ID2204: Constraint Programming

Introduction & Overview

Lecture 01, 2018-03-19

Christian Schulte
cschulte@kth.se

Software and Computer Systems
School of Electrical Engineering and Computer Science
KTH Royal Institute of Technology
Sweden
Sudoku

Assign blank fields digits such that:
digits distinct per rows, columns, blocks
Prune digits from fields such that:
digits distinct per rows, columns, blocks
Propagation

- Prune digits from fields such that: digits distinct per rows, columns, blocks
Propagation

- Prune digits from fields such that:
  - digits distinct per rows, **columns**, blocks

![Number puzzle image](image-url)
Propagation

<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>2</td>
<td></td>
<td></td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>9</td>
<td></td>
<td></td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>7</td>
<td></td>
<td>4</td>
<td>9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>9</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>4</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td></td>
<td></td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Prune digits from fields such that: digits distinct per rows, columns, **blocks**
Iterated Propagation

- Iterate propagation for rows, columns, blocks
- What if no assignment: search... later
Running Example: SMM

- Find distinct digits for letters, such that

\[
\begin{align*}
SEND &+ MORE \\
\underline{} &  \\
= & MONEY
\end{align*}
\]
Constraint Model for SMM

- Variables:
  \[ S, E, N, D, M, O, R, Y \in \{0, \ldots, 9\} \]

- Constraints:
  \[
  \text{distinct}(S, E, N, D, M, O, R, Y)
  \]
  \[
  10000 \times S + 100 \times E + 10 \times N + D
  \]
  \[
  + \quad 10000 \times M + 100 \times O + 10 \times R + E
  \]
  \[
  = 10000 \times M + 100 \times O + 100 \times N + 10 \times E + Y
  \]
  \[
  S \neq 0 \quad \quad M \neq 0
  \]
Finding a Solution

- Compute with possible values
  - rather than enumerating assignments

- Prune inconsistent values
  - constraint propagation

- Search
  - branch: define search tree
  - explore: explore search tree for solution
Constraint Propagation
Constraint Store

- Maps variables to possible values
- Others: finite sets, intervals, trees, ...

\[ x \in \{3, 4, 5\} \quad y \in \{3, 4, 5\} \]

finite domain constraints
Propagators

- Implement (non-basic) constraints

\[
distinct(x_1, \ldots, x_n)
\]

\[
x + 2*y = z
\]
Propagators

- Amplify store by constraint propagation
Propagators

- Amplify store by constraint propagation

\[ x \geq y \quad y > 3 \]
\[ x \in \{3, 4, 5\} \quad y \in \{4, 5\} \]
Propagators

- Amplify store by constraint propagation

\begin{align*}
x \geq y & \quad y > 3 \\
x \in \{3, 4, 5\} & \quad y \in \{4, 5\}
\end{align*}
Propagators

- Amplify store by constraint propagation

\[ x \geq y \quad y > 3 \]

\[ x \in \{4, 5\} \quad y \in \{4, 5\} \]
Propagators

- Amplify store by constraint propagation
- Disappear when done (subsumed, entailed)
  - no more propagation possible

\[ x \geq y, \quad y > 3 \]

\[ x \in \{4,5\}, \quad y \in \{4,5\} \]
Propagators

- Amplify store by constraint propagation
- Disappear when done (subsumed, entailed)
  - no more propagation possible

\[
x \geq y
\]

\[
x \in \{4, 5\}, \quad y \in \{4, 5\}
\]
Propagation for SMM

- Results in store
  \[ S \in \{9\} \quad E \in \{4, \ldots, 7\} \quad N \in \{5, \ldots, 8\} \quad D \in \{2, \ldots, 8\} \]
  \[ M \in \{1\} \quad O \in \{0\} \quad R \in \{2, \ldots, 8\} \quad Y \in \{2, \ldots, 8\} \]

- Propagation \textbf{alone} not sufficient!
  - create simpler sub-problems
  - branching
Constraints and Propagators

- Constraints state relations among variables
  - which value combinations satisfy constraint

- Propagators implement constraints
  - prune values in conflict with constraint

- Constraint propagation drives propagators for several constraints
Search
Search: Branching

- Create subproblems with additional information
  - enable further constraint propagation
Example Branching Strategy

- Pick variable $x$ with at least two values
- Pick value $n$ from domain of $x$
- Branch with $x=n$ and $x \neq n$

- Part of model
Search: Exploration

- Iterate propagation and branching
- Orthogonal: branching $\Leftrightarrow$ exploration
- Nodes:
  - Unsolved
  - Failed
  - Succeeded
SMM: Unique Solution

SEND
+ MORE
= MONEY

9567
+ 1085
= 10652
Heuristics for Branching

- **Which variable**
  - least possible values (first-fail)
  - application dependent heuristic

- **Which value**
  - minimum, median, maximum
    - $x=m$ or $x \neq m$
  - split with median $m$
    - $x < m$ or $x \geq m$

- **Problem specific**
SMM: Solution With First-fail

SEND

+ MORE

= MONEY

9567

+ 1085

= 10652
Send Most Money (SMM++)

- Find distinct digits for letters, such that

\[
\begin{align*}
\text{SEND} & \quad + \quad \text{MOST} \\
\hline
\text{MONEY} & \quad = \quad \text{MONEY}
\end{align*}
\]

and \text{MONEY} maximal
Best Solution Search

- Naïve approach:
  - compute all solutions
  - choose best

- Branch-and-bound approach:
  - compute first solution
  - add “betterness” constraint to open nodes
  - next solution will be “better”
  - prunes search space
Branch-and-bound Search

Find first solution
Branch-and-bound Search

- Explore with additional constraint
Branch-and-bound Search

 Guarantees better solutions
Branch-and-bound Search

- Last solution best
Branch-and-bound Search

Proof of optimality
Modelling SMM++

- Constraints and branching as before
- Order among solutions with constraints
  - so-far-best solution: \( S, E, N, D, M, O, T, Y \)
  - current node: \( S, E, N, D, M, O, T, Y \)
  - constraint added
    \[
    10000 \times M + 1000 \times O + 100 \times N + 10 \times E + Y
    \]
    \[
    <
    \]
    \[
    10000 \times M + 1000 \times O + 100 \times N + 10 \times E + Y
    \]
SMM++: Branch-and-bound

SEND
+ MOST
= MONEY

9782
+ 1094
= 10876
SMM: Strong Propagation

SEND
+ MORE
= MONEY

9567
+ 1085
= 10652
Acknowledgments

- I am grateful to Pierre Flener for helpful comments and bugreports on these slides