Monitoring and Visualizing Answer Set Solving

Arne König and Torsten Schaub

University of Potsdam
Outline

1. Motivation
2. Visualization
3. Clavis
4. Summary
1 Motivation
2 Visualization
3 Clavis
4 Summary
Motivation

- Declarative encoding
- Two-part computation
- Solving: conflict-driven search with learning
Declarative encoding

Two-part computation

First-Order Encoding

Grounder

Propositional Program

Solver

Stable Models

Solving: conflict-driven search with learning
Motivation

- Declarative encoding
- Two-part computation
- Solving: conflict-driven search with learning loop
  
  propagate
  
  \[ \text{if no conflict then} \]
  
  \[ \text{if all variables assigned then return variable assignment} \]
  
  else \text{ decide} \]

else

\[ \text{if top-level conflict then return unsatisfiable} \]

else

\text{analyze}

\text{backjump}
Motivation

- Declarative encoding
- Two-part computation
- Solving: conflict-driven search with learning
- Effective and robust
- Lacking insights
Motivation

- Declarative encoding
- Two-part computation
- Solving: conflict-driven search with learning
- Effective and robust
- Lacking insights
Outline

1. Motivation
2. Visualization
3. Clavis
4. Summary
## Approach

### Goals
- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

### Available Data
- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
**Approach**

**Goals**
- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

**Available Data**
- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
Approach

Goals

- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

Available Data

- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
Visualization

Approach

Goals

- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

Available Data

- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
Visualization

Approach

Goals

- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

Available Data

- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
Approach

Goals

- Explore the solving process
- Reconnect problem and solving
- Provide dynamic perspectives

Available Data

- Algorithmic Figures (backjump length, conflict level)
- Variable Properties (decisions, conflict involvement, symbols)
- Structural Data (program constraints, learnt constraints)
Outline

1. Motivation
2. Visualization
3. Clavis
4. Summary
Clavis

Propositional Program

clasp

Stable Models

logger

Logfile

Insight

Clavis

Configurable event logging (conflicts, backjumps, restarts, ...)

Insight

Offline visualization
Clavis

Propositional Program

clasp

logger

Stable Models

Logfile

Insight

Clavis

Configurable event logging (conflicts, backjumps, restarts, . . .)

Insight

Offline visualization
Diagrams

Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
- Force-directed layout
- Overlay with other graphs
- Color with variable properties
Diagrams

Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
- Force-directed layout
- Overlay with other graphs
- Color with variable properties
Diagrams

Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
  - Examples: program interaction, learned interaction
  - Force-directed layout
  - Overlay with other graphs
  - Color with variable properties
Diagrams

Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
  - Force-directed layout
  - Overlay with other graphs
  - Color with variable properties
Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
- Force-directed layout
- Overlay with other graphs
- Color with variable properties
Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
- Force-directed layout
- Overlay with other graphs
- Color with variable properties
Diagrams

Two-dimensional plots

Example: constraint length (per learned constraint)

Networks

- Interaction graphs
- Examples: program interaction, learned interaction
- Force-directed layout
- Overlay with other graphs
- Color with variable properties
Example
Example
### Example

<table>
<thead>
<tr>
<th>id</th>
<th>type</th>
<th>Decisions</th>
<th>symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>atom/body</td>
<td>1</td>
<td>move(4,c,1) on(4,c,1) ...</td>
</tr>
<tr>
<td>1</td>
<td>atom/body</td>
<td>0</td>
<td>move(4,b,1) on(4,b,1) ...</td>
</tr>
<tr>
<td>2</td>
<td>atom/body</td>
<td>0</td>
<td>move(4,c,2)</td>
</tr>
<tr>
<td>3</td>
<td>atom/body</td>
<td>6</td>
<td>move(3,c,2)</td>
</tr>
<tr>
<td>4</td>
<td>atom/body</td>
<td>0</td>
<td>move(4,b,2)</td>
</tr>
<tr>
<td>5</td>
<td>atom/body</td>
<td>0</td>
<td>move(3,b,2)</td>
</tr>
<tr>
<td>6</td>
<td>atom/body</td>
<td>0</td>
<td>move(4,a,2)</td>
</tr>
<tr>
<td>7</td>
<td>atom/body</td>
<td>0</td>
<td>move(4,c,3)</td>
</tr>
<tr>
<td>8</td>
<td>atom/body</td>
<td>0</td>
<td>move(3,c,3)</td>
</tr>
</tbody>
</table>
Example
Example
Example
Example
Example
More Features

- Query language
- Multiple layout algorithms
- Documented log file format and library
More Features

- Query language
- Multiple layout algorithms
- Documented logfile format and library
More Features

- Query language
- Multiple layout algorithms
- Documented logfile format and library
Outline

1. Motivation
2. Visualization
3. Clavis
4. Summary
ASP separates encoding and solving

- Benefits: Ease of use and efficient solving
- Drawback: reduces traceability

Reuniting encoding and solving with visualization

- Helps understanding of solving
- Try clavis!
ASP separates encoding and solving

- **Benefits:** Ease of use and efficient solving
- **Drawback:** reduces traceability

Reuniting encoding and solving with visualization

- Helps understanding of solving
- Try clavis!
Summary

ASP separates encoding and solving

- **Benefits**: Ease of use and efficient solving
- **Drawback**: reduces traceability

Reuniting encoding and solving with visualization

- Helps understanding of solving
- Try clavis!
Summary

ASP separates encoding and solving

- **Benefits**: Ease of use and efficient solving
- **Drawback**: reduces traceability

Reuniting encoding and solving with visualization

- Helps understanding of solving
- Try clavis!
Summary

ASP separates encoding and solving

- **Benefits**: Ease of use and efficient solving
- **Drawback**: reduces traceability

Reuniting encoding and solving with visualization

- **Helps understanding of solving**
- **Try clavis!**
ASP separates encoding and solving

- **Benefits:** Ease of use and efficient solving
- **Drawback:** reduces traceability

Reuniting encoding and solving with visualization

- Helps understanding of solving
- Try clavis!

http://cs.uni-potsdam.de/clavis
http://potassco.sf.net