Minimizing Models for Tseitin-Encoded SAT Instances

Markus Iser, Carsten Sinz, Mana Taghdiri

KARLSRUHE INSTITUTE OF TECHNOLOGY (KIT)

What is Model Minimization?

Given a full model M we are looking for smaller "models" $M' \subseteq M$

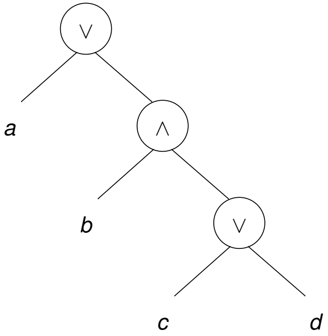
What is Model Minimization?

Given a full model M we are looking for smaller “models” $M' \subseteq M$

$$F = a \vee (b \wedge (c \vee d))$$

What is Model Minimization?

Given a full model M we are looking for smaller "models" $M' \subseteq M$

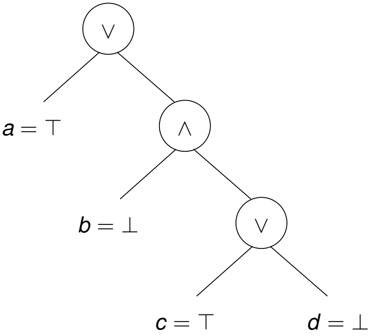


$$F = a \vee (b \wedge (c \vee d))$$

What is Model Minimization?

Given a full model M we are looking for smaller “models” $M' \subseteq M$

$$M = \{a, \neg b, c, \neg d\}$$

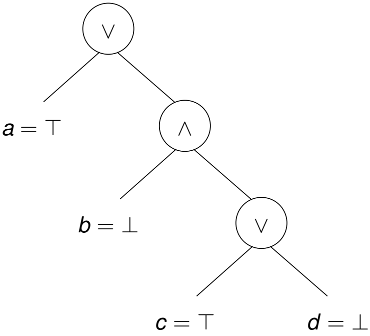


$$F = a \vee (b \wedge (c \vee d))$$

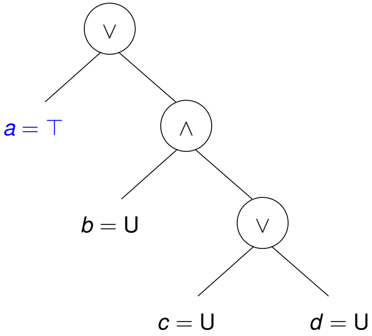
What is Model Minimization?

Given a full model M we are looking for smaller "models" $M' \subseteq M$

$$M = \{a, \neg b, c, \neg d\}$$



$$M' = \{a\}$$



$$F = a \vee (b \wedge (c \vee d))$$

Method Overview

Improved minimization of models for Tseitin-encoded formulas by reconstruction of original structure

First Step Naive minimization of models (Hitting Set Problem)

Second Step Minimize w.r.t. input variables

Third Step Reconstruct the original formula's structure and apply the procedure to a subset $F' \subseteq F_{cnf}$

Why minimize?

Shorter description of counterexamples help us to . . .

- boost abstraction-refinement loop in DPLL(T) based SMT solvers
- enhance usability of SAT-based verification tools
- boost model counting

Why minimize?

Karlsruhe Institute of Technology

Shorter description of counterexamples help us to . . .

- boost abstraction-refinement loop in DPLL(T) based SMT solvers
- enhance usability of SAT-based verification tools
- boost model counting

Why minimize?

Shorter description of counterexamples help us to . . .

- boost abstraction-refinement loop in DPLL(T) based SMT solvers
- enhance usability of SAT-based verification tools
- boost model counting

Why minimize?

Karlsruhe Institute of Technology

Shorter description of counterexamples help us to ...

- boost abstraction-refinement loop in DPLL(T) based SMT solvers
- enhance usability of SAT-based verification tools
- boost model counting

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Purification: remove all unsatisfied literals from formula

$p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

$M' = \{a, c, t_0, \neg t_1, t_2\}$

Minimizing Models for CNF Formulas

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Determine subset $M' \subseteq M$ such that each clause is satisfied

Purification: remove all unsatisfied literals from formula

$p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

$M' = \{a, c, t_0, \neg t_1, t_2\}$

Compute a minimal Hitting Set with SAT

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t_1'\}, \{t_1', t_2\}, \{c\}\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t_1'\}, \{t_1', t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t_1', \neg t_2, \neg a, \neg c\}\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t'_1\}, \{t'_1, t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t'_1, \neg t_2, \neg a, \neg c\}\}$
$M_1 = \{t_0, t'_1, \neg t_2, a, c\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t'_1\}, \{t'_1, t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t'_1, \neg t_2, \neg a, \neg c\}\}$
$M_1 = \{t_0, t'_1, \neg t_2, a, c\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_2\}\} \cup \{\{\neg t_0, \neg t'_1, \neg a, \neg c\}\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t'_1\}, \{t'_1, t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t'_1, \neg t_2, \neg a, \neg c\}\}$
$M_1 = \{t_0, t'_1, \neg t_2, a, c\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_2\}\} \cup \{\{\neg t_0, \neg t'_1, \neg a, \neg c\}\}$
$\emptyset$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t_1'\}, \{t_1', t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t_1', \neg t_2, \neg a, \neg c\}\}$
$M_1 = \{t_0, t_1', \neg t_2, a, c\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_2\}\} \cup \{\{\neg t_0, \neg t_1', \neg a, \neg c\}\}$
$\emptyset$

Denormalize M_1: $M'' = \{a, c, t_0, \neg t_1\}$

Computing a Minimal Hitting Set with SAT

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$

Purification: $p(F_{cnf}) = \{\{t_0\}, \{a\}, \{\neg t_1\}, \{\neg t_1, t_2\}, \{c\}\}$

Normalization: $n(F_{cnf}) = \{\{t_0\}, \{a\}, \{t'_1\}, \{t'_1, t_2\}, \{c\}\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_0, \neg t'_1, \neg t_2, \neg a, \neg c\}\}$

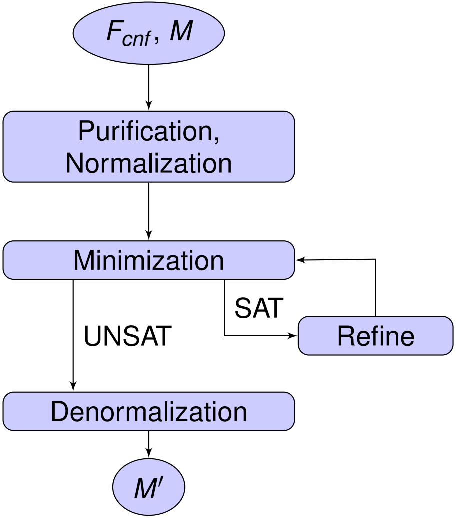
$M_1 = \{t_0, t'_1, \neg t_2, a, c\}$

Solve: $G = n(F_{cnf}) \cup \{\{\neg t_2\}\} \cup \{\{\neg t_0, \neg t'_1, \neg a, \neg c\}\}$

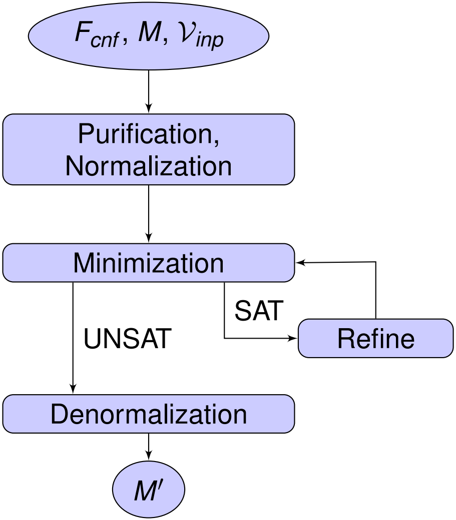
$\emptyset$

Denormalize M_1: $M'' = \{a, c, t_0, \neg t_1\}$

Algorithm Overview



Algorithm Overview



The Effect of the Tseitin Encoding on Model Size

Karlsruhe Institute of Technology

$F = a \vee (b \wedge (c \vee d))$

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

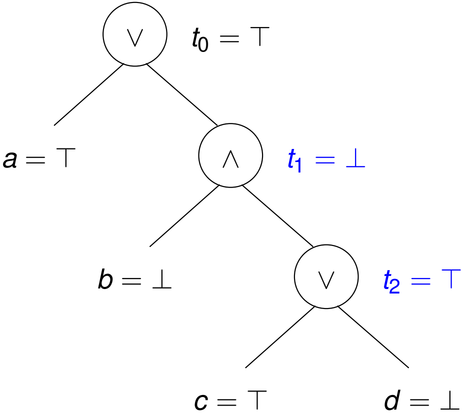
Full Model:	$M = \{a, \neg b, c, \neg d, t_0, \neg t_1, t_2\}$
Hitting Set:	$M' = \{a, c, t_0, \neg t_1\}$
Input Variables:	$M'' = \{a, c\}$

- What happened to our shorter model $\{a\}$ for the original formula?
- Is there a way to obtain that shorter model from F_{cnf}?

Analyze Assignment of Encoding Variables

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$



$t_0 \rightarrow a \vee t_1$

$t_1 \rightarrow b \wedge t_2$

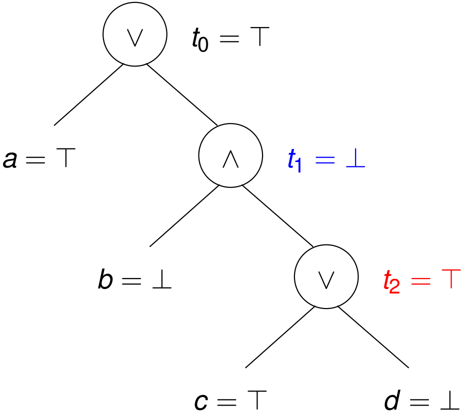
$t_2 \rightarrow c \vee d$

Analyze Assignment of Encoding Variables

Karlsruhe Institute of Technology

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$



$t_0 \rightarrow a \vee t_1$

$t_1 \rightarrow b \wedge t_2$

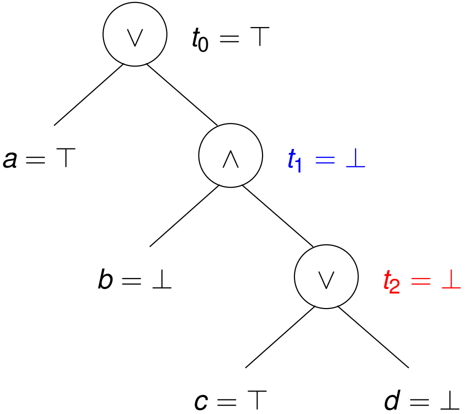
$t_2 \rightarrow c \vee d$

Analyze Assignment of Encoding Variables

Karlsruhe Institute of Technology

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$



$t_0 \rightarrow a \vee t_1$

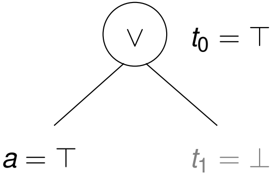
$t_1 \rightarrow b \wedge t_2$

$t_2 \rightarrow c \vee d$

Analyze Assignment of Encoding Variables

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$

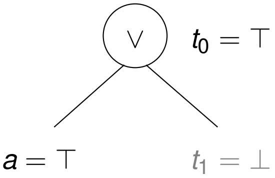


$t_0 \rightarrow a \vee t_1$

Analyze Assignment of Encoding Variables

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$

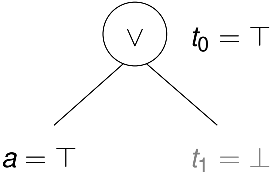


$t_0 \rightarrow a \vee t_1$

Analyze Assignment of Encoding Variables

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$M = \{t_0, a, \neg t_1, \neg b, t_2, c, \neg d\}$



$t_0 \rightarrow a \vee t_1$

We can reconstruct the original formula's structure

Reconstructing the original formula structure

$$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}, \quad \text{root} = t_0$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication

t_0

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication

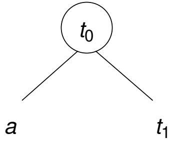
t_0 $\{\neg t_0, a, t_1\}$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication



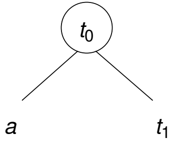
$\{\neg t_0, a, t_1\}$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}, \quad \text{root} = t_0$

Follow the Implication



$\{\neg t_0, a, t_1\}$

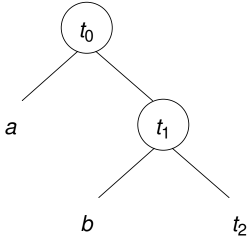
$\{\neg t_1, b\}, \{\neg t_1, t_2\}$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication



$\{\neg t_0, a, t_1\}$

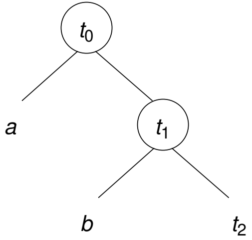
$\{\neg t_1, b\}, \{\neg t_1, t_2\}$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication



$\{\neg t_0, a, t_1\}$

$\{\neg t_1, b\}, \{\neg t_1, t_2\}$

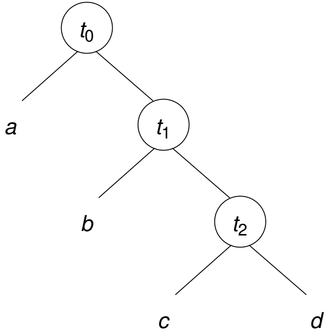
$\{\neg t_2, c, d\}$

Reconstructing the original formula structure

$F_{cnf} = \{\{t_0\}, \{\neg t_0, a, t_1\}, \{\neg t_1, b\}, \{\neg t_1, t_2\}, \{\neg t_2, c, d\}\}$

$\mathcal{V}_{inp} = \{a, b, c, d\}$, root = t_0

Follow the Implication

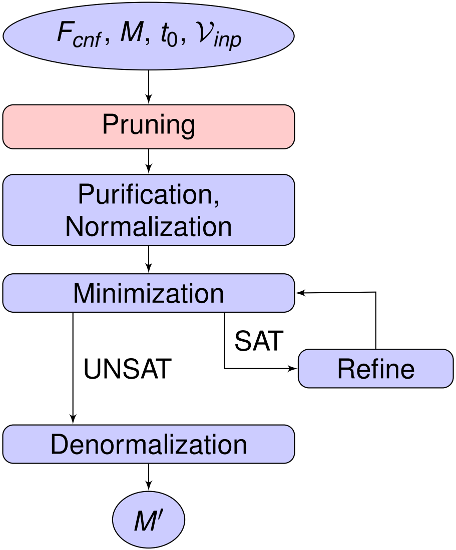


$\{\neg t_0, a, t_1\}$

$\{\neg t_1, b\}, \{\neg t_1, t_2\}$

$\{\neg t_2, c, d\}$

Pruning as a Preprocessing Step



Current/Future Research

- Consider different encodings (e.g. only partially Tseitin)
- Model Counting (with an application to Quantitative Information Flow Analysis)

Current/Future Research

Karlsruhe Institute of Technology

- Consider different encodings (e.g. only partially Tseitin)
- Model Counting (with an application to Quantitative Information Flow Analysis)

Current/Future Research

- Consider different encodings (e.g. only partially Tseitin)
- Model Counting (with an application to Quantitative Information Flow Analysis)

Current/Future Research

- Consider different encodings (e.g. only partially Tseitin)
- Model Counting (with an application to Quantitative Information Flow Analysis)