# Symmetry Propagation

## Improved Dynamic Symmetry Breaking in SAT

Jo Devriendt, Bart Bogaerts, Christopher Mears, Broes De Cat, Marc Denecker

KU Leuven and Monash University

ICTAI'12

# Outline

# SAT problem



- SAT **theory**: conjunction of clauses
- **clause**: disjunction of literals
- **literal**: variable or its negation

$\Rightarrow$ SAT problem consists of deciding whether a model exists for a theory.

# SAT problem: Pigeonhole problem

Variables for the pigeonhole problem (4 pigeons, 3 holes):

$_i in_j$ means "pigeon $i$ is in hole $j$".

Theory:

Each pigeon in a hole:

$(_1in_1 \vee {}_1in_2 \vee {}_1in_3)\wedge$
$(_2in_1 \vee {}_2in_2 \vee {}_2in_3)\wedge$
$(_3in_1 \vee {}_3in_2 \vee {}_3in_3)\wedge$
$(_4in_1 \vee {}_4in_2 \vee {}_4in_3)$

No two pigeons in one hole:

$(\neg_1in_1 \vee \neg_2in_1)\wedge$
$(\neg_1in_1 \vee \neg_3in_1)\wedge$
$(\neg_1in_1 \vee \neg_4in_1)\wedge$
$(\neg_2in_1 \vee \neg_3in_1)\wedge$
$(\neg_2in_1 \vee \neg_4in_1)\wedge$
$(\neg_3in_1 \vee \neg_4in_1)\wedge$
$\ldots$

# SAT solving



- **Assignment** $\alpha$: set of literals currently true
- **Decision literals** $\delta \subseteq \alpha$: choices made during search
- Literals in $\alpha \setminus \delta$: **Propagated literals**
  - are **logical consequences** of $T + \delta$
    - often, but not always, derived via **unit propagation**
  - have **explanation clauses**
    - explanation clauses are **unit** given $\alpha$.
- Upon conflict: conflict resolution on explanation clauses
  - produces a **learned clause**
    - is a **logical consequence** of $T$

Notation: $T \models x$ means $x$ is logical consequence of $T$.

# SAT solving: Pigeonhole problem

Choose ${}_1in_1$:

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \{{}_1in_1\}$
$\alpha = \{{}_1in_1, \neg{}_2in_1, \neg{}_3in_1, \neg{}_4in_1\}$

# SAT solving: Pigeonhole problem



After further search, conflict occurs.
Learned clause: $(\neg {}_1in_1)$

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \emptyset$
$\alpha = \{\neg {}_1in_1\}$

# SAT solving: Pigeonhole problem



Continue search after choosing new decision literal.

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

A symmetrical part of the search space is explored . . .

# Symmetry in SAT



Jo Devriendt, Bart Bogaerts, Christopher Mears, Broes De Cat, Marc Denecker



How to avoid visiting symmetrical branches of search tree?

# Symmetries



- A **symmetry** of $T$ is a permutation on the literals of $T$ such that
  - $\sigma(\neg\ell) = \neg\sigma(\ell)$
  - $\alpha$ is a model of $T \Leftrightarrow \sigma(\alpha)$ is a model of $T$
- Symmetry definition lifts naturally to clauses and assignments.

# Symmetries: Pigeonhole problem



E.g. The symmetry $\sigma$ swapping pigeon 1 and 2:

$$
\begin{aligned}
{}_1in_1 &\leftrightarrow {}_2in_1 \\
{}_1in_2 &\leftrightarrow {}_2in_2 \\
{}_1in_3 &\leftrightarrow {}_2in_3 \\
\neg{}_1in_1 &\leftrightarrow \neg{}_2in_1 \\
\neg{}_1in_2 &\leftrightarrow \neg{}_2in_2 \\
\neg{}_1in_3 &\leftrightarrow \neg{}_2in_3
\end{aligned}
$$

# Symmetries: basic properties



- A **symmetry** of $T$ is a permutation on the literals of $T$ such that
  - $\sigma(\neg\ell) = \neg\sigma(\ell)$
  - $\alpha$ is a model of $T \Leftrightarrow \sigma(\alpha)$ is a model of $T$
- Symmetries form **symmetry groups** under composition
  - The number of symmetries in a symmetry group can be exponential (in the domain of the problem).

For all symmetries $\sigma$ of $T$ and for all possible clauses $c$:
$T \models c$ implies $T \models \sigma(c)$.

# Previous work



- For every learned clause $c$: $T \models c$.
  - For every learned clause $c$, add $\sigma(c)$ to $T$, for every symmetry $\sigma$ of $T$.
- Does not work.
  - Generates too many clauses.
- Symmetrical Learning Scheme: (Benhamou et al. 2010)
  - use only a small generator set $G$ of the symmetry group.
  - for every learned clause $c$, add $\sigma(c)$ for $\sigma \in G$
- Observations:
  - Many generated clauses are never used for propagation
  - Possible propagating clauses are never generated

# Idea 1

- Also use generator set $G$.
- Heuristic idea: use symmetrical of explanation clauses. $\sigma(c)$ is more likely to be unit given $\alpha$ if $c$ is an explanation clause given $\alpha$.
- implementation:
  - after unit propagation, detect a $\sigma(c)$ which is **unit and not satisfied**.
    - $c$ is an explanation clause given $\alpha$
    - $\sigma \in G$
  - add $\sigma(c)$ to $T$ and propagate
  - go back to unit propagation, repeat.
- All added $\sigma(c)$ will propagate at least once.
- $\sigma(c)$ is an explanation clause, so $\sigma(\sigma(c))$ will be checked

# The pigeonhole problem continued

Recall this situation - learned clause was $(\neg {}_1in_1)$

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \emptyset$
$\alpha = \{\neg {}_1in_1\}$

## The pigeonhole problem continued

- $G = \{\sigma_{1\leftrightarrow 2}, \sigma_{2\leftrightarrow 3}, \sigma^{1\leftrightarrow 2}, \sigma^{2\leftrightarrow 3}, \sigma^{3\leftrightarrow 4}\}$
  - $\sigma^{i\leftrightarrow j}$: swap pigeon $i$ with $j$
  - $\sigma_{i\leftrightarrow j}$: swap hole $i$ with $j$
- Explanation clause is $(\neg_1 in_1)$, so symmetrical clauses are $(\neg_2 in_1)$ and $(\neg_1 in_2)$.
- Both are unit and unsatisfied $\Rightarrow$ learn them.
- Propagate $\neg_2 in_1$, $\neg_1 in_2$.

| | | | |
|---|---|---|---|
| $_1 in_1$ | $_2 in_1$ | $_3 in_1$ | $_4 in_1$ |
| $_1 in_2$ | $_2 in_2$ | $_3 in_2$ | $_4 in_2$ |
| $_1 in_3$ | $_2 in_3$ | $_3 in_3$ | $_4 in_3$ |

$\delta = \emptyset$

$\alpha = \{\neg_1 in_1, \neg_1 in_2, \neg_2 in_1\}$

# The pigeonhole problem continued



- Unit propagation derives ${}_1in_3$.
- $(\neg{}_1in_2)$, $(\neg{}_2in_1)$ are now explanation clauses.
- Further addition of symmetrical clauses leads to conflict.

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| $\underline{{}_1in_3}$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \emptyset$

$\alpha = \{\neg{}_1in_1, \neg{}_1in_2, \neg{}_2in_1, {}_1in_3, \neg{}_2in_2, \neg{}_1in_3, \neg{}_3in_1\}$

# Observation



- Observation: expensive check to discover whether $\sigma(c)$ is unit for arbitrary explanation clauses $c$ given $\alpha$, for all $\sigma \in G$.
- Improvement is possible.
  - Mears et al. offers a partial solution.

# Active symmetries



Mears et al. 2008:

- Symmetry $\sigma$ is **active** under $\alpha$ if:
  - $\sigma(\alpha) = \alpha$

For all active symmetries $\sigma$:
$T + \alpha \models \ell$ implies $T + \alpha \models \sigma(\ell)$.

- The explanation clause for $\sigma(\ell)$ is $\sigma(c)$, with $c$ the explanation of $\ell$.
- Given $\alpha$, for active symmetries $\sigma \in G$, for all new explanation clauses $c$: $\sigma(c)$ will be unit.

# Active symmetries



We can do better . . .

# Idea 2

- Symmetry $\sigma$ is **weakly active** given $\delta$ and $\alpha$ if:
  - $\sigma(\delta) \subseteq \alpha$

For all weakly active symmetries $\sigma$:
$T + \delta \models \ell$ implies $T + \delta \models \sigma(\ell)$.

- Generalization of active symmetries:
  all active symmetries are weakly active, but not vice versa.
  $\Rightarrow$ there are more weakly active symmetries

# The pigeonhole problem continued



Recall propagations by Idea 1:

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \emptyset$
$\alpha = \{\neg {}_1in_1, \neg {}_1in_2, \neg {}_2in_1, \neg {}_2in_2, \neg {}_1in_3, \neg {}_3in_1\}$

# The pigeonhole problem continued



Literals that would be propagated by weakly active symmetries (Idea 2):

| | | | |
|---|---|---|---|
| ${}_1in_1$ | ${}_2in_1$ | ${}_3in_1$ | ${}_4in_1$ |
| ${}_1in_2$ | ${}_2in_2$ | ${}_3in_2$ | ${}_4in_2$ |
| ${}_1in_3$ | ${}_2in_3$ | ${}_3in_3$ | ${}_4in_3$ |

$\delta = \emptyset$

$\alpha = \{\neg{}_1in_1, \neg{}_1in_2, \neg{}_2in_1, \neg{}_2in_2, \neg{}_1in_3, \neg{}_3in_1\}$

# Symmetry Propagation



- Both ideas can be combined
  - For weakly active symmetries: Idea 2
  - For weakly inactive symmetries: Idea 1
- We call this **"Symmetry Propagation"** (SP).
- SP **only infers logical consequences**.
  - No actual symmetry **"breaking"** occurs!
  - No solutions are pruned.

# Experimental Results

Experimental results showed Symmetry Propagation is **competitive to state-of-the-art** symmetry breaking in SAT.

# Summary and Future Directions



- Symmetry Propagation is new approach to dynamic symmetry breaking.
  - Efficently detecting logical consequences $\sigma(c)$ or $\sigma(\ell)$, of a theory during search.
- Future work:
  - Application to general constraint programming.
  - Search heuristics to maximize weakly active symmetries.

# Thank you!



Questions?