MDD Propagation for Sequence Constraints

Andre A. Cire and Willem-Jan van Hoeve
Tepper School of Business, Carnegie Mellon University

Sequence constraint

Example: Employee must work between 4 and 7 days every 9 consecutive days

| Sun | Mon | Tue | Wed | Thu | Fri | Sat | Sun | Mon | Tue | Wed | Thu | Fri | Sat | Sun | Mon | Tue | Wed | Thu | Fri | Sat | Sun | Mon | Tue | Wed | Thu | Fri | Sat | Sun | Mon | Tue | Wed | Thu | Fri | Sat |
|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
| x₁  | x₂  | x₃  | x₄  | x₅  | x₆  | x₇  | x₈  | x₉  | x₁₀ | x₁₁ | x₁₂ |

4 ≤ x₁ + x₂ + ... + x₉ ≤ 7
4 ≤ x₂ + x₃ + ... + x₁₀ ≤ 7
4 ≤ x₃ + x₄ + ... + x₁₁ ≤ 7
4 ≤ x₄ + x₅ + ... + x₁₂ ≤ 7

=: Sequence([x₁, x₂, ..., x₃], q=9, S={1}, l=4, u=7)

\[
\text{Sequence}(X, q, S, l, u) := \bigwedge_{i} l \leq \sum_{x \in X} (x \in S) \leq u
\]

Exact MDD Representation

Exact MDD for Sequence(X, q=3, S={1}, l=1, u=2)

MDD-Based CP

- Maintain limited-width MDD
  - Serves as relaxation
  - Typically start with width 1 (initial variable domains)
  - Dynamically adjust MDD, based on constraints

- Constraint Propagation
  - Edge filtering: Remove provably inconsistent edges (those that do not participate in any solution)
  - Node refinement: Split nodes to separate edge information

- Search
  - As in classical CP, but can be guided by MDD

Hardness of MDD Consistency

Goal: Given an arbitrary MDD and a Sequence constraint, remove all inconsistent edges from the MDD

Theorem: Establishing MDD consistency for Sequence on an arbitrary MDD is NP-hard

Proof: Reduction from 3-SAT
- Literal xᵢ in clause cᵢ represented by variable yᵢ
- MDD size O(6^((2mn+1)))
- Ensure that a variable takes the same value in each clause:

\[
\text{Sequence}(Y, q=2n, S={1}, l=n, u=n)
\]

Example:

\[
c₁ = (x₁ \lor \overline{x₃} \lor \overline{x₄})
\]

\[
c₂ = (x₂ \lor x₃ \lor \overline{x₄})
\]

Experimental Results

Random instances

Shift scheduling problems

Table: 

<table>
<thead>
<tr>
<th>Instance</th>
<th>Domain Filtering</th>
<th>MDD - width 1</th>
<th>MDD - width 2</th>
<th>MDD - width 4</th>
<th>MDD - width 8</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>time</td>
<td>time</td>
<td>time</td>
<td>time</td>
<td>time</td>
</tr>
<tr>
<td></td>
<td>backtracks</td>
<td>backtracks</td>
<td>backtracks</td>
<td>backtracks</td>
<td>backtracks</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Type P I</td>
<td>40</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
<td>17,054</td>
</tr>
<tr>
<td>Type P II</td>
<td>40</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
</tr>
<tr>
<td></td>
<td>60</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
</tr>
<tr>
<td></td>
<td>70</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
</tr>
<tr>
<td></td>
<td>80</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
<td>126,406</td>
</tr>
</tbody>
</table>

- Domain Filtering
- MDD - width 1
- MDD - width 2
- MDD - width 4
- MDD - width 8

- Time [s] – width 32