A Reactive Strategy for High-Level Consistency During Search

R.J.Woodward¹,² B.Y.Choueiry¹ C.Bessiere²
¹Constraint Systems Laboratory • University of Nebraska-Lincoln • USA
²CNRS • Université de Montpellier • France

1. Background: Local Consistency

Variables: A, B, C
Domains: {1, 2}
With: A = B, B = C, A + C < 4
{(A,1), (B,1), (C,1)} is a solution

Generalized Arc Consistency (GAC) ensures a value in the domain of a variable in the scope of a relation can be extended to a tuple satisfying the relation. E.g., all values are GAC.

Singleton Arc Consistency (SAC) removes 2 from domains of A, B, C.
SAC is an example of a High Level Consistency (HLC).

Enforcing consistency during search
• The higher the consistency level, the stronger the pruning and the smaller the search space.
• However, HLC can be costly in time and space.

2. Our View

The challenge is decide when, where, and how much HLC to enforce during search.

<table>
<thead>
<tr>
<th>Where?</th>
<th>When?</th>
<th>How much?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entire CSP</td>
<td>One variable</td>
<td>Always GAC</td>
</tr>
<tr>
<td>Always HLC</td>
<td>Always GAC</td>
<td>Stop early</td>
</tr>
<tr>
<td>Until fixpoint</td>
<td>HLC</td>
<td>When?</td>
</tr>
</tbody>
</table>

3. Our Solution

PREPEAK⁺, a simple and effective reactive strategy that
• Monitors search performance
• When search starts thrashing, triggers an HLC
• Then, conservatively reverts to GAC

We validate PREPEAK⁺
• With POAC as HLC (stronger than SAC) [Bennaceur+ CP 2001]
• Using the APOAC algorithm [Balafrej+ AAAI 2014]

4. PREPEAK⁺

When

<table>
<thead>
<tr>
<th>HLC</th>
<th>Peak depth</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>20</td>
</tr>
<tr>
<td>20</td>
<td>25</td>
</tr>
</tbody>
</table>

PREPEAK⁺ = PREPEAK + ‘How Much’

How Much
Terminate HLC as soon as either:
• Half the propagation queue is processed or
• HLC has consumed a total CPU time $\frac{2}{3}$ Time(GAC)

5. Empirical Evaluations

PREPEAK⁺, APOAC, and GAC compared on the number of cpu time and using dom/wdeg.

6. Visualization

Experiments conducted on equipment of Holland Computing Center (UNL).
Research support:
NSF Grants No. RI-111795 and RI-1619344
NSF Graduate Research Fellowship Grant No.1041000 and Chateaubriand Fellowship (R.J. Woodward)
ANR projects Demograph (ANR-16-CE40-0028) and Contredo (ANR-16-CE33-0024) (C. Bessiere)