oplus s_2 = 0 \end{array}\right. \\
\text{independent constraint} & s_2 \oplus x_4 \oplus x_5 = 1
\end{array}
\Bigg\} \; S(X)
$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

Deriving the split representation

- Introduce the clauses in $D(X)$ for every XOR constraint X in the matrix.
- Derive the independent constraint by addition

Splitter

$$
\dfrac{\dfrac{x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1 \qquad \dfrac{}{x_1 \oplus x_2 \oplus s_1 = 0}\,\text{DEF}}{s_1 \oplus x_3 \oplus x_4 \oplus x_5 = 1}\,\text{ADD} \qquad \dfrac{}{s_1 \oplus x_3 \oplus s_2 = 0}\,\text{DEF}}{s_2 \oplus x_4 \oplus x_5 = 1}\,\text{ADD}
$$

Split representation of XOR constraints

We assume a total order in variables with $x_1 < \cdots < x_n$

XOR constraint splitting

$$\left.\begin{array}{rrl} \text{matrix} & x_1 \oplus x_2 \oplus s_1 & = 0 \\ & s_1 \oplus x_3 \oplus s_2 & = 0 \\ \text{independent constraint} & s_2 \oplus x_4 \oplus x_5 & = 1 \end{array}\right\} S(X)$$

$$x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1$$

Deriving the split representation

- Introduce the clauses in $D(X)$ for every XOR constraint X in the matrix.
- Derive the independent constraint by addition
- Translate using the direct translation

Splitter

$$\dfrac{\dfrac{x_1 \oplus x_2 \oplus x_3 \oplus x_4 \oplus x_5 = 1 \qquad \dfrac{}{x_1 \oplus x_2 \oplus s_1 = 0}\,\text{DEF}}{s_1 \oplus x_3 \oplus x_4 \oplus x_5 = 1}\,\text{ADD} \qquad \dfrac{}{s_1 \oplus x_3 \oplus s_2 = 0}\,\text{DEF}}{s_2 \oplus x_4 \oplus x_5 = 1}\,\text{ADD}$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = \mathbf{0}) \;+\; (x \oplus z \oplus t \oplus v \oplus w = \mathbf{1}) \;=\; (y \oplus u \oplus v \oplus w = \mathbf{1})$$

Split representations

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$$x \oplus y \oplus s_0 = 0$$

$$s_0 \oplus z \oplus s_1 = 0$$

$$s_1 \oplus t \oplus u = 0$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$$x \oplus y \oplus s_0 = 0$$

$$s_0 \oplus z \oplus s_1 = 0$$

$$s_1 \oplus t \oplus u = 0$$

$$x \oplus z \oplus s_2 = 0$$

$$s_2 \oplus t \oplus s_3 = 0$$

$$s_3 \oplus v \oplus w = 1$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$$x \oplus y \oplus s_0 = 0$$
$$s_0 \oplus z \oplus s_1 = 0$$
$$s_1 \oplus t \oplus u = 0$$
$$x \oplus z \oplus s_2 = 0$$
$$s_2 \oplus t \oplus s_3 = 0$$
$$s_3 \oplus v \oplus w = 1$$
$$y \oplus u \oplus s_4 = 0$$
$$s_4 \oplus v \oplus w = 1$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$x \oplus y \oplus s_0 = 0$ axiom because X is a premise

$s_0 \oplus z \oplus s_1 = 0$ axiom because X is a premise

$s_1 \oplus t \oplus u = 0$ axiom because X is a premise

$x \oplus z \oplus s_2 = 0$

$s_2 \oplus t \oplus s_3 = 0$

$s_3 \oplus v \oplus w = 1$

$y \oplus u \oplus s_4 = 0$

$s_4 \oplus v \oplus w = 1$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$x \oplus y \oplus s_0 = 0$

$s_0 \oplus z \oplus s_1 = 0$

$s_1 \oplus t \oplus u = 0$

$x \oplus z \oplus s_2 = 0$ axiom because Y is a premise

$s_2 \oplus t \oplus s_3 = 0$ axiom because Y is a premise

$s_3 \oplus v \oplus w = 1$ axiom because Y is a premise

$y \oplus u \oplus s_4 = 0$

$s_4 \oplus v \oplus w = 1$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$x \oplus y \oplus s_0 = 0$

$s_0 \oplus z \oplus s_1 = 0$

$s_1 \oplus t \oplus u = 0$

$x \oplus z \oplus s_2 = 0$

$s_2 \oplus t \oplus s_3 = 0$

$s_3 \oplus v \oplus w = 1$

$y \oplus u \oplus s_4 = 0$ introduced by RAT because s_4 is fresh

$s_4 \oplus v \oplus w = 1$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$$x \oplus y \oplus s_0 = 0$$

$$s_0 \oplus z \oplus s_1 = 0$$

$$s_1 \oplus t \oplus u = 0$$

$$x \oplus z \oplus s_2 = 0$$

$$s_2 \oplus t \oplus s_3 = 0$$

$$s_3 \oplus v \oplus w = 1$$

$$y \oplus u \oplus s_4 = 0$$

$$s_4 \oplus v \oplus w = 1$$ to be derived

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations

$$x \oplus y \oplus s_0 = 0$$
$$s_0 \oplus z \oplus s_1 = 0$$
$$s_1 \oplus t \oplus u = 0$$
$$x \oplus z \oplus s_2 = 0$$
$$s_2 \oplus t \oplus s_3 = 0$$
$$s_3 \oplus v \oplus w = 1$$
$$y \oplus u \oplus s_4 = 0$$
$$s_4 \oplus v \oplus w = 1$$

to be derived

Proposition $s_4 \oplus v \oplus w = 0$ results from adding all other XOR constraints

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \;+\; (x \oplus z \oplus t \oplus v \oplus w = 1) \;=\; (y \oplus u \oplus v \oplus w = 1)$$

Split representations

Sorted XOR constraints

$$x \oplus y \oplus s_0 = 0$$

$$s_0 \oplus z \oplus s_1 = 0$$

$$s_1 \oplus t \oplus u = 0$$

$$x \oplus z \oplus s_2 = 0$$

$$s_2 \oplus t \oplus s_3 = 0$$

$$s_3 \oplus v \oplus w = 1$$

$$y \oplus u \oplus s_4 = 0$$

$$s_4 \oplus v \oplus w = 1$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations

Sorted XOR constraints

$$x \oplus y \oplus s_0 = 0$$

$$s_0 \oplus z \oplus s_1 = 0$$

$$s_1 \oplus t \oplus u = 0$$

$$x \oplus z \oplus s_2 = 0$$

$$s_2 \oplus t \oplus s_3 = 0$$

$$s_3 \oplus v \oplus w = 1$$

$$y \oplus u \oplus s_4 = 0$$

$$s_4 \oplus v \oplus w = 1$$

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	
$x \oplus z \oplus s_2 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	
$x \oplus z \oplus s_2 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation   Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	
$s_3 \oplus v \oplus w = 1$	
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$
$y \oplus u \oplus s_4 = 0$	
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$
$s_4 \oplus v \oplus w = 1$	

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$
$s_4 \oplus v \oplus w = 1$	

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	
$s_4 \oplus v \oplus w = 1$		

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \; + \; (x \oplus z \oplus t \oplus v \oplus w = 1) \; = \; (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	
$s_4 \oplus v \oplus w = 1$		

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	
$s_4 \oplus v \oplus w = 1$		

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	$u \oplus s_1 \oplus s_2 \oplus s_4 = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	
$s_4 \oplus v \oplus w = 1$		

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation   Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	$u \oplus s_1 \oplus s_2 \oplus s_4 = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	$t \oplus s_2 \oplus s_4 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	
$s_4 \oplus v \oplus w = 1$		

Proposition   in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) + (x \oplus z \oplus t \oplus v \oplus w = 1) = (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	$u \oplus s_1 \oplus s_2 \oplus s_4 = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	$t \oplus s_2 \oplus s_4 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	$s_3 \oplus s_4 = 0$
$s_4 \oplus v \oplus w = 1$		

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	$u \oplus s_1 \oplus s_2 \oplus s_4 = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	$t \oplus s_2 \oplus s_4 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	$s_3 \oplus s_4 = 0$
$s_4 \oplus v \oplus w = 1$		$s_4 \oplus v \oplus w = 1$

Proposition in this order, the size of partial additions is bounded by 6

T-translation of XOR inferences

Order on variables: $x < y < z < t < u < v < w$

T-translation Consider the XOR addition of the form $X + Y = Z$:

$$(x \oplus y \oplus z \oplus t \oplus u = 0) \ + \ (x \oplus z \oplus t \oplus v \oplus w = 1) \ = \ (y \oplus u \oplus v \oplus w = 1)$$

Split representations	Sorted XOR constraints	Cumulative addition
$x \oplus y \oplus s_0 = 0$	$x \oplus y \oplus s_0 = 0$	
$s_0 \oplus z \oplus s_1 = 0$	$x \oplus z \oplus s_2 = 0$	
$s_1 \oplus t \oplus u = 0$	$y \oplus u \oplus s_4 = 0$	$y \oplus z \oplus s_0 \oplus s_2 = 0$
$x \oplus z \oplus s_2 = 0$	$s_0 \oplus z \oplus s_1 = 0$	$z \oplus u \oplus s_0 \oplus s_2 \oplus s_4 = 0$
$s_2 \oplus t \oplus s_3 = 0$	$s_1 \oplus t \oplus u = 0$	$u \oplus s_1 \oplus s_2 \oplus s_4 = 0$
$s_3 \oplus v \oplus w = 1$	$s_2 \oplus t \oplus s_3 = 0$	$t \oplus s_2 \oplus s_4 = 0$
$y \oplus u \oplus s_4 = 0$	$s_3 \oplus v \oplus w = 1$	$s_3 \oplus s_4 = 0$
$s_4 \oplus v \oplus w = 1$		$s_4 \oplus v \oplus w = 1$

Direct translations vs T-translations

Direct translations

$$A_0 \rightsquigarrow A_n$$

Direct translations vs T-translations

Direct translations

$$A_0 \Longrightarrow A_1 \Longrightarrow \cdots \Longrightarrow A_{n-1} \Longrightarrow A_n$$

Direct translations vs T-translations

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[r, Rightarrow] \arrow[d, mapsto] & A_1 \arrow[r, Rightarrow] \arrow[d, mapsto] & \cdots \arrow[r, Rightarrow] \arrow[d, mapsto] & A_{n-1} \arrow[r, Rightarrow] \arrow[d, mapsto] & A_n \\
D(A_0) \arrow[r, rightsquigarrow] & D(A_1) \arrow[r, rightsquigarrow] & \cdots \arrow[r, rightsquigarrow] & D(A_{n-1}) \arrow[r, rightsquigarrow] & D(A_n)
\end{tikzcd}
$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Direct translations vs T-translations

Direct translations

$$A_0 \Longrightarrow A_1 \Longrightarrow \cdots \Longrightarrow A_{n-1} \Longrightarrow A_n$$

$$D(A_0) \rightsquigarrow D(A_1) \rightsquigarrow \cdots \rightsquigarrow D(A_{n-1}) \rightsquigarrow D(A_n)$$

T-translation

$$A_0 \rightsquigarrow A_n$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Direct translations vs T-translations

Direct translations

$$\begin{tikzcd}
A_0 \arrow[r, Rightarrow] \arrow[d, mapsto] & A_1 \arrow[r, Rightarrow] \arrow[d, mapsto] & \cdots \arrow[r, Rightarrow] & A_{n-1} \arrow[r, Rightarrow] \arrow[d, mapsto] & A_n \arrow[d, mapsto] \\
D(A_0) \arrow[r, rightsquigarrow] & D(A_1) \arrow[r, rightsquigarrow] & \cdots \arrow[r, rightsquigarrow] & D(A_{n-1}) \arrow[r, rightsquigarrow] & D(A_n)
\end{tikzcd}$$

T-translation

$$A_0 \Longrightarrow A_1 \Longrightarrow \cdots \Longrightarrow A_n$$

Remember

- $D(X)$ CNF formula: direct encoding of X

Direct translations vs T-translations

Direct translations

$$\begin{array}{ccccccccc}
A_0 & \Longrightarrow & A_1 & \Longrightarrow & \cdots & \Longrightarrow & A_{n-1} & \Longrightarrow & A_n \\
 & \downarrow & & \downarrow & & \downarrow & & \downarrow & \\
D(A_0) & \rightsquigarrow & D(A_1) & \rightsquigarrow & \cdots & \rightsquigarrow & D(A_{n-1}) & \rightsquigarrow & D(A_n)
\end{array}$$

T-translation

$$\begin{array}{ccccccc}
A_0 & \Longrightarrow & A_1 & \Longrightarrow & \cdots & \Longrightarrow & A_n \\
 & \downarrow & & \downarrow & & \downarrow & \\
S(A_0) & \rightsquigarrow & S(A_1) & \rightsquigarrow & \cdots & \rightsquigarrow & S(A_n)
\end{array}$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Direct translations vs T-translations

Direct translations

$$
\begin{array}{ccccccccc}
A_0 & \Longrightarrow & A_1 & \Longrightarrow & \cdots & \Longrightarrow & A_{n-1} & \Longrightarrow & A_n \\
 & \downarrow & & \downarrow & & \downarrow & & \downarrow & \\
D(A_0) & \rightsquigarrow & D(A_1) & \rightsquigarrow & \cdots & \rightsquigarrow & D(A_{n-1}) & \rightsquigarrow & D(A_n)
\end{array}
$$

T-translation

$$
\begin{array}{ccccccc}
A_0 & \Longrightarrow & A_1 & \Longrightarrow & \cdots & \Longrightarrow & A_n \\
 & \downarrow & & \downarrow & & \downarrow & \\
S(A_0) & \rightsquigarrow & S(A_1) & \rightsquigarrow & \cdots & \rightsquigarrow & S(A_n) \\
 & \downarrow & & \downarrow & & \downarrow & \\
D(S(A_0)) & \rightsquigarrow & D(S(A_1)) & \rightsquigarrow & \cdots & \rightsquigarrow & D(S(A_n))
\end{array}
$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Direct translations vs T-translations

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_{n-1} \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
D(A_0) \arrow[rr, rightsquigarrow] & {} & D(A_1) \arrow[rr, rightsquigarrow] & {} & \cdots \arrow[rr, rightsquigarrow] & {} & D(A_{n-1}) \arrow[rr, rightsquigarrow] & {} & D(A_n)
\end{tikzcd}
$$

T-translation

$$
\begin{tikzcd}
 & A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
A_0 \arrow[r, rightsquigarrow] & S(A_0) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(A_1) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(A_n) \\
 & D(S(A_0)) \arrow[rr, rightsquigarrow] & {} & D(S(A_1)) \arrow[rr, rightsquigarrow] & {} & \cdots \arrow[rr, rightsquigarrow] & {} & D(S(A_n))
\end{tikzcd}
$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_{n-1} \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
D(A_0) \arrow[rr, decorate, decoration={coil}] & {} & D(A_1) \arrow[rr, decorate, decoration={coil}] & {} & \cdots \arrow[rr, decorate, decoration={coil}] & {} & D(A_{n-1}) \arrow[rr, decorate, decoration={coil}] & {} & D(A_n)
\end{tikzcd}
$$

T-translation

$$
\begin{tikzcd}
 & & A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
A_0 \arrow[rr, squiggly] & {} \arrow[d, mapsto] & S(A_0) \arrow[rr, squiggly] & {} \arrow[d, mapsto] & S(A_1) \arrow[rr, squiggly] & {} \arrow[d, mapsto] & \cdots \arrow[rr, squiggly] & {} \arrow[d, mapsto] & S(A_n) \\
D(A_0) \arrow[rr, decorate, decoration={coil}] & {} & D(S(A_0)) \arrow[rr, decorate, decoration={coil}] & {} & D(S(A_1)) \arrow[rr, decorate, decoration={coil}] & {} & \cdots \arrow[rr, decorate, decoration={coil}] & {} & D(S(A_n))
\end{tikzcd}
$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Direct translations vs T-translations

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_{n-1} \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
D(A_0) \arrow[rr, rightsquigarrow] & {} & D(A_1) \arrow[rr, rightsquigarrow] & {} & \cdots \arrow[rr, rightsquigarrow] & {} & D(A_{n-1}) \arrow[rr, rightsquigarrow] & {} & D(A_n)
\end{tikzcd}
$$

T-translation

$$
\begin{tikzcd}
 & & A_0 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_1 \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, Rightarrow] & {} \arrow[d, mapsto] & A_n \\
A_0 \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(A_0) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(A_1) \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & \cdots \arrow[rr, rightsquigarrow] & {} \arrow[d, mapsto] & S(A_n) \\
D(A_0) \arrow[rr, rightsquigarrow] & {} & D(S(A_0)) \arrow[rr, rightsquigarrow] & {} & D(S(A_1)) \arrow[rr, rightsquigarrow] & {} & \cdots \arrow[rr, rightsquigarrow] & {} & D(S(A_n)) \arrow[d, rightsquigarrow] \\
 & & & & & & & & D(A_n)
\end{tikzcd}
$$

Remember

- $D(X)$ CNF formula: direct encoding of X
- $S(X)$ set of XOR constraints: split representation of X

Direct translations vs T-translations

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[r, Rightarrow] & A_1 \arrow[r, Rightarrow] & \cdots \arrow[r, Rightarrow] & A_{n-1} \arrow[r, Rightarrow] & A_n \\
D(A_0) \arrow[r, squiggly] & D(A_1) \arrow[r, squiggly] & \cdots \arrow[r, squiggly] & D(A_{n-1}) \arrow[r, squiggly] & D(A_n)
\end{tikzcd}
$$

T-translation

$$
\begin{tikzcd}
 & A_0 \arrow[r, Rightarrow] & A_1 \arrow[r, Rightarrow] & \cdots \arrow[r, Rightarrow] & A_n \\
A_0 \arrow[r, squiggly] & S(A_0) \arrow[r, squiggly] & S(A_1) \arrow[r, squiggly] & \cdots \arrow[r, squiggly] & S(A_n) \\
D(A_0) \arrow[r, squiggly] & D(S(A_0)) \arrow[r, squiggly] & D(S(A_1)) \arrow[r, squiggly] & \cdots \arrow[r, squiggly] & D(S(A_n)) \arrow[d, squiggly] \\
 & & & & D(A_n)
\end{tikzcd}
$$

Theorem direct translations are exponential in $|D(A_0)| + |D(A_n)|$

Direct translations vs T-translations

Direct translations

$$
\begin{tikzcd}
A_0 \arrow[r, Rightarrow, ""{name=U0, below}] & A_1 \arrow[r, Rightarrow, ""{name=U1, below}] & \cdots \arrow[r, Rightarrow, ""{name=U2, below}] & A_{n-1} \arrow[r, Rightarrow, ""{name=U3, below}] & A_n \\
D(A_0) \arrow[r, rightsquigarrow, ""{name=D0}] & D(A_1) \arrow[r, rightsquigarrow, ""{name=D1}] & \cdots \arrow[r, rightsquigarrow, ""{name=D2}] & D(A_{n-1}) \arrow[r, rightsquigarrow, ""{name=D3}] & D(A_n)
\arrow[mapsto, from=U0, to=D0]
\arrow[mapsto, from=U1, to=D1]
\arrow[mapsto, from=U2, to=D2]
\arrow[mapsto, from=U3, to=D3]
\end{tikzcd}
$$

T-translation

$$
\begin{tikzcd}
 & A_0 \arrow[r, Rightarrow, ""{name=T1, below}] & A_1 \arrow[r, Rightarrow, ""{name=T2, below}] & \cdots \arrow[r, Rightarrow, ""{name=T3, below}] & A_n \\
A_0 \arrow[r, rightsquigarrow, ""{name=M0}] & S(A_0) \arrow[r, rightsquigarrow, ""{name=M1}] & S(A_1) \arrow[r, rightsquigarrow, ""{name=M2}] & \cdots \arrow[r, rightsquigarrow, ""{name=M3}] & S(A_n) \\
D(A_0) \arrow[r, rightsquigarrow, ""{name=B0}] & D(S(A_0)) \arrow[r, rightsquigarrow, ""{name=B1}] & D(S(A_1)) \arrow[r, rightsquigarrow, ""{name=B2}] & \cdots \arrow[r, rightsquigarrow, ""{name=B3}] & D(S(A_n)) \arrow[d, rightsquigarrow] \\
 & & & & D(A_n)
\arrow[mapsto, from=T1, to=M1]
\arrow[mapsto, from=T2, to=M2]
\arrow[mapsto, from=T3, to=M3]
\arrow[mapsto, from=M0, to=B0]
\arrow[mapsto, from=M1, to=B1]
\arrow[mapsto, from=M2, to=B2]
\arrow[mapsto, from=M3, to=B3]
\end{tikzcd}
$$

Theorem direct translations are exponential in $|D(A_0)| + |D(A_n)|$

Theorem T-translations translations are polynomial in $|D(A_0)| + |D(A_n)|$

BDD-based approach

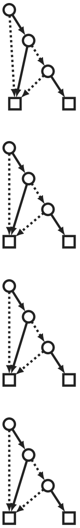
Proofs for BDD reasoning *Biere, Sinz (2006)*

C_1

C_2

C_3

C_4



BDD-based approach

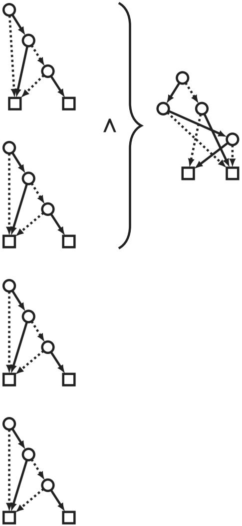
Proofs for BDD reasoning *Biere, Sinz (2006)*

C_1

C_2

C_3

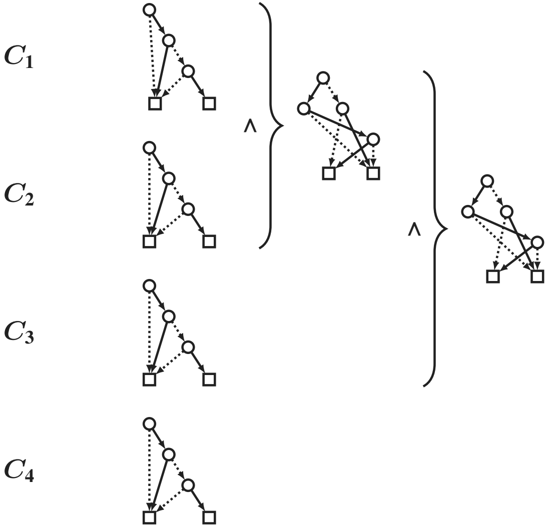
C_4



BDD-based approach

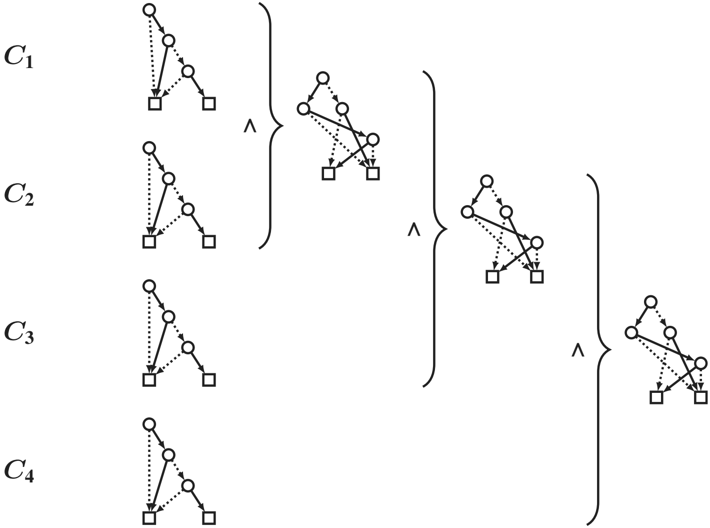
Proofs for BDD reasoning

Biere, Sinz (2006)



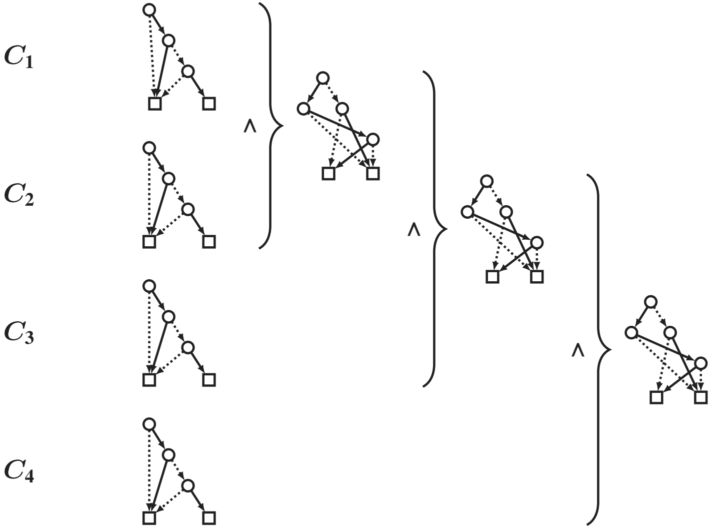
BDD-based approach

Proofs for BDD reasoning *Biere, Sinz (2006)*



BDD-based approach

Proofs for BDD reasoning *Biere, Sinz (2006)*



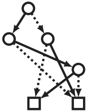
$B(Q)$ CNF encoding of BDD Q

BDD-based approach

Proofs for BDD reasoning *Biere, Sinz (2006)*

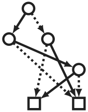
$C_1 \rightsquigarrow B(C_1)$

$\wedge$



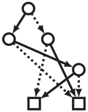
$C_2 \rightsquigarrow B(C_2)$

$\wedge$



$C_3 \rightsquigarrow B(C_3)$

$\wedge$



$C_4 \rightsquigarrow B(C_4)$

$B(Q)$ CNF encoding of BDD Q

BDD-based approach

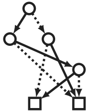
Proofs for BDD reasoning *Biere, Sinz (2006)*

$C_1 \rightsquigarrow B(C_1)$

$B(C_1 \land C_2)$

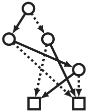
$C_2 \rightsquigarrow B(C_2)$

$\land$



$C_3 \rightsquigarrow B(C_3)$

$\land$



$C_4 \rightsquigarrow B(C_4)$

$B(Q)$ CNF encoding of BDD Q

BDD-based approach

Proofs for BDD reasoning *Biere, Sinz (2006)*

$C_1 \rightsquigarrow B(C_1)$

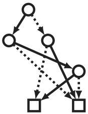
$B(C_1 \wedge C_2)$

$C_2 \rightsquigarrow B(C_2)$

$B(C_2 \wedge C_2 \wedge C_3)$

$C_3 \rightsquigarrow B(C_3)$

$\wedge$



$C_4 \rightsquigarrow B(C_4)$

$B(Q)$ CNF encoding of BDD Q

BDD-based approach

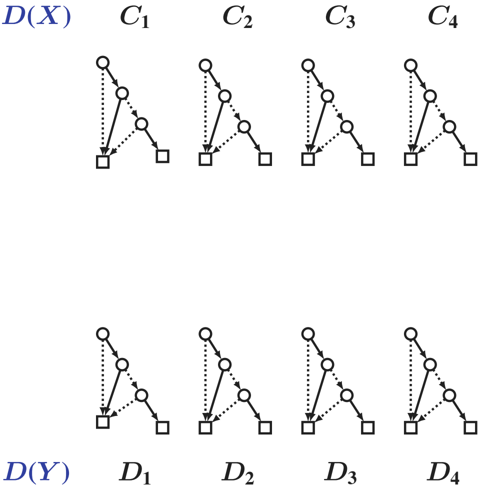
Proofs for BDD reasoning *Biere, Sinz (2006)*

$$
\begin{tikzcd}
C_1 \arrow[r, rightsquigarrow] & B(C_1) \arrow[dr, rightsquigarrow] & & & \\
& & B(C_1 \wedge C_2) \arrow[ddr, rightsquigarrow] & & \\
C_2 \arrow[r, rightsquigarrow] & B(C_2) \arrow[ur, rightsquigarrow] & & & \\
& & & B(C_2 \wedge C_2 \wedge C_3) \arrow[ddr, rightsquigarrow] & \\
C_3 \arrow[r, rightsquigarrow] & B(C_3) \arrow[urr, rightsquigarrow] & & & \\
& & & & B(C_1 \wedge C_2 \wedge C_3 \wedge C_4) \\
C_4 \arrow[r, rightsquigarrow] & B(C_4) \arrow[urrr, rightsquigarrow] & & &
\end{tikzcd}
$$

$B(Q)$ CNF encoding of BDD Q

Adaption to XOR reasoning

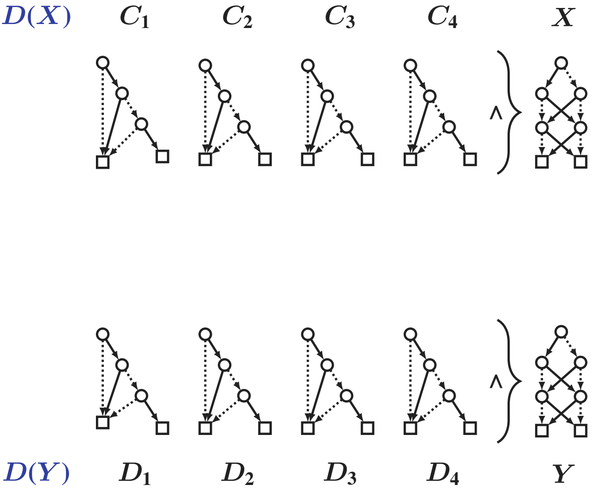
- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

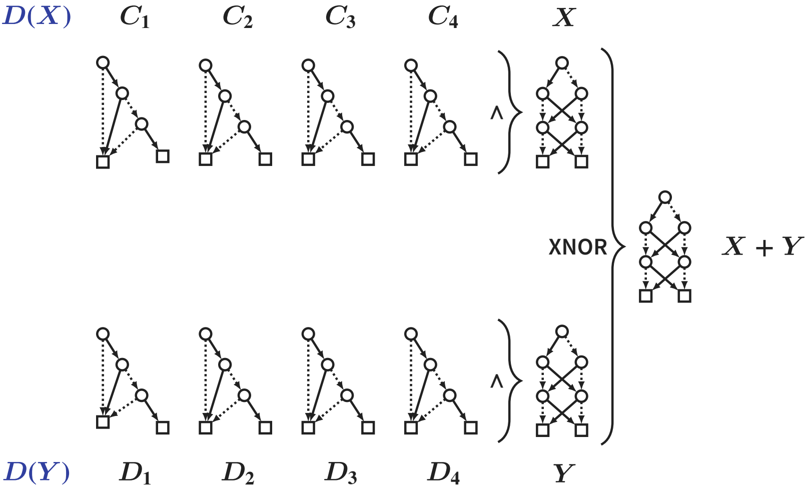
- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

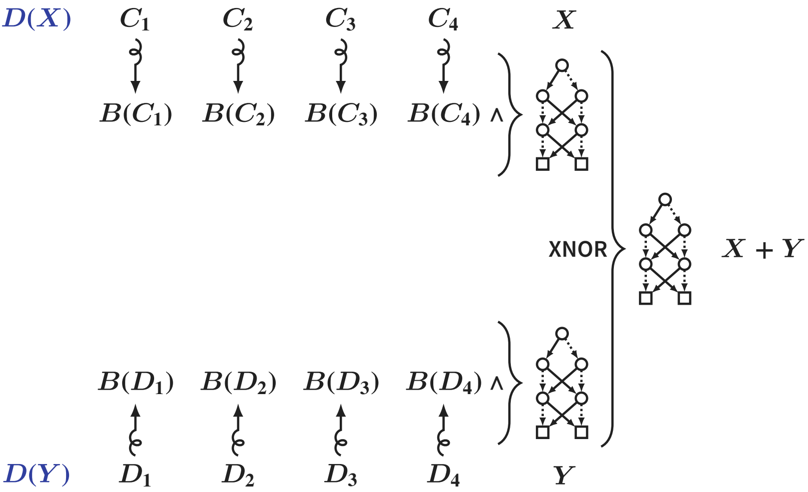
- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

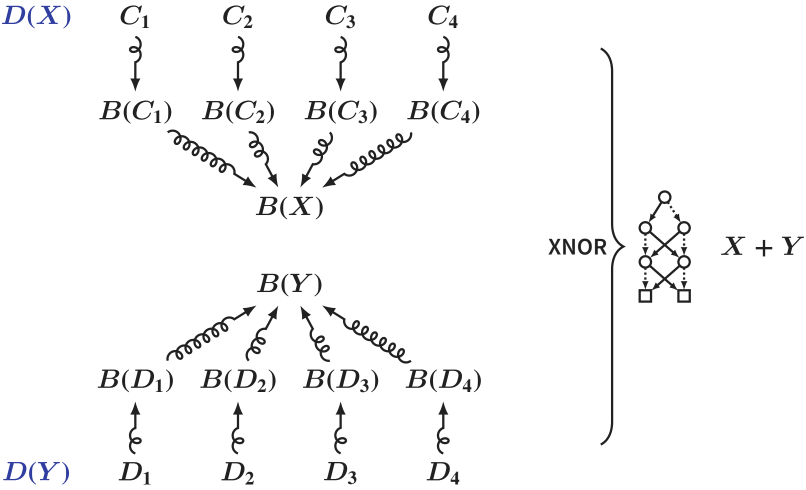
- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

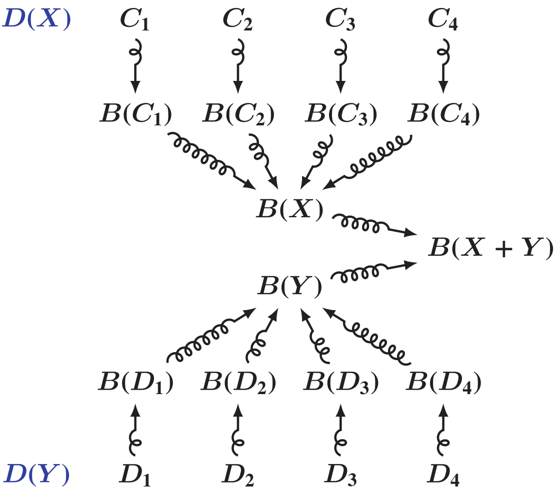
- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.



BDD-based approach

Adaption to XOR reasoning

- The previous method works for any binary operation f with $f(1,1) = 1$.
- Addition of XOR constraints corresponds to applying XNOR.