Search Strategy for Constraint Programming

Constraint programming

\[ \text{Constraint model} + \text{Search procedure} \]

Search procedure is crucial to solve hard combinatorial (NP-complete) problems.

The Clp2Zinc Theorem

Reified constraints

\[ \text{Constraint model } M \quad \text{Constraint model } M \cup M', \]

Tree search procedure \( t \) \(~\sim\) Basic labeling \( t' \)

ClpZinc

A Modeling Language for Constraints and Search.

- Modeling search independently from the underlying constraint solver through tree search procedures with state variables.
- Extending MiniZinc with **Horn clauses with constraints** (Prolog-like search description language).

Available compiler targeting most common solvers:

http://lifeware.inria.fr/~tmartine/clp2zinc

Dichotomic Search: The Code

\[
dichotomy(X, Min, Max) ::= \\
\{ \\
\quad \text{dichotomy}(X, \text{ceil}(\log(2, Max - Min + 1))). \\
\quad \text{dichotomy}(X, \text{Depth}) ::= \\
\quad \text{Depth} > 0, \\
\quad \text{Middle} = (\text{min}(X) + \text{max}(X)) \text{ div } 2, \\
\quad (X \ll \text{Middle}; X > \text{Middle}), \\
\quad \text{dichotomy}(X, \text{Depth} - 1). \\
\quad \text{dichotomy}(X, 0). \\
\}
\]

\[
\text{var } 0..5: x; \\
\text{dichotomy}(x, 0, 5).
\]

Interval Splitting: The Code

\[
\text{interval_splitting}(X, \text{Step}, \text{Min}, \text{Max}) ::= \\
\{ \\
\quad \text{Min} + \text{Step} \leq \text{Max}, \text{NextX} = \text{min}(X) + \text{Step}, \\
\quad \{ \\
\quad \quad X < \text{NextX}, \\
\quad \quad \text{interval_splitting}(X, \text{Step}, \text{Min} + \text{Step}, \text{Max}) \\
\quad \}; \\
\quad \text{NextX} = \text{NextX}, \\
\quad \text{interval_splitting}(X, \text{Step}, \text{Min} + \text{Step}, \text{Max}) \\
\}; \\
\}
\]

\[
\text{var } 0..5: x; \\
\text{interval_splitting}(x, 2, 0, 5).
\]

Korf’s packing problem

Given an integer \( n \geq 1 \), find an enclosing rectangle of smallest area containing \( n \) squares from sizes \( 1 \times 1, 2 \times 2, \) up to \( n \times n \), without overlap.


One **provably optimal placement** for \( n = 26 \):

![Korf's packing solution](image)

In practice, ClpZinc programs is 2–3 \( \times \) slower than native programs: the constant is small w.r.t. to the combinatorial complexity.

Meta-interpretation, beyond tree search

Meta-interpretation:

- Limited discrepancy search (LDS)
- Symmetry breaking during search (SBDS)

State variables, persistent through backtracking:

Optimization procedure, branch-and-bound.

Dichotomic Search: The Search Tree

For \( x \in [0, 2^d] \). Obtained by **domain filtering** and **constraint propagation** of the equality \( x = \sum_{0 \leq k < d} x_k 2^k \) with \( x_k \in \{0,1\} \).

![Dichotomic search tree](image)

Interval Splitting: The Search Tree

For a fixed step \( s \geq 1 \) and for \( x \in [0, n] \). Obtained by **domain filtering** and **constraint propagation** of the equality \( x = s \cdot x + q \cdot r \) where \( r \in [0, s] \).

![Interval splitting search tree](image)