Incremental Answer Completion in XSB-Prolog

Terrance Swift, Alexandre Miguel Pinto, Luís Moniz Pereira
CENTRIA - New University of Lisbon

ICLP’09, Pasadena, July 14th
### Outline

- **XSB-Prolog and the WFS**
- **Detecting unfounded SCCs**
- **Deleting unfounded answers**
- **Conclusions**
XSB-Prolog and the WFS

XSB-Prolog’s SLG-WAM implements the Well-Founded Semantics

But, there are some particular cases where the current SLG-WAM considers “undefined” literals which should be “true” (or “false”)

Those cases only occur in some particular unfounded Strongly Connected Components (SCCs)

Tuesday, July 14, 2009
Detecting unfounded SCCs

\[-\text{table } p/O, r/O, s/O.\]

\[p:\neg p.\]
\[s:\neg \neg r.\]
\[r:\neg \neg s.\]

<table>
<thead>
<tr>
<th>Subg</th>
<th>Answers</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>&lt;p, [p^p]&gt;</td>
<td>Comp</td>
</tr>
<tr>
<td>s</td>
<td>&lt;s, [ ]&gt;</td>
<td>Comp</td>
</tr>
<tr>
<td>r</td>
<td></td>
<td>Comp</td>
</tr>
</tbody>
</table>
Deleting unfounded answers

When an unfounded answer is found:

- delete it

- propagate the effects of its deletion to other answers that depend on it (Simplification)

- re-iterate until fixed-point is reached
Conclusions

- **SCC detection complexity is in**
  \[ \#\text{atoms}(P) \times \#\text{rules}(P) \]
  which is the known complexity of the WFS

- **SLG-WAM already does all it is possible to prevent such cases from occurring**

- **Performance:** at most, 18% increase in computation time

- **Current work:** implementation of heuristics to reduce unnecessary calls to Answer Completion
Thank You!