

# Exploitation of dynamic symmetries for solving SAT problems

Doctorat de Sorbonne Université

Hakan METIN

Le, 18 décembre 2019

## Rapporteurs:

PASCAL FONTAINE  
LAURE PETRUCCI

Professeur, Université de Liège  
Professeur, Université Paris 13

## Examineurs:

BART BOGAERTS  
JEAN-MICHEL COUVREUR  
EMMANUELLE ENCRENAZ

Assistant Professor, Vrije Universiteit Brussel  
Professeur, Université d'Orléans  
Maître de conférences, Sorbonne Université

## Directeurs:

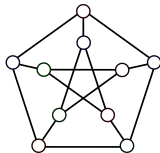
SOUHEIB BAARIR  
FABRICE KORDON

Maître de conférences, Université Paris Nanterre  
Professeur, Sorbonne Université

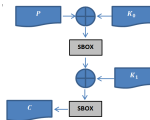


# Motivation

Graph coloring



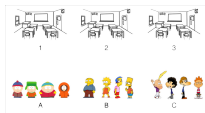
Cryptanalysis



Hardware verification

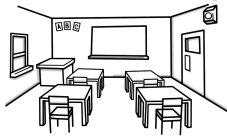


Planning

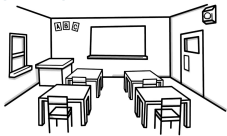


Boolean  
SATisfiability

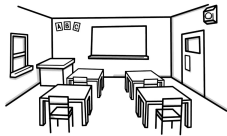
## SAT: an example (1/2)



1



2



3



A



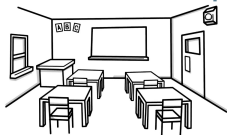
B



C

Is it possible to attribute each group to a unique classroom?

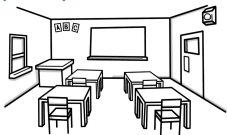
## SAT: an example (1/2)



1  
↑



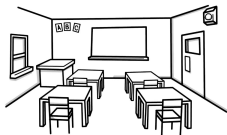
A



2  
↑



B



3  
↑

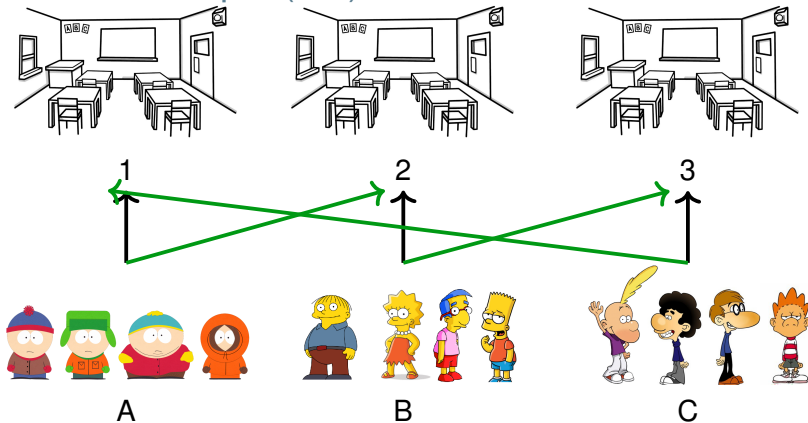


C

Is it possible to attribute each group to a unique classroom?

YES! SATisfiable  $\alpha = (A, 1), (B, 2), (C, 3)$

## SAT: an example (1/2)

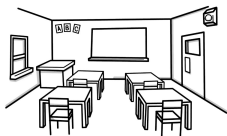
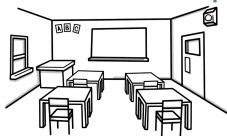


Is it possible to attribute each group to a unique classroom?

YES! SATisfiable  $\alpha = (A, 1), (B, 2), (C, 3)$

Many solutions  $\alpha' = (A, 2), (B, 3), (C, 1)$

# SAT: an example (1/2)



A



B



C

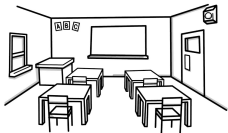
Is it possible to attribute each group to a unique classroom?

YES! SATisfiable  $\alpha = (A, 1), (B, 2), (C, 3)$

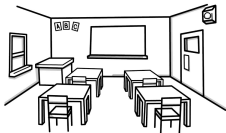
Many solutions  $\alpha' = (A, 2), (B, 3), (C, 1)$

$\alpha'' = \dots$

## SAT: an example (2/2)



1



2



A



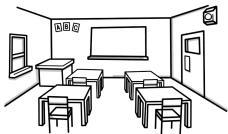
B



C

Is it possible to attribute each group to a unique classroom?

## SAT: an example (2/2)



1



2



A



B



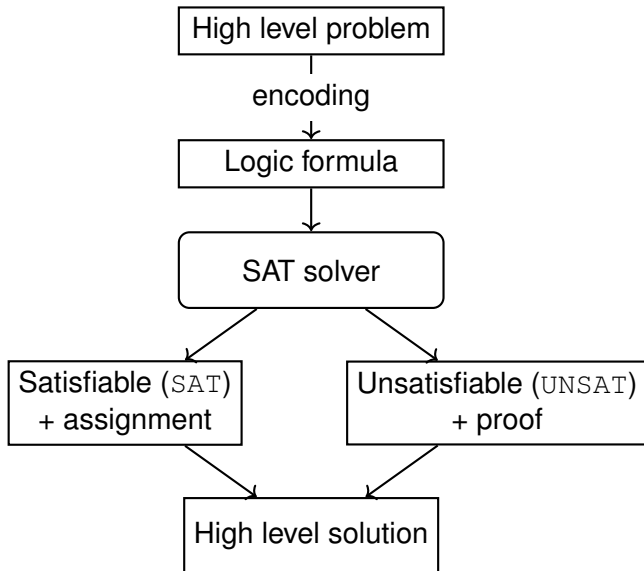
C

Is it possible to attribute each group to a unique classroom?

No! UNSATISFIABLE



# From high level problem to the solution through SAT solving



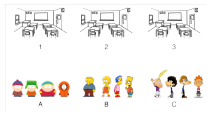
# Encoding the problem

$$\begin{array}{c} \overbrace{(A, 1)}^{x_1} \overbrace{(A, 2)}^{x_2} \overbrace{(A, 3)}^{x_3} \\ (B, 1)(B, 2)(B, 3) \\ (C, 1)(C, 2)(C, 3) \end{array}$$

$$(x_1 \vee x_2 \vee x_3) \wedge$$

$$(x_4 \vee x_5 \vee x_6) \wedge$$

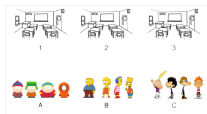
$$(x_7 \vee x_8 \vee x_9) \wedge$$



# Encoding the problem

$$\begin{array}{c} \overbrace{(A, 1)}^{x_1} \overbrace{(A, 2)}^{x_2} \overbrace{(A, 3)}^{x_3} \\ (B, 1)(B, 2)(B, 3) \\ (C, 1)(C, 2)(C, 3) \end{array}$$

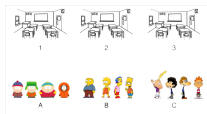
$$\begin{array}{l} \neg(A, 1) \neg(B, 1) \\ \neg(A, 1) \neg(C, 1) \\ \neg(B, 1) \neg(C, 1) \end{array}$$



$$\begin{array}{l} (x_1 \vee x_2 \vee x_3) \wedge \\ (x_4 \vee x_5 \vee x_6) \wedge \\ (x_7 \vee x_8 \vee x_9) \wedge \end{array}$$

$$\begin{array}{l} (\neg x_1 \vee \neg x_4) \wedge \\ (\neg x_1 \vee \neg x_7) \wedge \\ (\neg x_4 \vee \neg x_7) \wedge \end{array}$$

# Encoding the problem



$$\begin{matrix} x_1 & x_2 & x_3 \\ \overbrace{(A, 1)} & \overbrace{(A, 2)} & \overbrace{(A, 3)} \\ \overbrace{(B, 1)} & \overbrace{(B, 2)} & \overbrace{(B, 3)} \\ \overbrace{(C, 1)} & \overbrace{(C, 2)} & \overbrace{(C, 3)} \end{matrix}$$

$$\neg(A, 1) \neg(B, 1)$$

$$\neg(A, 1) \neg(C, 1)$$

$$\neg(B, 1) \neg(C, 1)$$

$$\neg(A, 2) \neg(B, 2)$$

$$\neg(A, 2) \neg(C, 2)$$

$$\neg(B, 2) \neg(C, 2)$$

$$\neg(A, 3) \neg(B, 3)$$

$$\neg(A, 3) \neg(C, 3)$$

$$\neg(B, 3) \neg(C, 3)$$

$$(x_1 \vee x_2 \vee x_3) \wedge$$

$$(x_4 \vee x_5 \vee x_6) \wedge$$

$$(x_7 \vee x_8 \vee x_9) \wedge$$

$$(\neg x_1 \vee \neg x_4) \wedge$$

$$(\neg x_1 \vee \neg x_7) \wedge$$

$$(\neg x_4 \vee \neg x_7) \wedge$$

$$(\neg x_2 \vee \neg x_5) \wedge$$

$$(\neg x_2 \vee \neg x_8) \wedge$$

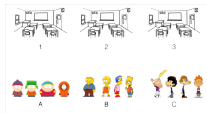
$$(\neg x_5 \vee \neg x_8) \wedge$$

$$(\neg x_3 \vee \neg x_6) \wedge$$

$$(\neg x_3 \vee \neg x_9) \wedge$$

$$(\neg x_6 \vee \neg x_9)$$

# Encoding the problem



$$\begin{matrix} x_1 & x_2 & x_3 \\ \overbrace{(A, 1)} & \overbrace{(A, 2)} & \overbrace{(A, 3)} \\ \overbrace{(B, 1)} & \overbrace{(B, 2)} & \overbrace{(B, 3)} \\ \overbrace{(C, 1)} & \overbrace{(C, 2)} & \overbrace{(C, 3)} \end{matrix}$$

$$\neg(A, 1) \neg(B, 1)$$

$$\neg(A, 1) \neg(C, 1)$$

$$\neg(B, 1) \neg(C, 1)$$

$$\neg(A, 2) \neg(B, 2)$$

$$\neg(A, 2) \neg(C, 2)$$

$$\neg(B, 2) \neg(C, 2)$$

$$\neg(A, 3) \neg(B, 3)$$

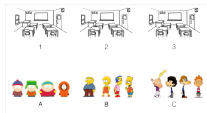
$$\neg(A, 3) \neg(C, 3)$$

$$\neg(B, 3) \neg(C, 3)$$

Clause

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge \\ & (x_4 \vee x_5 \vee x_6) \wedge \\ & (x_7 \vee x_8 \vee x_9) \wedge \\ & (\neg x_1 \vee \neg x_4) \wedge \\ & (\neg x_1 \vee \neg x_7) \wedge \\ & (\neg x_4 \vee \neg x_7) \wedge \\ & (\neg x_2 \vee \neg x_5) \wedge \\ & (\neg x_2 \vee \neg x_8) \wedge \\ & (\neg x_5 \vee \neg x_8) \wedge \\ & (\neg x_3 \vee \neg x_6) \wedge \\ & (\neg x_3 \vee \neg x_9) \wedge \\ & (\neg x_6 \vee \neg x_9) \end{aligned}$$

# Encoding the problem



$$\begin{matrix} x_1 & x_2 & x_3 \\ \overbrace{(A, 1)} & \overbrace{(A, 2)} & \overbrace{(A, 3)} \\ \overbrace{(B, 1)} & \overbrace{(B, 2)} & \overbrace{(B, 3)} \\ \overbrace{(C, 1)} & \overbrace{(C, 2)} & \overbrace{(C, 3)} \end{matrix}$$

$$\neg(A, 1) \neg(B, 1)$$

$$\neg(A, 1) \neg(C, 1)$$

$$\neg(B, 1) \neg(C, 1)$$

$$\neg(A, 2) \neg(B, 2)$$

$$\neg(A, 2) \neg(C, 2)$$

$$\neg(B, 2) \neg(C, 2)$$

$$\neg(A, 3) \neg(B, 3)$$

$$\neg(A, 3) \neg(C, 3)$$

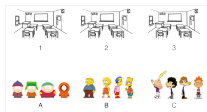
$$\neg(B, 3) \neg(C, 3)$$

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge \\ & (x_4 \vee x_5 \vee x_6) \wedge \\ & (x_7 \vee x_8 \vee x_9) \wedge \\ & (\neg x_1 \vee \neg x_4) \wedge \\ & (\neg x_1 \vee \neg x_7) \wedge \\ & (\neg x_4 \vee \neg x_7) \wedge \\ & (\neg x_2 \vee \neg x_5) \wedge \\ & (\neg x_2 \vee \neg x_8) \wedge \\ & (\neg x_5 \vee \neg x_8) \wedge \\ & (\neg x_3 \vee \neg x_6) \wedge \\ & (\neg x_3 \vee \neg x_9) \wedge \\ & (\neg x_6 \vee \neg x_9) \end{aligned}$$

Clause

Conjunctive  
Normal  
Form (CNF)

# Encoding the problem



$$\begin{matrix} x_1 & x_2 & x_3 \\ \overbrace{(A, 1)} & \overbrace{(A, 2)} & \overbrace{(A, 3)} \\ \overbrace{(B, 1)} & \overbrace{(B, 2)} & \overbrace{(B, 3)} \\ \overbrace{(C, 1)} & \overbrace{(C, 2)} & \overbrace{(C, 3)} \end{matrix}$$

$$\neg(A, 1) \neg(B, 1)$$

$$\neg(A, 1) \neg(C, 1)$$

$$\neg(B, 1) \neg(C, 1)$$

$$\neg(A, 2) \neg(B, 2)$$

$$\neg(A, 2) \neg(C, 2)$$

$$\neg(B, 2) \neg(C, 2)$$

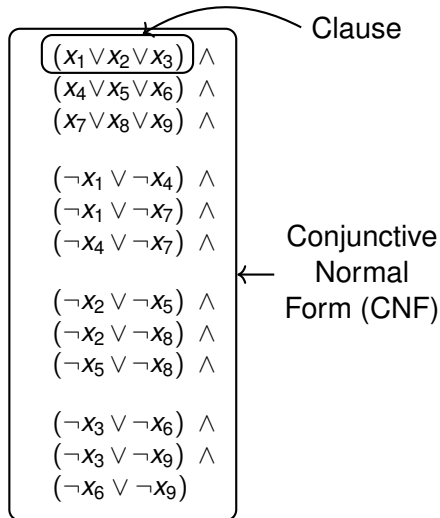
$$\neg(A, 3) \neg(B, 3)$$

$$\neg(A, 3) \neg(C, 3)$$

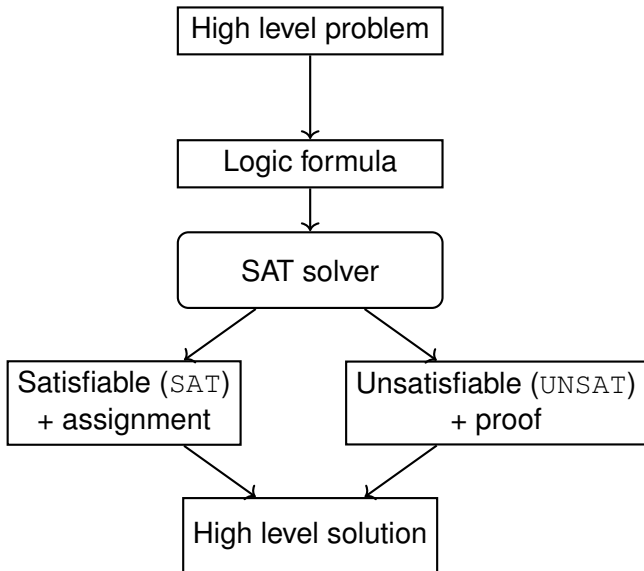
$$\neg(B, 3) \neg(C, 3)$$

Clause represented as a set:

$$(x_1 \vee x_2 \vee x_3) \rightarrow \{x_1, x_2, x_3\}$$

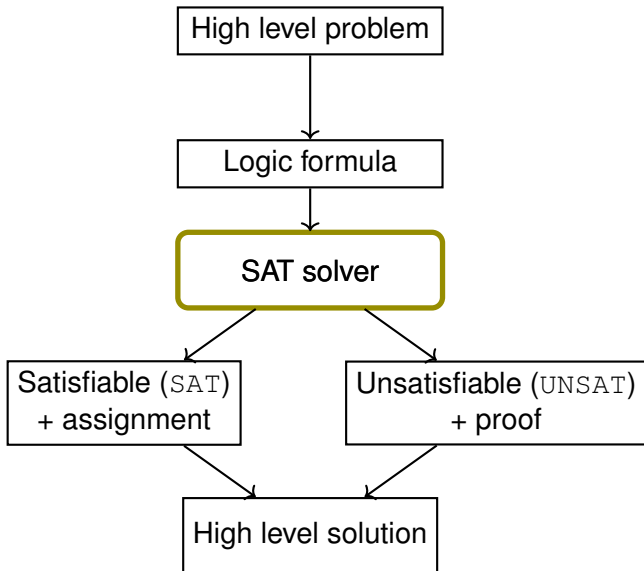


# From high level problem to the solution through SAT solving





# From high level problem to the solution through SAT solving



# SAT Solving

Solving SAT formula is known to be **NP-complete** [Coo71]

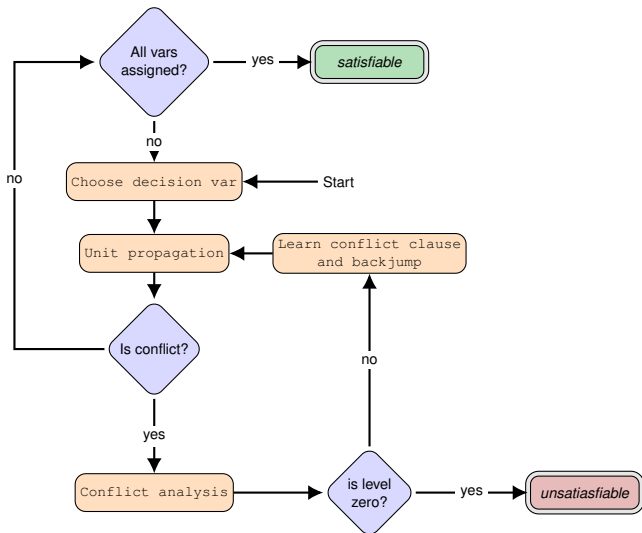
Good performance in practice:

- Handle large problem (million variables and clauses)
- International SAT competition each year on academic and industrial problems

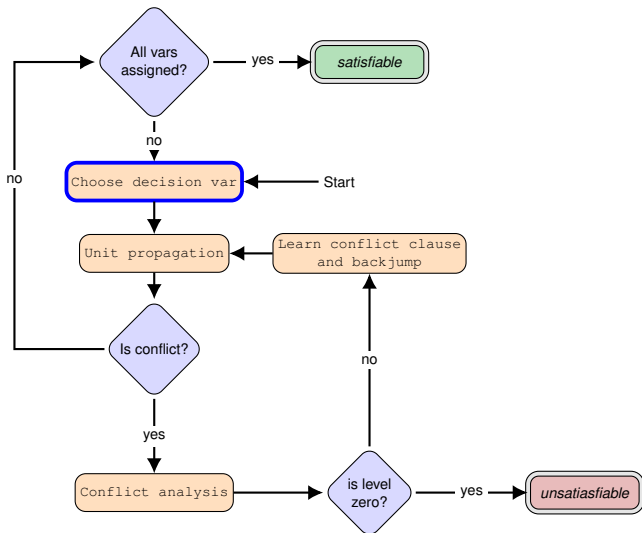
Enumerative algorithms:

- Davis, Putnam, Logemann, and Loveland (DPLL) [DLL62]
  - Boolean Constraint Propagation (BCP)
- **Conflict Driven Clause Learning** (CDCL) [MSS99]
  - Derived from DPLL
  - Clause learning

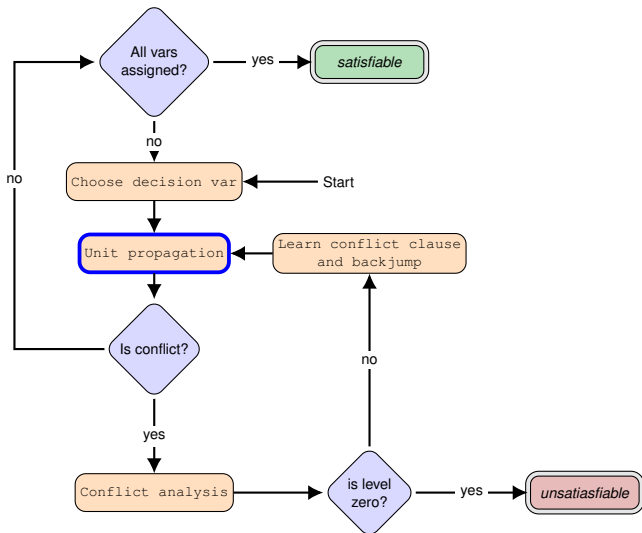
# CDCL in detail



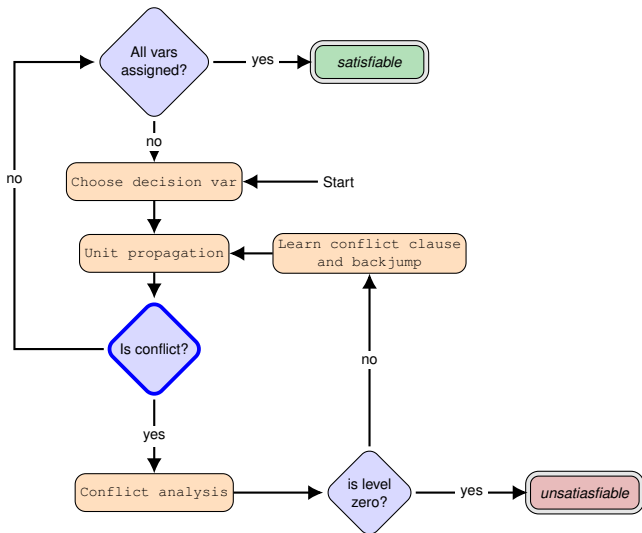
# CDCL in detail



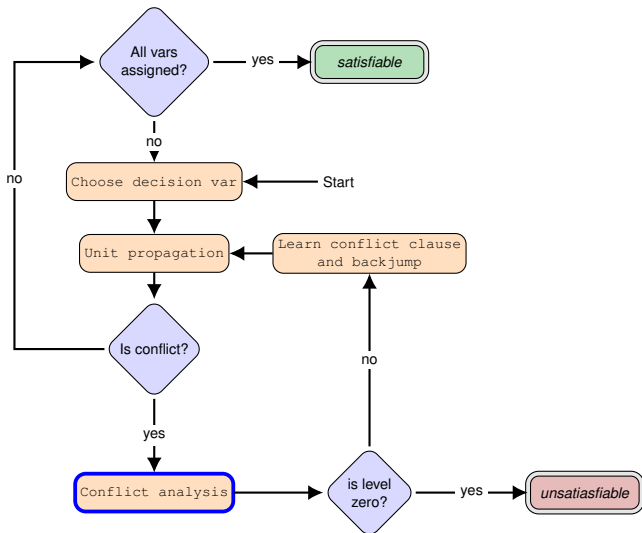
# CDCL in detail



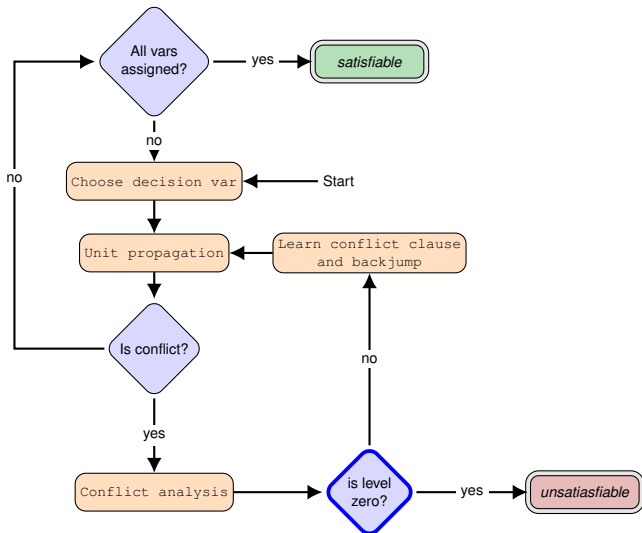
# CDCL in detail



# CDCL in detail

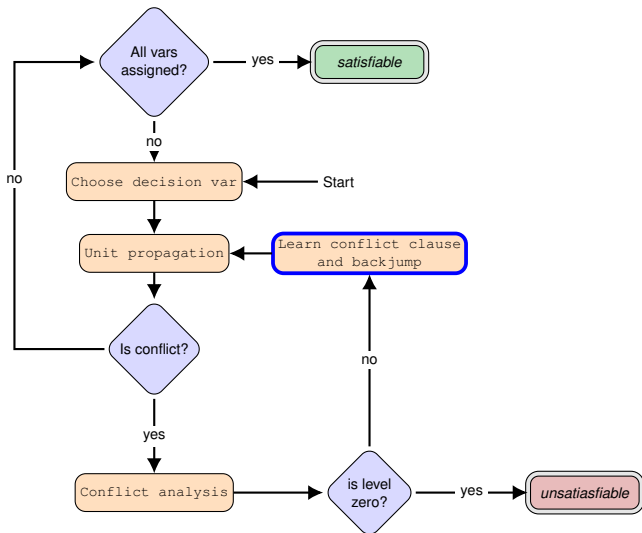


# CDCL in detail

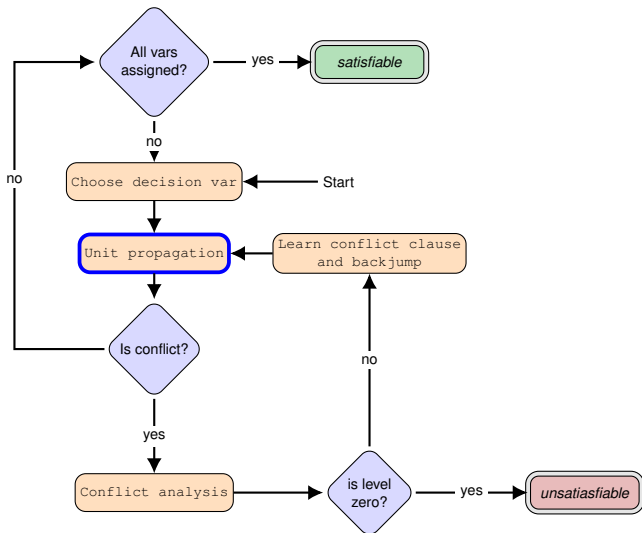




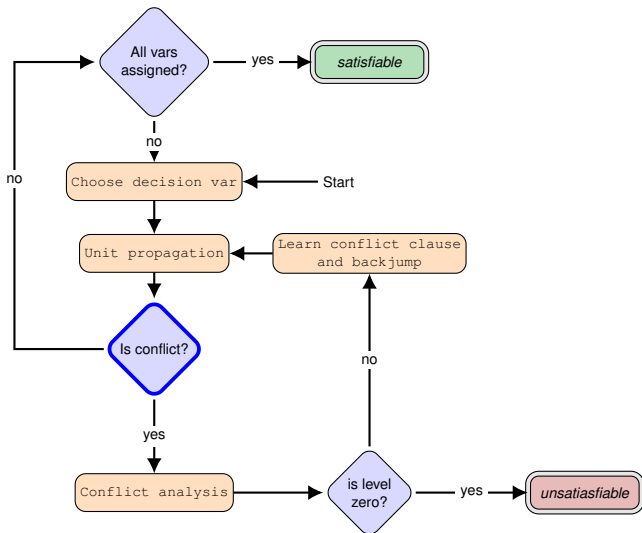
# CDCL in detail



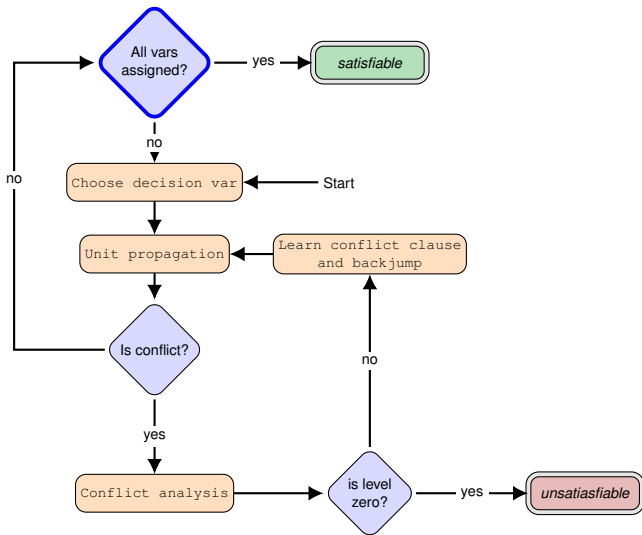
# CDCL in detail



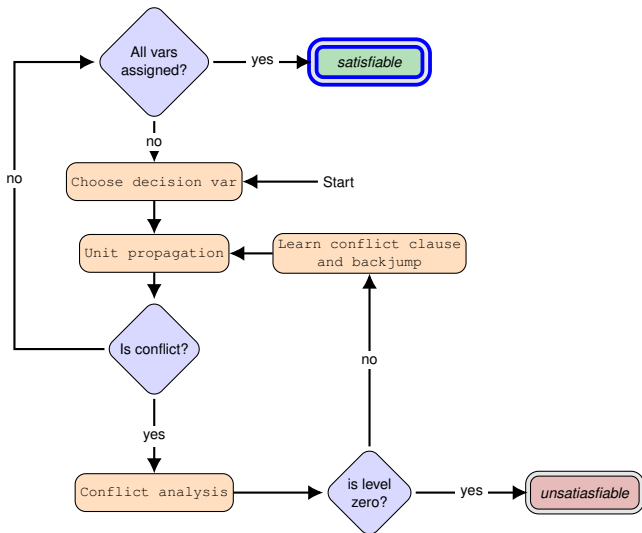
# CDCL in detail



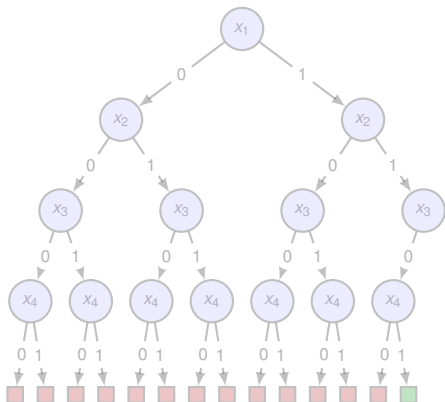
# CDCL in detail



# CDCL in detail



# CDCL in action



○ decision    □ propagation

$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

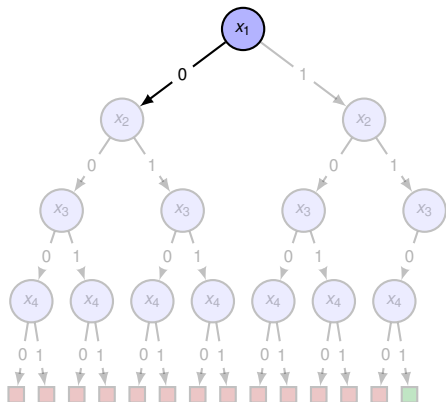
$$\omega_6 = \{x_3, x_4\}$$

$$\alpha = \{\}$$

# CDCL in action

Choose decision var

Unit Propagation



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{\mathbf{x}_1, \neg x_4\}$$

$$\omega_3 = \{\mathbf{x}_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

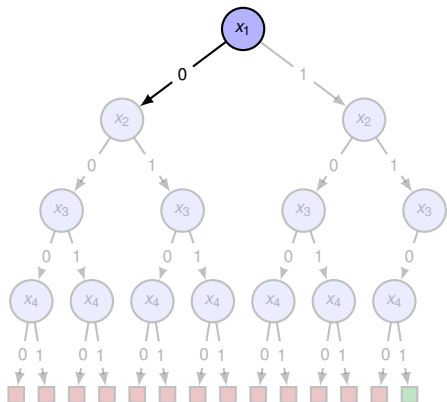
$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

$$\alpha = \{\neg x_1\}$$

# CDCL in action

Conflict Analysis



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{\mathbf{x}_1, \neg \mathbf{x}_4\}$$

$$\omega_3 = \{\mathbf{x}_1, \mathbf{x}_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

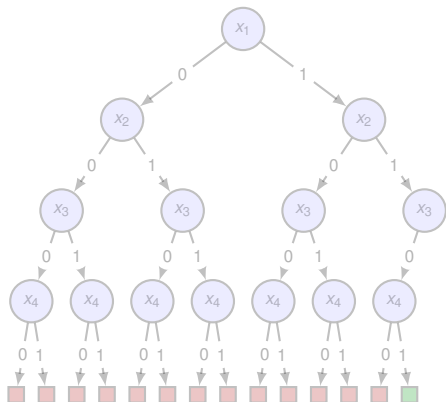
$$\omega_6 = \{x_3, x_4\}$$

$$\alpha = \{\neg \mathbf{x}_1\}$$



# CDCL in action

Learn conflict clause  
and backjump



○ decision    □ propagation

$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

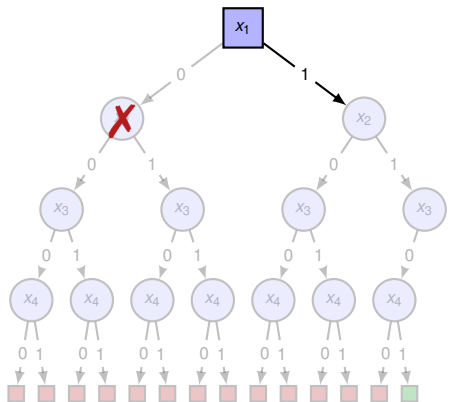
$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

$$\alpha = \{\}$$

# CDCL in action

Unit Propagation



$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

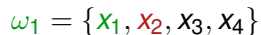
$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

$$\alpha = \{x_1\}$$

```
graph LR; A[Choose decision var] --> B[Unit Propagation]
```



$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

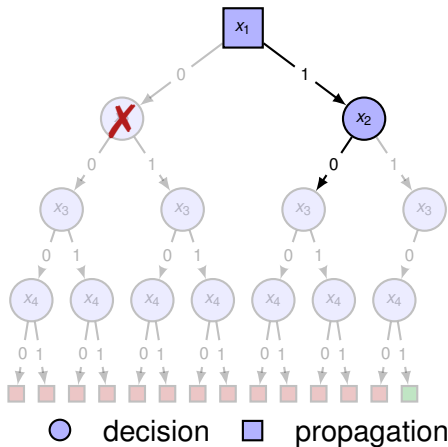
$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

$$\alpha = \{X_1, \neg X_2\}$$

# CDCL in action



$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

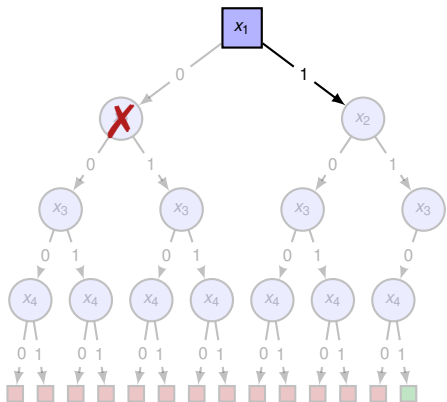
$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

$$\alpha = \{x_1, \neg x_2\}$$

## CDCL in action

Learn conflict clause  
and backjump



○ decision    □ propagation

$$\alpha = \{X_1\}$$

$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

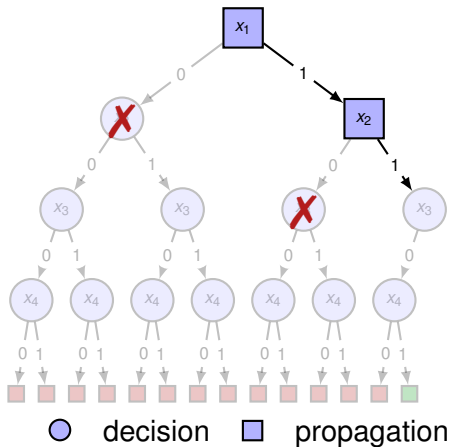
$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

$$\omega_8 = \{x_2\}$$

# CDCL in action

Unit Propagation



$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

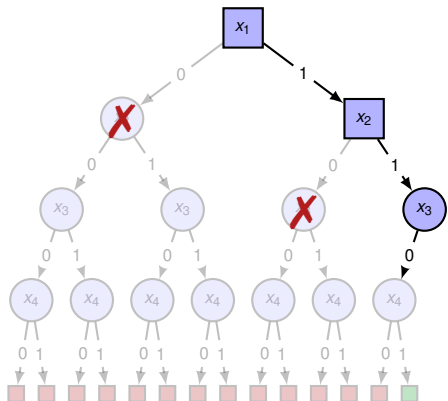
$$\omega_8 = \{x_2\}$$

$$\alpha = \{x_1, x_2\}$$

# CDCL in action

Choose decision var

Unit Propagation



○ decision    □ propagation

$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{x_1, x_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

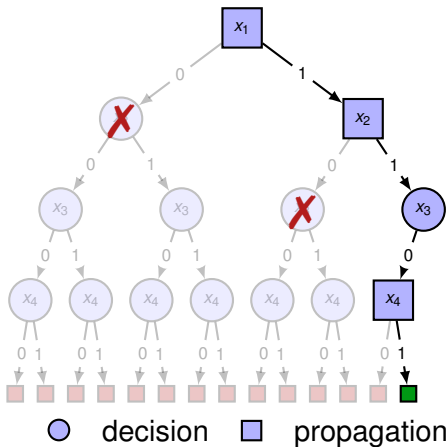
$$\omega_7 = \{x_1\}$$

$$\omega_8 = \{x_2\}$$

$$\alpha = \{x_1, x_2, \neg x_3\}$$

## CDCL in action

*satisfiable*



$$\omega_1 = \{x_1, x_2, x_3, x_4\}$$

$$\omega_2 = \{x_1, \neg x_4\}$$

$$\omega_3 = \{X_1, X_4\}$$

$$\omega_4 = \{x_2, \neg x_4\}$$

$$\omega_5 = \{x_2, x_4\}$$

$$\omega_6 = \{x_3, x_4\}$$

$$\omega_7 = \{x_1\}$$

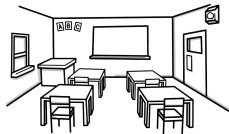
$$\omega_8 = \{x_2\}$$

$$\alpha = \{x_1, x_2, \neg x_3, x_4\}$$

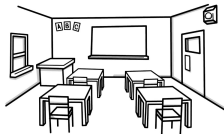


# SAT and symmetries

# Presence of symmetries



1



2



A

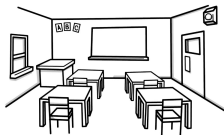
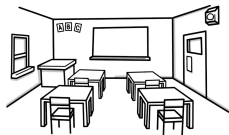


B



C

# Presence of symmetries



A

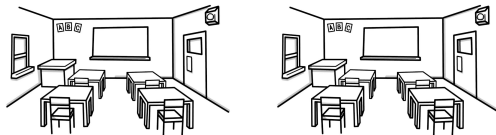


B



C

# Presence of symmetries



1 ← → 2



A

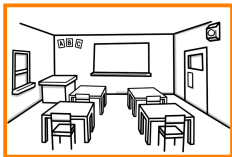


B

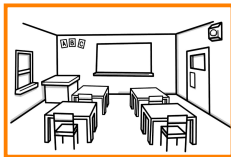


C

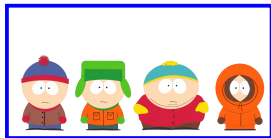
# Presence of symmetries



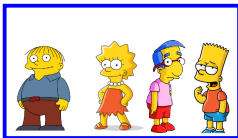
1



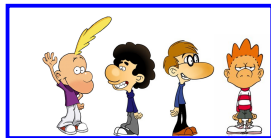
2



A

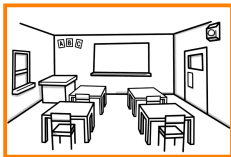


B

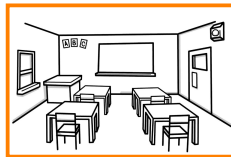


C

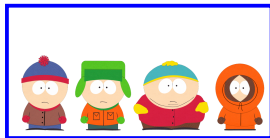
# Presence of symmetries



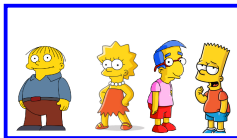
$\neq 2$



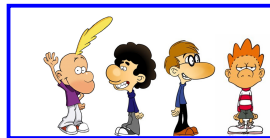
$\neq 1$



A



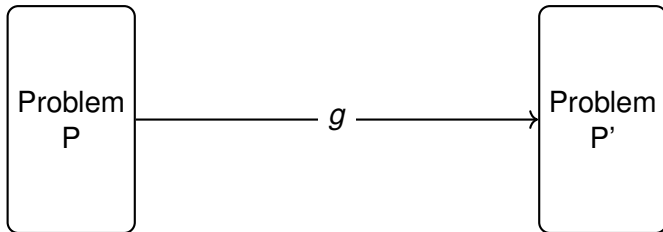
B C



$\in B$

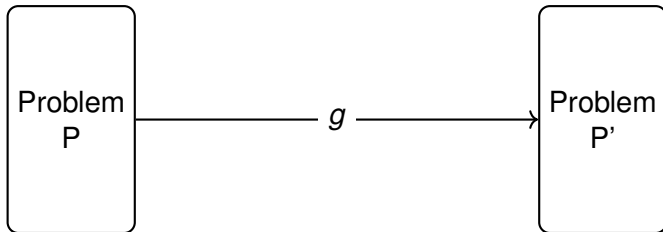
# Symmetry in high level

Let  $g$  a symmetry



# Symmetry in high level

Let  $g$  a symmetry



Equi-satisfiability

$$\text{solution} \models P \Leftrightarrow g.\text{solution} \models P'$$



# Syntactic symmetry

A symmetry (permutation)  $g$  is a bijective function (on variables) that leaves the formula  $\varphi$  invariant

# Syntactic symmetry

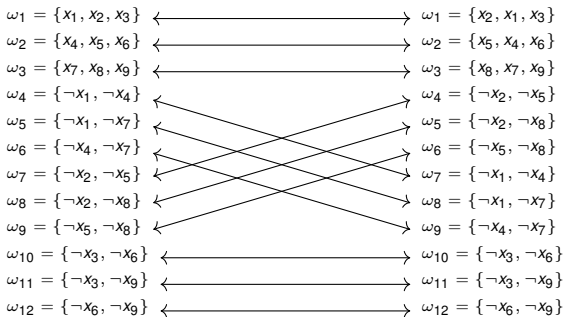
A symmetry (permutation)  $g$  is a bijective function (on variables) that leaves the formula  $\varphi$  invariant

$$g = \left( \begin{array}{cccccccccc} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 \\ x_2 & x_1 & x_3 & x_5 & x_4 & x_6 & x_8 & x_7 & x_9 \end{array} \right) \rightarrow (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

# Syntactic symmetry

A symmetry (permutation)  $g$  is a bijective function (on variables) that leaves the formula  $\varphi$  invariant

$$g = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 \\ x_2 & x_1 & x_3 & x_5 & x_4 & x_6 & x_8 & x_7 & x_9 \end{pmatrix} \rightarrow (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$



# Computing symmetries of a SAT problem

*CNF formula*

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6) \wedge (x_7 \vee x_8 \vee x_9) \\ & \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_1 \vee \neg x_7) \wedge (\neg x_4 \vee \neg x_7) \\ & \wedge (\neg x_2 \vee \neg x_5) \wedge (\neg x_2 \vee \neg x_8) \wedge (\neg x_5 \vee \neg x_8) \\ & \wedge (\neg x_3 \vee \neg x_6) \wedge (\neg x_3 \vee \neg x_9) \wedge (\neg x_6 \vee \neg x_9) \end{aligned}$$

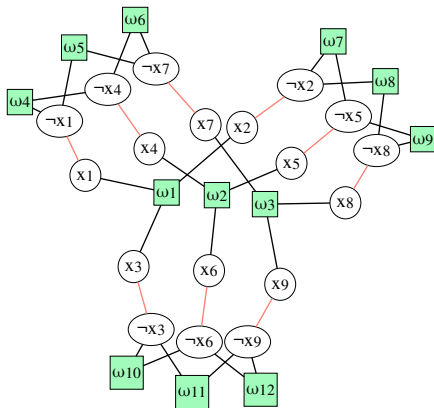
# Computing symmetries of a SAT problem

CNF formula

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6) \wedge (x_7 \vee x_8 \vee x_9) \\ & \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_1 \vee \neg x_7) \wedge (\neg x_4 \vee \neg x_7) \\ & \wedge (\neg x_2 \vee \neg x_5) \wedge (\neg x_2 \vee \neg x_8) \wedge (\neg x_5 \vee \neg x_8) \\ & \wedge (\neg x_3 \vee \neg x_6) \wedge (\neg x_3 \vee \neg x_9) \wedge (\neg x_6 \vee \neg x_9) \end{aligned}$$



colored graph



# Computing symmetries of a SAT problem

*CNF formula*

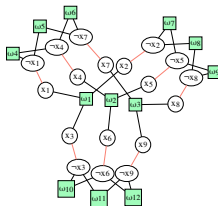


colored graph



graph automorphism

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6) \wedge (x_7 \vee x_8 \vee x_9) \\ & \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_1 \vee \neg x_7) \wedge (\neg x_4 \vee \neg x_7) \\ & \wedge (\neg x_2 \vee \neg x_5) \wedge (\neg x_2 \vee \neg x_8) \wedge (\neg x_5 \vee \neg x_8) \\ & \wedge (\neg x_3 \vee \neg x_6) \wedge (\neg x_3 \vee \neg x_9) \wedge (\neg x_6 \vee \neg x_9) \end{aligned}$$



(bliss, saucy, ...)

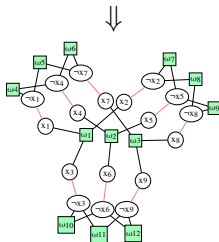
# Computing symmetries of a SAT problem

CNF formula

$$\begin{aligned} & (x_1 \vee x_2 \vee x_3) \wedge (x_4 \vee x_5 \vee x_6) \wedge (x_7 \vee x_8 \vee x_9) \\ & \wedge (\neg x_1 \vee \neg x_4) \wedge (\neg x_1 \vee \neg x_7) \wedge (\neg x_4 \vee \neg x_7) \\ & \wedge (\neg x_2 \vee \neg x_5) \wedge (\neg x_2 \vee \neg x_8) \wedge (\neg x_5 \vee \neg x_8) \\ & \wedge (\neg x_3 \vee \neg x_6) \wedge (\neg x_3 \vee \neg x_9) \wedge (\neg x_6 \vee \neg x_9) \end{aligned}$$



colored graph



⇓  
graph automorphism

⇓  
set of symmetries

⇓  
(bliss, saucy, ...)

⇓

$$\begin{aligned} g_1 &= (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9) \\ g_2 &= (x_4 \ x_7)(x_5 \ x_8)(x_6 \ x_9) \\ g_3 &= (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8) \\ g_4 &= (x_1 \ x_4)(x_2 \ x_5)(x_3 \ x_6) \end{aligned}$$

The set of symmetries of a formula is a group noted  $\langle G, \circ \rangle$

Exploitation of symmetries

Static symmetry breaking



# Orbit

Orbit of an assignment  $\alpha$  for a group  $G$ :

$$G.\alpha = \{g.\alpha \mid g \in G\}$$

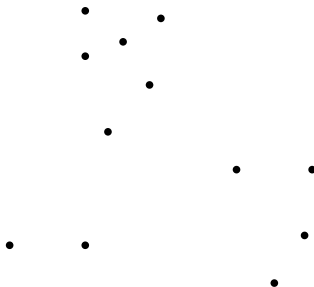
# Orbit

Orbit of an assignment  $\alpha$  for a group  $G$ :

$$G.\alpha = \{g.\alpha \mid g \in G\}$$

Example:

- full assignment



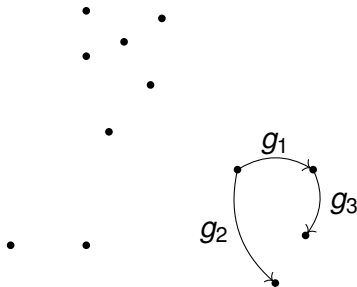
# Orbit

Orbit of an assignment  $\alpha$  for a group  $G$ :

$$G.\alpha = \{g.\alpha \mid g \in G\}$$

Example:

- full assignment



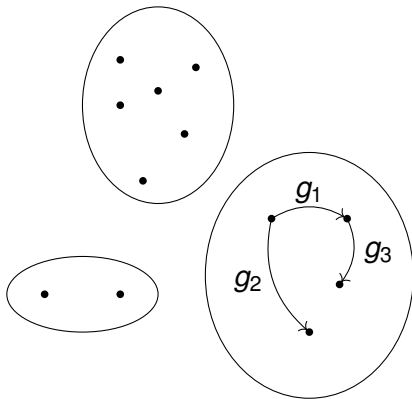
# Orbit

Orbit of an assignment  $\alpha$  for a group  $G$ :

$$G.\alpha = \{g.\alpha \mid g \in G\}$$

Example:

- full assignment
- orbit



Equivalence relation with respect to SAT:

- Either  $G.\alpha$  contains no solution
- Or all elements of  $G.\alpha$  are solutions

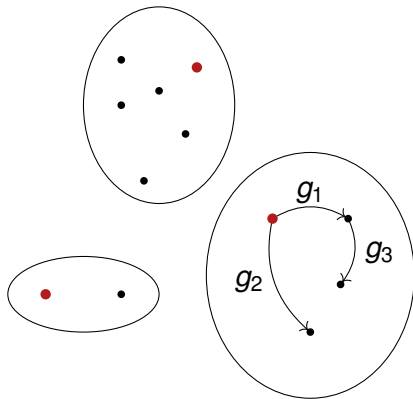
# Orbit

Orbit of an assignment  $\alpha$  for a group  $G$ :

$$G.\alpha = \{g.\alpha \mid g \in G\}$$

Example:

- full assignment
- orbit
- representative



Equivalence relation with respect to SAT:

- Either  $G.\alpha$  contains no solution
- Or all elements of  $G.\alpha$  are solutions

# Comparing assignments (1/2)

Define an ordering relation to compare assignments ( $\prec$ )

- Total ordering on variables
- Minimum value:  $F < T$  or  $T < F$

Allow only minimal value (lex-leader)

Forbid other assignments in each orbit

→ Add Symmetry breaking predicates (SBP)

## Comparing assignments (2/2)

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$	T	F	F	F	F	F	F	F

## Comparing assignments (2/2)

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$	T	F	F	F	F	F	F	F
$g.\alpha$	F	T	F	F	F	F	F	F



## Comparing assignments (2/2)

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$	T	F	F	F	F	F	F	F
$g.\alpha$	F	T	F	F	F	F	F	F

$$g.\alpha \prec \alpha$$

## Comparing assignments (2/2)

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

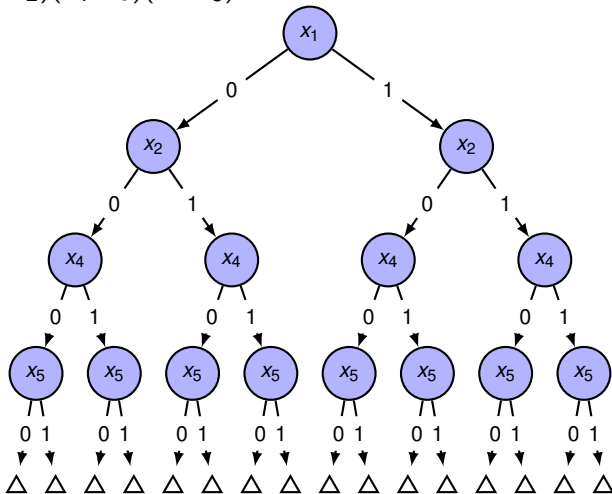
$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$	T	F	F	F	F	F	F	F
$g.\alpha$	F	T	F	F	F	F	F	F

$$g.\alpha \prec \alpha \Rightarrow \text{Generate SBP } \omega = \{\neg x_1, x_2\}$$

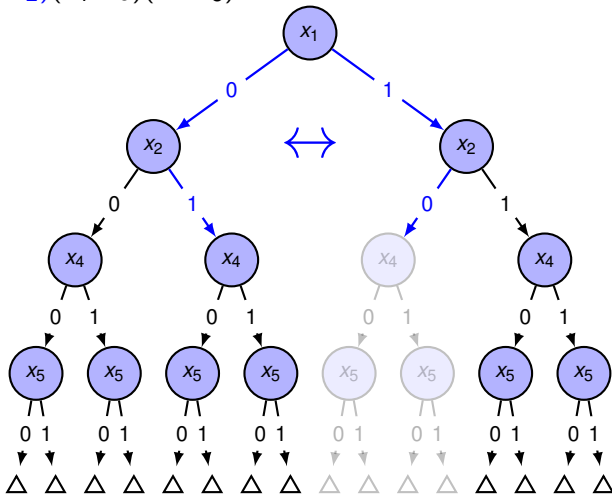
# Using symmetries to prune the search space

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

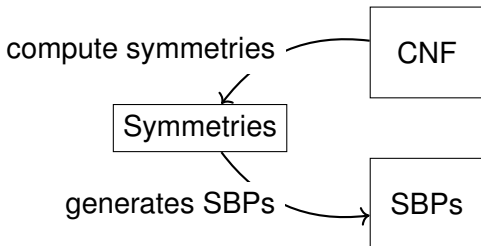


# Using symmetries to prune the search space

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$



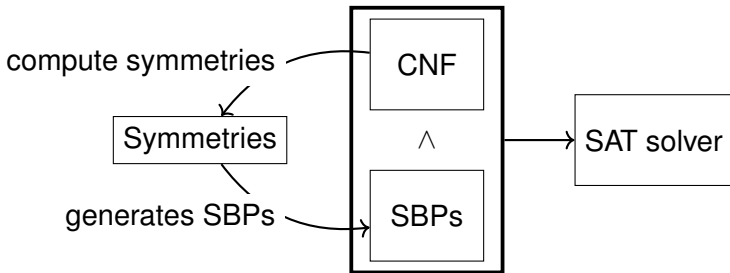
# Static symmetry breaking



State-of-the-art approaches:

- Shatter [ASM06]
- BreakID [DBBD16]

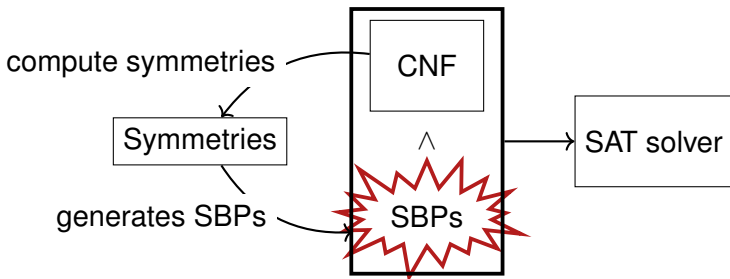
# Static symmetry breaking



State-of-the-art approaches:

- Shatter [ASM06]
- BreakID [DBBD16]

# Static symmetry breaking



State-of-the-art approaches:

- Shatter [ASM06]
- BreakID [DBBD16]

The solver can "explode" instead of being helped

- generate not needed clause
- flooding the solver

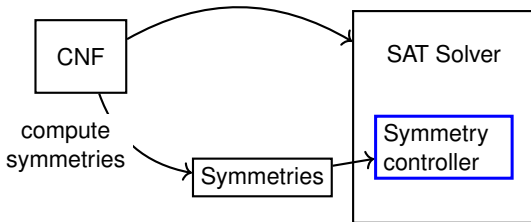
First contribution

CDCL[sym] Introducing Effective Symmetry  
Breaking in SAT Solving

TACAS'18 [MBCK18]



# General idea

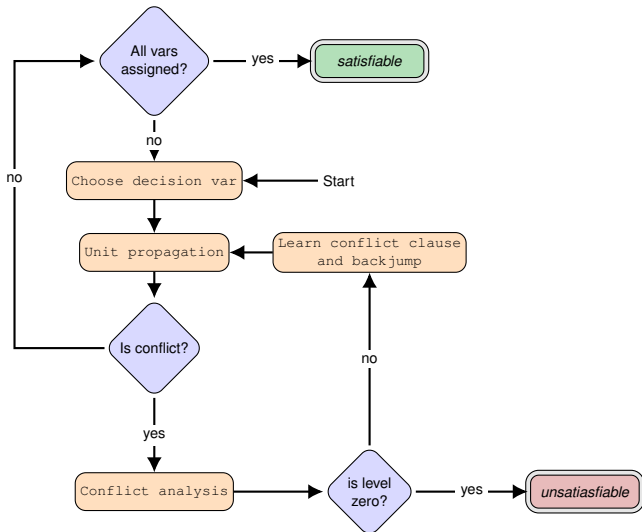


## Symmetry controller:

- generates SBP on the fly
- only when needed
- intrusive on solver

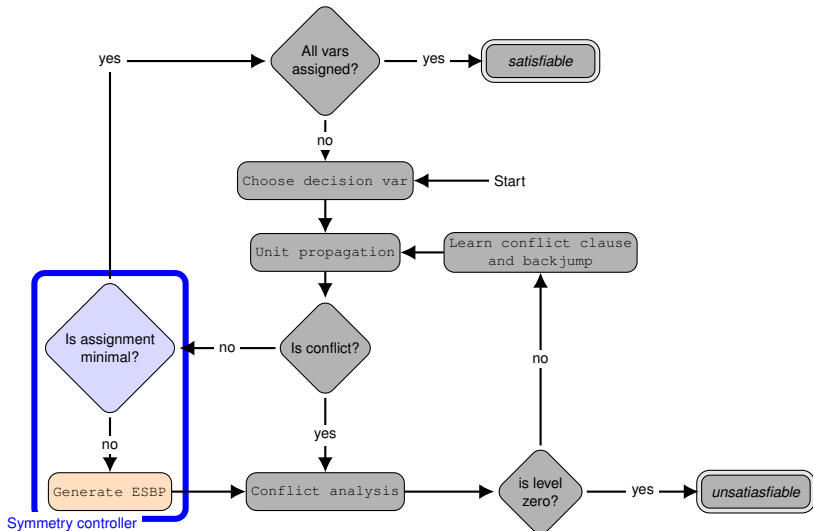
# CDCL[Sym]

Compute and inject SBP *opportunistically*, during the solving



# CDCL[Sym]

Compute and inject SBP *opportunistically*, during the solving



# Is assignment minimal? Symmetry status

- reducer:  $g.\alpha \prec \alpha$
- inactive:  $\alpha \prec g.\alpha$
- active: *not enough information*

# Is assignment minimal? Symmetry status

- reducer:  $g.\alpha \prec \alpha$
- inactive:  $\alpha \prec g.\alpha$
- active: *not enough information*

## Efficient implementation of symmetry status

**Keep track the smallest unassigned variable  $x$ :**

- 1  $\alpha(g.x) \leq \alpha(x)$ , then  $g$  is `reducer`  $\Rightarrow$  Effective SBP (ESBP)
- 2  $\alpha(x) \leq \alpha(g.x)$ , then  $g$  is `inactive`  $\Rightarrow g$  cannot reduce  $\alpha$
- 3  $\alpha(g.x)$  or  $\alpha(x)$  is unassigned then  $g$  is `active`

# Example

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$\downarrow$								
	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	
$\alpha$	U	U	U	U	U	U	U	U	
$g.\alpha$	U	U	U	U	U	U	U	U	

$$g.\alpha \quad \alpha$$

status of permutation  $g$ : active

# Example

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$\downarrow$								
		$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$		U	$\textcolor{red}{F}$	U	U	U	U	U	U
$g.\alpha$		$\textcolor{red}{F}$	U	U	U	U	U	U	U

$$g.\alpha \quad \alpha$$

status of permutation  $g$ : active

# Example

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$\downarrow$							
	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$	U	$\textcolor{red}{F}$	U	$\textcolor{red}{F}$	U	U	U	U
$g.\alpha$	$\textcolor{red}{F}$	U	U	U	$\textcolor{red}{F}$	U	U	U

$$g.\alpha \quad \alpha$$

status of permutation  $g$ : active



## Example

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$\downarrow$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$		$\textcolor{green}{T}$	$\textcolor{red}{F}$	U	$\textcolor{red}{F}$	U	U	U	U
$g.\alpha$		$\textcolor{red}{F}$	$\textcolor{green}{T}$	U	U	$\textcolor{red}{F}$	U	U	U

$$g.\alpha \prec \alpha$$

status of permutation g: reducer

# Example

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$\downarrow$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$
$\alpha$		$\textcolor{green}{T}$	$\textcolor{red}{F}$	U	$\textcolor{red}{F}$	U	U	U	U
$g.\alpha$		$\textcolor{red}{F}$	$\textcolor{green}{T}$	U	U	$\textcolor{red}{F}$	U	U	U

$$g.\alpha \prec \alpha$$

status of permutation g: reducer

Generate ESBP  $\omega = \{\neg x_1, x_2\}$

# CDCL[Sym] implementation

- C++ Implementation
  - Packaged as a library **cosy**<sup>1</sup> (Controller of Symmetry)
  - Lightweight
  - Fast update
  - Low memory consumption
  - Follows symmetry status
- 
- Works with any enumerative SAT solver
  - Can be integrated easily
- e.g. +3% LOC on MiniSAT<sup>2</sup>.

---

<sup>1</sup><https://github.com/lip6/cosy>

<sup>2</sup>90 lines out of 3090

# Experiments

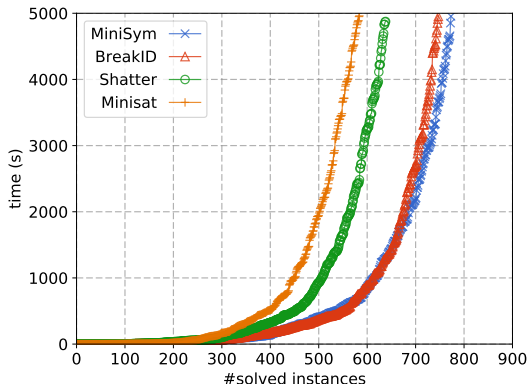
## Benchmark:

- from SAT contests 2012 – 2017
- filter: `bliss` finds symmetries in 1000 seconds
- 36 % of instances, 1 350/3 700

## Setup:

- four tools
  - MiniSat (no symmetry, baseline)
  - MiniSat + BreakID (SOTA SAT solver using symmetries)
  - MiniSat + Shatter (SOTA SAT solver using symmetries)
  - **MiniSym** = MiniSat + CDCL[Sym] (our approach)
- 5000 seconds timeout, 8GB memory
- includes time to compute symmetries (except for MiniSat)

# Experimental results



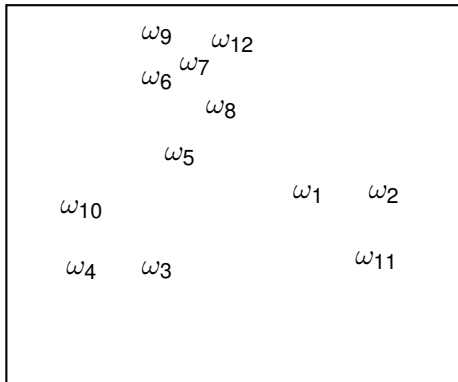
Solver	PAR-2	SAT	UNSAT
MiniSAT	2243h	325	261
Shatter	2088h	316	324
BreakID	1790h	334	415
MiniSym	1735h	336	439

Exploitation of symmetries

Dynamic symmetry breaking

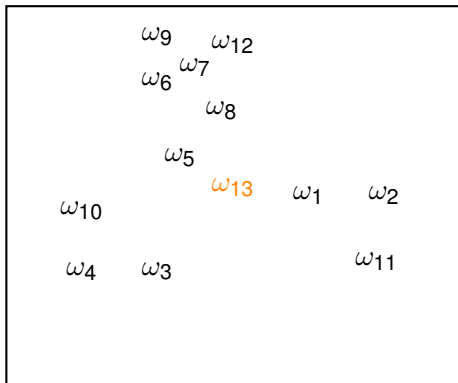
# Learn symmetrical clauses

- formula
- $\omega$  clause



# Learn symmetrical clauses

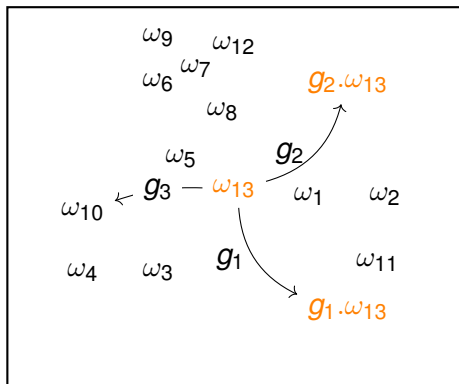
- formula
- $\omega$  clause
- $\omega$  learnt clause





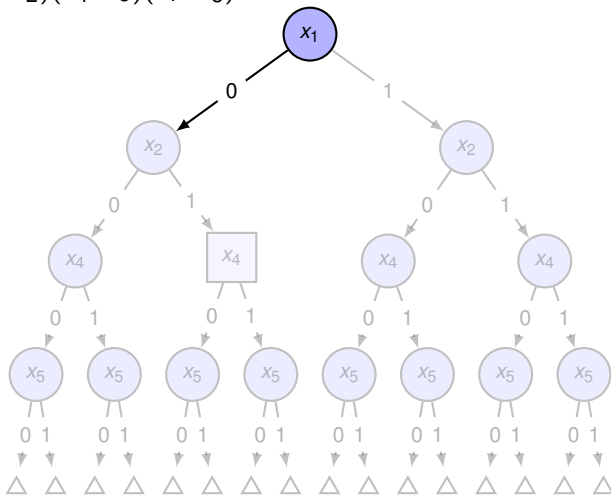
# Learn symmetrical clauses

- formula
- $\omega$  clause
- $\omega$  learnt clause



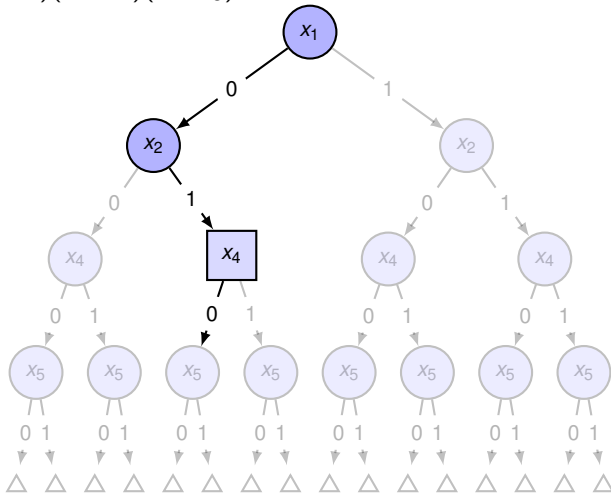
# Using symmetries to accelerate the tree traversal

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$



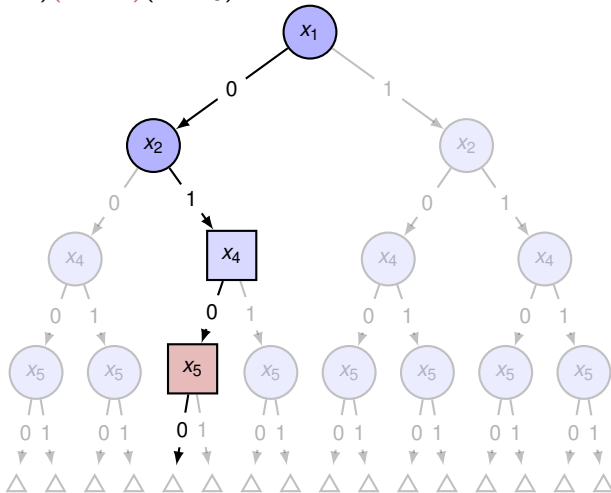
## Using symmetries to accelerate the tree traversal

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$



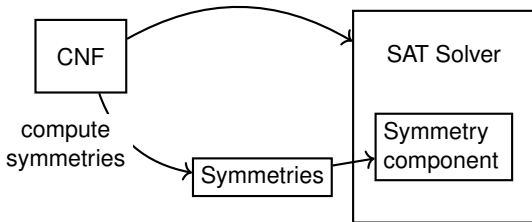
# Using symmetries to accelerate the tree traversal

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$



Use symmetries to deduce symmetrical facts.

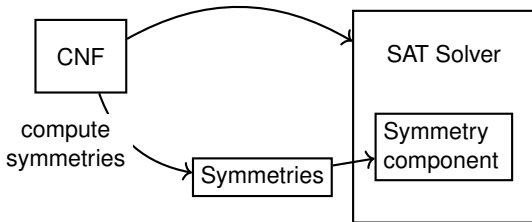
# Dynamic Symmetry Breaking



State-of-the-art:

- Symmchaff [Sab05]
- Symmetry Propagation (SP) [DBdC<sup>+</sup>12]
- Symmetry Learning Scheme (SLS) [BNOS10]
- Symmetry Explanation Learning (SEL) [DBB17]

# Dynamic Symmetry Breaking



State-of-the-art:

- Symmchaff [Sab05]
- Symmetry Propagation (SP) [DBdC<sup>+</sup>12]
- Symmetry Learning Scheme (SLS) [BNOS10]
- Symmetry Explanation Learning (SEL) [DBB17]

Observation:

Cannot handle some instances solved by static approach

Second contribution

Composing Symmetry Propagation and  
Effective Symmetry Breaking for SAT Solving

NFM'19 [MBK19]

# ESBP + SP

Compose the symmetry propagation and the ESBP

*prune the decision tree while accelerating its traversal*

Problems:

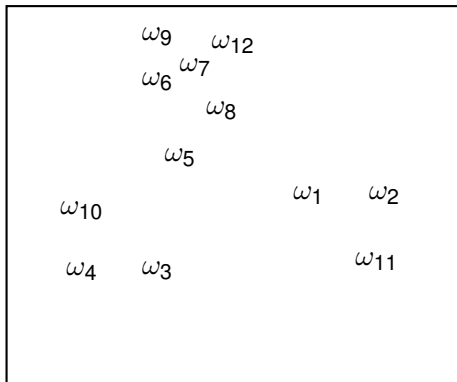
- ESBP breaks symmetries (incrementally)
- SP considers the manipulated symmetries valid all time

In a hybrid approach, SP must be able to identify  
**valid symmetries**



# Local symmetry

- formula
- $\omega$  clause
- $\omega$  learnt clause

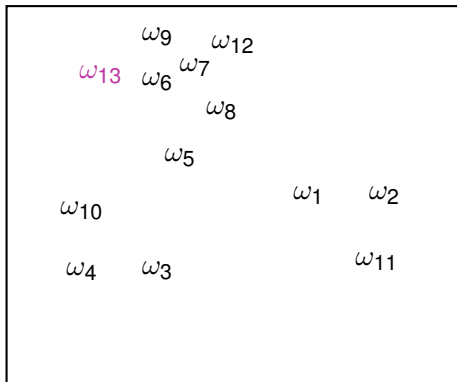


Local symmetries:

$$\omega \leftarrow \{g_1, g_2, g_3\}$$

# Local symmetry

- formula
- $\omega$  clause
- $\omega$  learnt clause
- $\omega$  esbp

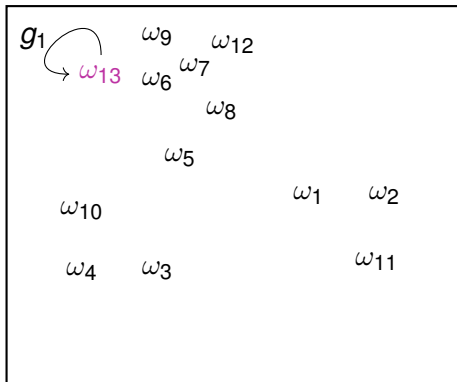


Local symmetries:

$$\omega \leftarrow \{g_1, g_2, g_3\}$$

# Local symmetry

- formula
- $\omega$  clause
- $\omega$  learnt clause
- $\omega$  esbp



Local symmetries:

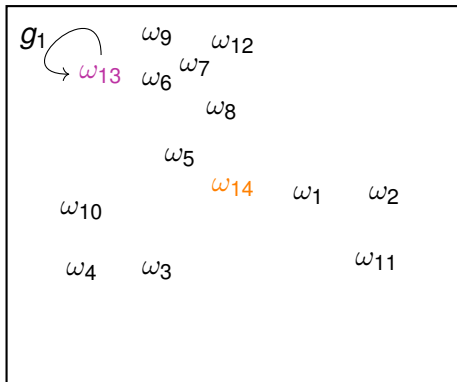
$$\omega \leftarrow \{g_1, g_2, g_3\}$$

$$\omega \leftarrow \{g_1\}$$

- Compute valid local symmetries
- On the fly
- At minimal cost

# Local symmetry

- formula
- $\omega$  clause
- $\omega$  learnt clause
- $\omega$  esbp



Local symmetries:

$$\omega \leftarrow \{g_1, g_2, g_3\}$$

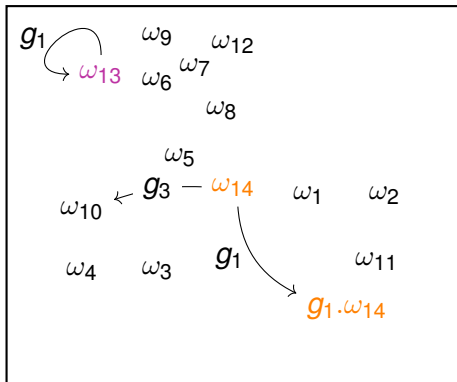
$$\omega \leftarrow \{g_1\}$$

- Compute valid local symmetries
- On the fly
- At minimal cost

**Inductive construction**

# Local symmetry

- formula
- $\omega$  clause
- $\omega$  learnt clause
- $\omega$  esbp



Local symmetries:

$$\omega \leftarrow \{g_1, g_2, g_3\}$$

$$\omega \leftarrow \{g_1\}$$

$$\omega \leftarrow \{g_1, g_3\}$$

- Compute valid local symmetries
- On the fly
- At minimal cost

**Inductive construction**

# Experimental results

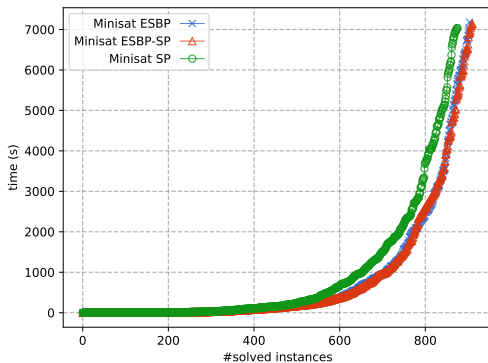
## Benchmark:

- from SAT contests 2012 – 2018
- filter: `bliss` finds symmetries in 1000 seconds
- 1400 symmetric instances (out of 4000)

## Setup:

- three tools
  - MiniSat SP (Minisat with Symmetry Propagation)
  - MiniSat ESBP (Minisat with CDCL[Sym])
  - **Minisat ESBP-SP** (our approach)
- 7200 seconds timeout

# Experimental results



Solver	PAR-2	SAT	UNSAT
SP	1674h00	406	470
ESBP	1578h30	416	488
<b>ESBP-SP</b>	<b>1570h15</b>	<b>420</b>	<b>491</b>

# Conclusion & Perspectives

## Conclusion

- A new dynamic symmetry breaking approach
  - Generation of SBP on the fly
  - Package as a library cosy usable with any CDCL solver
- A new hybrid approach (ESBP-SP)
  - Take advantage of static and dynamic approach



# Conclusion & Perspectives

## Conclusion

- A new dynamic symmetry breaking approach
  - Generation of SBP on the fly
  - Package as a library cosy usable with any CDCL solver
- A new hybrid approach (ESBP-SP)
  - Take advantage of static and dynamic approach
  - Combination of CDCL[Sym] with SEL

# Conclusion & Perspectives

## Conclusion

- A new dynamic symmetry breaking approach
  - Generation of SBP on the fly
  - Package as a library cosy usable with any CDCL solver
- A new hybrid approach (ESBP-SP)
  - Take advantage of static and dynamic approach
  - Combination of CDCL[Sym] with SEL

## Perspectives

- Exploitation of partial symmetries
- Symmetries and parallel SAT solver

# Conclusion & Perspectives

## Conclusion

- A new dynamic symmetry breaking approach
  - Generation of SBP on the fly
  - Package as a library cosy usable with any CDCL solver
- A new hybrid approach (ESBP-SP)
  - Take advantage of static and dynamic approach
  - **Combination of CDCL[Sym] with SEL**

## Perspectives

- Exploitation of partial symmetries
- Symmetries and parallel SAT solver

**Thanks !**



Fadi A. Aloul, Karem A. Sakallah, and Igor L. Markov.  
Efficient symmetry breaking for boolean satisfiability.  
*IEEE Trans. Computers*, 55(5):549–558, 2006.



Belaid Benhamou, Tarek Nabhani, Richard Ostrowski, and Mohamed Reda Saidi.

Enhancing clause learning by symmetry in sat solvers.

In *2010 22nd IEEE International Conference on Tools with Artificial Intelligence*, volume 1, pages 329–335. IEEE, 2010.



Stephen A Cook.

The complexity of theorem-proving procedures.

In *Proceedings of the third annual ACM symposium on Theory of computing*, pages 151–158. ACM, 1971.



Jo Devriendt, Bart Bogaerts, and Maurice Bruynooghe.

Symmetric explanation learning: Effective dynamic symmetry handling for sat.

In *International Conference on Theory and Applications of Satisfiability Testing*, pages 83–100. Springer, 2017.



Jo Devriendt, Bart Bogaerts, Maurice Bruynooghe, and Marc Denecker.

Improved static symmetry breaking for sat.

In *International Conference on Theory and Applications of Satisfiability Testing*, pages 104–122. Springer, 2016.



Jo Devriendt, Bart Bogaerts, Broes de Cat, Marc Denecker, and Christopher Mears.

Symmetry propagation: Improved dynamic symmetry breaking in SAT.

In *IEEE 24th International Conference on Tools with Artificial Intelligence, ICTAI 2012, Athens, Greece, November 7-9, 2012*, pages 49–56, 2012.



Martin Davis, George Logemann, and Donald Loveland.

A machine program for theorem-proving.

*Commun. ACM*, 5(7):394–397, July 1962.



Hakan Metin, Souheib Baarir, Maximilien Colange, and Fabrice Kordon.

Cdclsym: Introducing effective symmetry breaking in sat solving.

In *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, pages 99–114. Springer, 2018.



Hakan Metin, Souheib Baarir, and Fabrice Kordon.

Composing symmetry propagation and effective symmetry breaking for sat solving.

In *NASA Formal Methods Symposium*, pages 316–332. Springer, 2019.



Joao P Marques-Silva and Karem A Sakallah.

Grasp: A search algorithm for propositional satisfiability.

*IEEE Transactions on Computers*, 48(5):506–521, 1999.



Ashish Sabharwal.

Symchaff: A structure-aware satisfiability solver.

In *AAAI*, volume 5, pages 467–474, 2005.

# CDCL in action TODO



$$\omega_1 = \{x_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, x_6\}$$

$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg x_6\}$$

# CDCL in action TODO



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, x_6\}$$

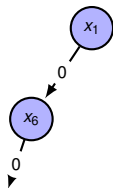
$$\omega_3 = \{\neg \mathbf{x}_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg x_6\}$$

# CDCL in action TODO



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, \mathbf{x}_6\}$$

$$\omega_3 = \{\neg \mathbf{x}_1, \neg x_5\}$$

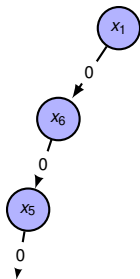
$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg \mathbf{x}_6\}$$



# CDCL in action TODO



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3\}$$

$$\omega_2 = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$$

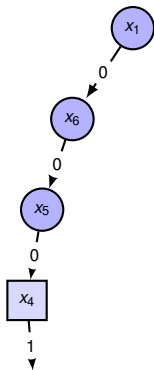
$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg \mathbf{x}_6\}$$

# CDCL in action TODO



$$\omega_1 = \{\mathbf{x}_1, x_2, x_3\}$$

$$\omega_2 = \{\mathbf{x}_4, \mathbf{x}_5, \mathbf{x}_6\}$$

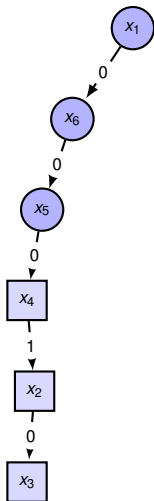
$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg \mathbf{x}_2, \neg \mathbf{x}_4\}$$

$$\omega_5 = \{\neg \mathbf{x}_3, \neg \mathbf{x}_4\}$$

$$\omega_6 = \{\neg x_3, \neg \mathbf{x}_6\}$$

# CDCL in action TODO



$$\omega_1 = \{x_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, x_6\}$$

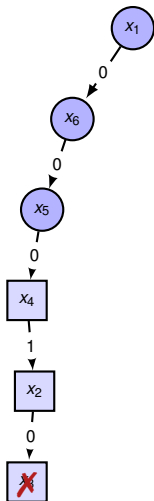
$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg x_6\}$$

# CDCL in action TODO



$$\omega_1 = \{x_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, x_6\}$$

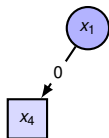
$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg x_6\}$$

# CDCL in action TODO



$$\omega_1 = \{x_1, x_2, x_3\}$$

$$\omega_2 = \{x_4, x_5, x_6\}$$

$$\omega_3 = \{\neg x_1, \neg x_5\}$$

$$\omega_4 = \{\neg x_2, \neg x_4\}$$

$$\omega_5 = \{\neg x_3, \neg x_4\}$$

$$\omega_6 = \{\neg x_3, \neg x_6\}$$

$$\omega_7 = \{x_1, \neg x_4\}$$

# Weakly active symmetries

## Logical consequence

When  $\omega$  is satisfied in all satisfying assignments of  $\varphi$ , we say that  $\omega$  is a logical consequence of  $\varphi$ , and we denote this by  $\varphi \vdash \omega$ .

# Weakly active symmetries

## Logical consequence

When  $\omega$  is satisfied in all satisfying assignments of  $\varphi$ , we say that  $\omega$  is a logical consequence of  $\varphi$ , and we denote this by  $\varphi \vdash \omega$ .

## Weakly active symmetries

Let a subset  $\delta \subseteq \alpha$ , a symmetry  $\sigma$  of  $\varphi$  such that  $\varphi \cup \delta \vdash \varphi \cup \alpha \wedge \sigma.\delta \subseteq \alpha$  then  $\sigma$  is weakly active symmetry.

# Weakly active symmetries

## Logical consequence

When  $\omega$  is satisfied in all satisfying assignments of  $\varphi$ , we say that  $\omega$  is a logical consequence of  $\varphi$ , and we denote this by  $\varphi \vdash \omega$ .

## Weakly active symmetries

Let a subset  $\delta \subseteq \alpha$ , a symmetry  $\sigma$  of  $\varphi$  such that  $\varphi \cup \delta \vdash \varphi \cup \alpha \wedge \sigma.\delta \subseteq \alpha$  then  $\sigma$  is weakly active symmetry.

## Symmetry propagation

Let  $\sigma$  a weakly active symmetry, then

$$\varphi \cup \alpha \vdash \{I\} \Leftrightarrow \varphi \cup \alpha \vdash \sigma.\{I\}$$



# Local symmetries

## Logical consequence

When  $\omega$  is satisfied in all satisfying assignments of  $\varphi$ , we say that  $\omega$  is a logical consequence of  $\varphi$ , and we denote this by  $\varphi \vdash \omega$ .

## Local Symmetries

Let  $\varphi$  be a formula. We define  $L_{\omega, \varphi}$ , the set of *local symmetries* for a clause  $\omega$ , and with respect to a formula  $\varphi$ , as follows:

$$L_{\omega, \varphi} = \{\sigma \in \mathfrak{S} \mid \varphi \vdash \sigma.\omega\}$$

# Local symmetries

## Logical consequence

When  $\omega$  is satisfied in all satisfying assignments of  $\varphi$ , we say that  $\omega$  is a logical consequence of  $\varphi$ , and we denote this by  $\varphi \vdash \omega$ .

## Local Symmetries

Let  $\varphi$  be a formula. We define  $L_{\omega, \varphi}$ , the set of *local symmetries* for a clause  $\omega$ , and with respect to a formula  $\varphi$ , as follows:

$$L_{\omega, \varphi} = \{\sigma \in \mathfrak{S} \mid \varphi \vdash \sigma.\omega\}$$

We can state that:

$$\bigcap_{\omega \in \varphi} L_{\omega, \varphi} \subseteq G.$$

# Computing local symmetries

Formula can be decomposed as :  $\varphi = \varphi_o \cup \varphi_e \cup \varphi_d$  where

- $\varphi_o$  is the set of the original clauses
- $\varphi_e$  is the set of ESBPs
- $\varphi_d$  is the set of deduced clauses.

## Local symmetries

- $\omega \in \varphi_o, L_{\omega, \varphi} \supseteq G$
- $\omega \in \varphi_e, L_{\omega, \varphi} \supseteq \text{Stab}(\omega) = \{\sigma \in G \mid \omega = \sigma.\omega\}$
- $\omega \in \varphi_d, L_{\omega, \varphi} \supseteq (\bigcap_{\omega' \in \varphi_1} L_{\omega', \varphi}) \cup \text{Stab}(\omega)$

where  $\varphi_1$  is the set of clauses that derives  $\omega$ .

# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\dots$	lex-leader	SBP

# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\dots$	lex-leader	SBP
$O_1$	$\textcolor{red}{F}$	$\textcolor{green}{T}$	-	-	-	$\dots$	✓	

# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\dots$	lex-leader	SBP
$O_1$	F	T	—	—	—	$\dots$	✓	$\rightarrow \neg x_1 \vee x_2$
	T	F	—	—	—	$\dots$	✗	

# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \textcolor{red}{F} < \textcolor{green}{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\dots$	lex-leader	SBP
$O_1$	$\textcolor{red}{F}$	$\textcolor{green}{T}$	—	—	—	$\dots$	✓	$\rightarrow \neg x_1 \vee x_2$
	$\textcolor{green}{T}$	$\textcolor{red}{F}$	—	—	—	$\dots$	✗	
$O_2$	$\textcolor{red}{F}$	$\textcolor{red}{F}$	—	$\textcolor{red}{F}$	$\textcolor{green}{T}$	$\dots$	✓	

# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\dots$	lex-leader	SBP
$O_1$	F	T	—	—	—	$\dots$	✓	$\rightarrow \neg x_1 \vee x_2$
	T	F	—	—	—	$\dots$	✗	
$O_2$	F	F	—	F	T	$\dots$	✓	$\rightarrow x_1 \vee x_2 \vee \neg x_4 \vee x_5$
	F	F	—	T	F	$\dots$	✗	



# Generates symmetry breaking predicates (SBP)

- Define lexicographic order
  - Define total order on variables
  - Define minimal value
- Forbid non minimal assignment for each orbit

Example:

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8; \text{F} < \text{T}$$

$$g = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

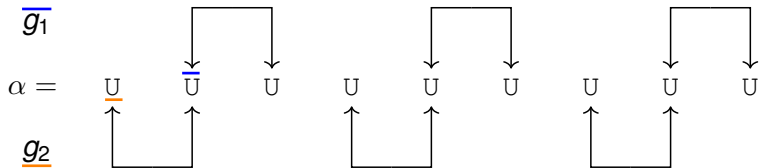
	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$\cdots$	lex-leader	SBP
$O_1$	F	T	—	—	—	$\cdots$	✓	$\rightarrow \neg x_1 \vee x_2$
	T	F	—	—	—	$\cdots$	✗	
$O_2$	F	F	—	F	T	$\cdots$	✓	$\rightarrow x_1 \vee x_2 \vee \neg x_4 \vee x_5$
	F	F	—	T	F	$\cdots$	✗	
$\dots$								

# Example

$$g_1 = (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9)$$

$$g_2 = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$$\textcolor{red}{F} < \textcolor{green}{T} \quad x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 \leq x_9$$

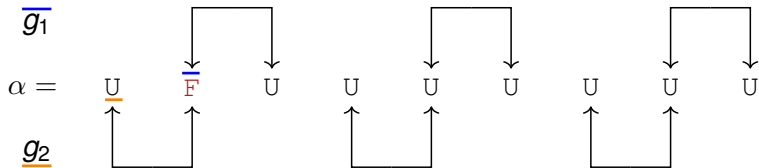


# Example

$$g_1 = (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9)$$

$$g_2 = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$$\textcolor{red}{F} < \textcolor{green}{T} \quad x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 \leq x_9$$

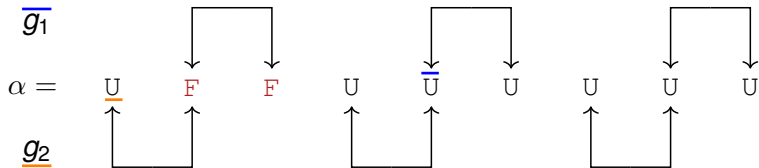


# Example

$$g_1 = (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9)$$

$$g_2 = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$$\textcolor{red}{F} < \textcolor{green}{T} \quad x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 \leq x_9$$

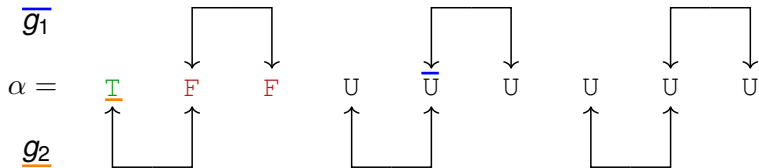


# Example

$$g_1 = (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9)$$

$$g_2 = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$$\textcolor{red}{F} < \textcolor{green}{T} \quad x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 \leq x_9$$

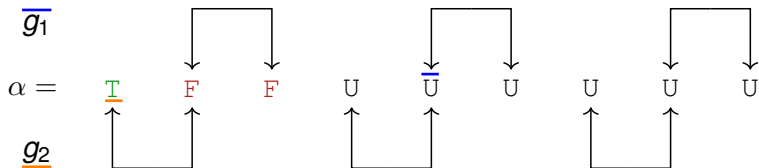


# Example

$$g_1 = (x_2 \ x_3)(x_5 \ x_6)(x_8 \ x_9)$$

$$g_2 = (x_1 \ x_2)(x_4 \ x_5)(x_7 \ x_8)$$

$$\textcolor{red}{F} < \textcolor{green}{T} \quad x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 \leq x_9$$



$$g_2 \text{ generates ESBP } \omega = \{\neg x_1, x_2\}$$

# Example

- 1 reducer:  $\alpha(g.x) \leq \alpha(x)$
- 2 inactive:  $\alpha(x) \leq \alpha(g.x)$
- 3 active:  $\alpha(g.x)$  or  $\alpha(x)$  is unassigned

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 ; \text{ F } < \text{ T }$$

$$g_1 = \begin{array}{ccc} (x_2 & x_3) & (x_5 & x_6) & (x_8 & x_9) \end{array} \left| \begin{array}{l} x = x_2 \\ g.x = x_3 \\ \text{active} \end{array} \right.$$

↑

$$g_2 = \begin{array}{ccc} (x_1 & x_2) & (x_4 & x_5) & (x_7 & x_8) \end{array} \left| \begin{array}{l} x = x_1 \\ g.x = x_2 \\ \text{active} \end{array} \right.$$

↑

...

$$\alpha = \{ \quad \quad \quad \}$$

# Example

- 1 reducer:  $\alpha(g.x) \leq \alpha(x)$
- 2 inactive:  $\alpha(x) \leq \alpha(g.x)$
- 3 active:  $\alpha(g.x)$  or  $\alpha(x)$  is unassigned

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 ; \text{ F } < \text{ T }$$

$$g_1 = \begin{array}{ccc} (\textcolor{red}{x}_2 & x_3) & (x_5 & x_6) & (x_8 & x_9) \end{array} \left| \begin{array}{l} x = \textcolor{red}{x}_2 \\ g.x = x_3 \\ \text{active} \end{array} \right.$$

$$g_2 = \begin{array}{ccc} (x_1 & \textcolor{red}{x}_2) & (x_4 & x_5) & (x_7 & x_8) \end{array} \left| \begin{array}{l} x = x_1 \\ g.x = \textcolor{red}{x}_2 \\ \text{active} \end{array} \right.$$

...

$$\alpha = \{ \neg x_2 \quad \}$$



# Example

- 1 reducer:  $\alpha(g.x) \leq \alpha(x)$
- 2 inactive:  $\alpha(x) \leq \alpha(g.x)$
- 3 active:  $\alpha(g.x)$  or  $\alpha(x)$  is unassigned

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 ; \text{ F } < \text{ T }$$

$$g_1 = \begin{array}{cc|cc} (x_2 & x_3) & (x_5 & x_6) & (x_8 & x_9) \\ \uparrow & & & & & \end{array} \left| \begin{array}{l} x = x_5 \\ g.x = x_6 \\ \text{active} \end{array} \right.$$

$$g_2 = \begin{array}{cc|cc} (x_1 & x_2) & (x_4 & x_5) & (x_7 & x_8) \\ \uparrow & & & & & \end{array} \left| \begin{array}{l} x = x_1 \\ g.x = x_2 \\ \text{reducer} \end{array} \right.$$

...

$$\alpha = \{\neg x_2, \neg x_3, x_1\}$$

# Example

- 1 reducer:  $\alpha(g.x) \leq \alpha(x)$
- 2 inactive:  $\alpha(x) \leq \alpha(g.x)$
- 3 active:  $\alpha(g.x)$  or  $\alpha(x)$  is unassigned

$$x_1 \leq x_2 \leq x_3 \leq x_4 \leq x_5 \leq x_6 \leq x_7 \leq x_8 ; \text{ F } < \text{ T }$$

$$g_1 = \begin{array}{cc|cc} (x_2 & x_3) & (x_5 & x_6) & (x_8 & x_9) \\ \uparrow & & & & & \end{array} \left| \begin{array}{l} x = x_5 \\ g.x = x_6 \\ \text{active} \end{array} \right.$$

$$g_2 = \begin{array}{cc|cc} (x_1 & x_2) & (x_4 & x_5) & (x_7 & x_8) \\ \uparrow & & & & & \end{array} \left| \begin{array}{l} x = x_1 \\ g.x = x_2 \\ \text{reducer} \end{array} \right.$$

...

$$\alpha = \{\neg x_2, \neg x_3, x_1\}$$

$$g_2 \text{ generates } \omega = \{\neg x_1, x_2\}$$

# Encoding the problem

$(A, 1)(A, 2)(A, 3)$   
 $(B, 1)(B, 2)(B, 3)$   
 $(C, 1)(C, 2)(C, 3)$

$x_1 \vee x_2 \vee x_3$

$x_4 \vee x_5 \vee x_6$

$x_7 \vee x_8 \vee x_9$

$\neg(A, 1)\neg(B, 1)$

$\neg x_1 \vee \neg x_4$

$\neg(A, 1)\neg(C, 1)$

$\neg x_1 \vee \neg x_7$

$\neg(B, 1)\neg(C, 1)$

$\neg x_4 \vee \neg x_7$

$\neg(A, 2)\neg(B, 2)$

$\neg x_2 \vee \neg x_5$

$\neg(A, 2)\neg(C, 2)$

$\neg x_2 \vee \neg x_8$

$\neg(B, 2)\neg(C, 2)$

$\neg x_5 \vee \neg x_8$

$\neg(A, 3)\neg(B, 3)$

$\neg x_3 \vee \neg x_6$

$\neg(A, 3)\neg(C, 3)$

$\neg x_3 \vee \neg x_9$

$\neg(B, 3)\neg(C, 3)$

$\neg x_6 \vee \neg x_9$