GrowingLeaf: Supporting Requirements Evolution over Time

Alicia M. Grubb, Gary Song, Marsha Chechik
{amgrubb, gary, chechik}@cs.toronto.edu

iStar’16 - Sep. 12, 2016
• Assumptions of early-phase requirements modeling:
  ➪ all model elements have a value
  ➪ model values are constant
• In reality intentions and relationships in the environments are not constant.
Example Questions

1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A’s dependums affect the Actor-A’s root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?
Example Questions

1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A’s dependums affect the Actor-A’s root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?
Example Questions

1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A's dependums affect the Actor-A's root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?

Use Qualitative Evaluation Labels with Forward Analysis and Backward Analysis
Example Questions

1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A’s dependums affect the Actor-A’s root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?
1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A’s dependums affect the Actor-A’s root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?
Example Questions

1. Is it possible to satisfy Goal-A and partially satisfy Goal-B? and how?

2. How does completing Task-A and Task-B but not Task-C affect the top level goals?

3. How do changes in Actor-A’s dependums affect the Actor-A’s root-level goals over time?

4. Which possible scenarios always satisfy Goal-A even if Goal-B becomes denied in the future?

5. Does the satisfaction order of Goal-C and Goal-D matter?
Example Questions

1. Is it possible to satisfy \textit{Goal-A} and partially satisfy \textit{Goal-B}? and how?

2. How does completing \textit{Task-A} and \textit{Task-B} but not \textit{Task-C} affect the top level goals?

3. How do changes in \textit{Actor-A’s} dependums affect the \textit{Actor-A’s} root-level goals over time?

4. Which possible scenarios \textit{always} satisfy \textit{Goal-A} even if \textit{Goal-B} becomes denied in the future?

5. Does the satisfaction order of \textit{Goal-C} and \textit{Goal-D} matter?
Contributions

Provide tooling to:

• enrich goal models intentions with dynamically changing evaluation

• analyze the impacts of dynamically changing intentions on decision making
Why another modeling tool?

- Why another modeling tool?
Why another modeling tool?
Why another modeling tool?
Why another modeling tool?

OpenOME
Why another modeling tool?

- Surveyed previous tools
  - Extend their iStar meta-model
  - Add icons/labels on top of intentions
- Web-based tool
  - Framework vs. self-built
  - Multi-view vs. multi-tab
Introducing GrowingLeaf
Introducing GrowingLeaf
Introducing GrowingLeaf

Step 1: Leaf Sim
Outline

- Modeling Problem and Tool Justification
- Tool Introduction
- Dynamic Intentions and Analysis
- Tool Functionality
- Discussion and Validation
- Status and Future Work
Modeling Dynamic Intentions

Diagram showing relationships between various activities and outcomes, such as:
- Use Current Dump
- Use New Dump
- Manage City Waste
- Comply with Standards
- Build Large Dump
- Build Small Dump
- Purchase Land
- Have Workers Union Contract
- Update Truck Route
- Upgrade Trucks
- Space in Dump
- Process Green Waste
- Reduce Operating Costs
- Build Green Centre
- GW Education Program
- Positive City Image
- Enjoy City
- Quality of Waste Separation
- Environmental Concern
- Willingness to Separate Waste

Arrows indicate dependencies, helps, and hurts between these elements.
Modeling Dynamic Intentions
Modeling Dynamic Intentions

Stochastic (R)

Patterns:

Examples:

Environmental Concern
Modeling Dynamic Intentions

Elementary Functions

Stochastic (R):

Increase (I):

Decrease (D):

Constant (C):

or
Modeling Dynamic Intentions

Denied-Satisfied (DS)

Patterns:

Examples:

- Build Small Dump
- Build Large Dump
- Build Green Centre
Modeling Dynamic Intentions

Denied-Satisfied (DS)

Patterns:

[Diagram showing patterns with a checkmark and an X]

Examples:

1. Build Small Dump
2. Build Large Dump
3. Build Green Centre
Modeling Dynamic Intentions

Monotonic Negative (MN)

Patterns:

Examples:

Space in Dump
## Common Compound Functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Denied-Satisfied (DS)</td>
<td>the satisfaction evaluation remains <strong>Denied</strong> until $t_i$ and then remains <strong>Satisfied</strong></td>
</tr>
<tr>
<td>Monotonic Negative (MN)</td>
<td>changes in satisfaction evaluation become “less true” to a $maxValue$ at $t_i$ and then remains constant at $constantValue$</td>
</tr>
</tbody>
</table>
# Common Compound Functions

<table>
<thead>
<tr>
<th>Function Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satisfied-Denied (SD)</td>
<td>the satisfaction evaluation remains <em>Satisfied</em> until $t_i$ and then remains <em>Denied</em></td>
</tr>
<tr>
<td>Denied-Satisfied (DS)</td>
<td>the satisfaction evaluation remains <em>Denied</em> until $t_i$ and then remains <em>Satisfied</em></td>
</tr>
<tr>
<td>Stochastic-Constant (RC)</td>
<td>changes in satisfaction evaluation are stochastic or random until $t_i$ and then remains constant at $constantValue$</td>
</tr>
<tr>
<td>Constant-Stochastic (CR)</td>
<td>the satisfaction evaluation remains constant at $constantValue$ until $t_i$ and then changes in evaluation are stochastic or random</td>
</tr>
<tr>
<td>Monotonic Positive (MP)</td>
<td>changes in satisfaction evaluation become “more true” to a $maxValue$ at $t_i$ and then remains constant at $constantValue$</td>
</tr>
<tr>
<td>Monotonic Negative (MN)</td>
<td>changes in satisfaction evaluation become “less true” to a $maxValue$ at $t_i$ and then remains constant at $constantValue$</td>
</tr>
</tbody>
</table>
Modeling Dynamic Intentions

User Defined (UD)

Repeating Function

Have Workers Union Contract
Analysis Strategies

(Strategy 1: Leaf Simulation) create a random path given initial states in the model

(Strategy 2: CSP Analysis) create a path given desired properties of the intermediate state (with optional properties over the initial or final state)

(Strategy 3: CSP History) create a path which is different than the previously seen path over the same constraints
Outline

- Modeling Problem and Tool Justification
- Tool Introduction
- Dynamic Intentions and Analysis
- **Tool Functionality**
- Discussion and Validation
- Status and Future Work
GrowingLeaf - Modeling Demo
GrowingLeaf - Modeling Demo

Modelling Relationships
Modelling Relationships

- Node name: Positive City Image
  - Initial Satisfaction Value: None
GrowingLeaf - Modeling Demo Summary

• Drag and drop interface
• Naming and adding elements
• Loading, saving, exporting, and zooming models
• Resizing label fonts
• Changing initial satisfaction values
• Changing dynamic function types
• Creating User Defined functions
GrowingLeaf - Analysis Demo
GrowingLeaf - Analysis Demo Summary

- How to run analysis
- Adjust simulation length
- Types of analysis
- Scrolling through analysis results
Improving Analysis with Constraints

- Undesirable results due to EB ordering
- Add constraints over EB order
  - Adding model links is inappropriate
  - Test relationship before updating the model
- Used on rare occasions
• Adding constraints between EBs
Outline

• Modeling Problem and Tool Justification

• Tool Introduction

• Dynamic Intentions and Analysis

• Tool Functionality

• Discussion and Validation

• Status and Future Work
Architecture

Client Side (Front-end)

- Chrome Browser
  - JointJS
  - Javascript

Tool Page Request

Server Side (Back-end)

- Application Server
  - Java
- File System Data Storage

Analysis (CGI in Python)
Design Decisions

- Browser versions and updates
- JointJS data model and constrains
Usability

• Two rounds user testing

• Found issues with
  • resizing
  • ‘enter’ key
  • ‘backspace’/‘delete’ key
  • selecting analysis techniques

• Further user studies are ongoing

• Built several models and examples
Examples and Case Studies

• City transportation planning
• Network maintenance
• Software supply chains
• Technical debt
• Compliance
• Sustainability

Further case studies are ongoing....
Ongoing Validation

• Evaluate usability / effectiveness with controlled experiment

• Prototype study at this week at iStar and RE

• Please Participate!!

http://www.cs.toronto.edu/~amgrubb/restudy.htm
Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf
Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf
Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf
Where do I get the tool?

[Image of the GrowingLeaf tool interface]
Where do I get the tool?

Use Google Chrome
Where do I get the tool?

http://www.cs.toronto.edu/~amgrubb/growing-leaf

Join the development team.
Future Work

• Update tool to use iStar 2.0 Language Guide
• External industrial case study
• Improve server connection (security)
• Multiple users to simultaneously edit
• Development for other browsers
Questions?
GrowingLeaf: Supporting Requirements Evolution over Time

GrowingLeaf
http://www.cs.toronto.edu/~amgrubb/growing-leaf

Tool Study at RE’16:
http://www.cs.toronto.edu/~amgrubb/restudy.htm

Alicia M. Grubb
amgrubb@cs.toronto.edu