VISPEC: A graphical tool for elicitation of MTL requirements

Bardh Hoxha, Nikolaos Mavridis and Georgios Fainekos

IEEE/RSJ International Conference on Intelligent Robots and Systems
Hamburg, Germany - September 2015

School of Computing, Informatics and Decision System Engineering
Arizona State University

bardhh@asu.edu

http://www.public.asu.edu/~bhoxha
Safety Critical Systems

Rehabilitation Robots

Image retrieved from:

Autonomous Vehicles

Image retrieved from:
http://cdn1.itpro.co.uk/sites/itpro/files/2015/06/google_driveless_car.jpg

Medical Devices


Disaster R&R Robots

Safety Critical Systems

Rehabilitation Robots

Medical Devices

Autonomous Vehicles

Disaster R&R Robots

Safety is of Paramount Importance
Safety Critical Systems
Safety Critical Systems

New System

Market
Safety Critical Systems

ISO
RTCA
EUROCAE
...

New System → Testing, Verification and Certification Process → Market
Safety Critical Systems

ISO
RTCA
EUROCAE
...

New System

Testing, Verification and Certification Process

S-TaLiRo, Fapas, SpaceEx
CheckMate, Flow, Breach, C2E2, KeYmaera
...

Market
Testing and verification

System

Specifications

Natural Language
Structured language, Unified Modeling Language, Message Sequence Charts ...

Informal

Semi-Formal

Formal

Logics: LTL, MTL, CTL, ....

Structured language, Unified Modeling Language, Message Sequence Charts ...

Informal

Semi-Formal

Formal

Logics: LTL, MTL, CTL, ....
Testing and verification

\begin{align*}
\phi_1 &= \neg F_{[0,100]} G_{[0,1]} (\text{fuelflowrate} = 0) \\
\phi_2 &= G((\lambda \text{ out of bounds}) \\
\rightarrow F_{[0,1]} G_{[0,1]} \neg (\lambda \text{ out of bounds}) \\
\phi_3 &= F_{[0,T]} ((v \geq \hat{v}) \land (\omega < \hat{\omega})) \\
\phi_4 &= \bigwedge_{i=1}^{4} G((\neg g_i \land X g_i) \rightarrow G_{[0.025,2.5]} g_i)
\end{align*}
Testing and verification

System

Specifications

Natural Language

Structured language, Unified Modeling Language, Message Sequence Charts ...

Informal

Semi-Formal

Logics: LTL, MTL, CTL, ...

Formal
Testing and verification

System

Specifications

Natural Language
Structured language, Unified Modeling Language, Message Sequence Charts ...

Potentially:
Ambiguous, Inconsistent, Imprecise

Semi-Formal

Logics:
LTL MTL CTL .... Formal
Overview of Contributions
Overview of Contributions
Overview of Contributions

- We present a graphical formalism that enables the development of formal specifications.
Overview of Contributions

• We present a graphical formalism that enables the development of formal specifications.

• We present a tool based on the graphical formalism.
Overview of Contributions

• We present a graphical formalism that enables the development of formal specifications.
• We present a tool based on the graphical formalism.
• We conducted a usability study to evaluate the tool.
Overview of Contributions

• We present a graphical formalism that enables the development of formal specifications.
• We present a tool based on the graphical formalism.
• We conducted a usability study to evaluate the tool.
• We present applications of the tool for real-world robots.
ViSpec - Visual Specification Tool
ViSpec - Visual Specification Tool

• Goals for the tool:
  • Intuitive to use
ViSpec - Visual Specification Tool

- Goals for the tool:
  - Intuitive to use
  - Does not have a high learning curve
ViSpec - Visual Specification Tool

• Goals for the tool:
  • Intuitive to use
  • Does not have a high learning curve
  • Wide class of specifications
ViSpec - Visual Specification Tool

• Goals for the tool:
  • Intuitive to use
  • Does not have a high learning curve
  • Wide class of specifications
  • Translate the graphical formalism to a formal language (Metric Temporal Logic)
ViSpec - Visual Specification Tool

• Goals for the tool:
  • Intuitive to use
  • Does not have a high learning curve
  • Wide class of specifications
  • Translate the graphical formalism to a formal language (Metric Temporal Logic)

• Development challenges
  • Expressivity vs. ease of use
Metric Temporal Logic: Semantic Intuition

Syntax: Boolean connectives with temporal operators

\[ \phi ::= T \mid \neg \phi \mid \phi_1 \lor \phi_2 \mid G \phi \mid F \phi \mid \phi_1 U \phi_2 \]

*G a* - always a

*F a* - eventually a

*a U b* - a until b

*a U_{[1,1.5]} b* - a until b

\(0 \quad 0.4 \quad 0.7 \quad 1.1 \quad 1.2 \quad 1.7\)
ViSpec - Visual Specification Tool
Automotive example:

NL Requirement: In the next 36 seconds, engine speed should always be less than 4000

ViSpec:

\[ \phi = G (rpm < 4000) \]
ViSpec – Specification Classes

Safety:
\[ G_I \phi \]

Reachability:
\[ F_I \phi \]

Stabilization:
\[ F_I G_I \phi \]

Recurrence:
\[ G_I F_I \phi \]

Implication:
\[ \phi \rightarrow \psi \]

Reactive Response:
\[ N_I (\phi \rightarrow M_I \psi) \]

Conjunction:
\[ \phi \land \psi \]

Non-strict Sequencing:
\[ N_I (\phi \land M_I \psi) \]

\[ M \in \{G, F\}, N \in \{G, F\} \]
ViSpec – Usability Study

Goal:
Evaluate whether ViSpec enables users to develop formal specifications

Two Cohorts

Cohort I: Non-expert users
No experience in working with requirements.
20 subjects from the student community at ASU

Cohort II: Expert users
Experienced in working with requirements (not necessarily formal requirements)
10 subjects from the industry in the Phoenix area
ViSpec – Usability Study
ViSpec – Usability Study

Questions investigated:

• Whether the graphical formalism enables non-experts and experts to formalize requirements accurately
ViSpec – Usability Study

Questions investigated:
• Whether the graphical formalism enables non-experts and experts to formalize requirements accurately
• How well the expert cohort performs in comparison to the non-expert cohort
ViSpec – Usability Study

Questions investigated:

• Whether the graphical formalism enables non-experts and experts to formalize requirements accurately
• How well the expert cohort performs in comparison to the non-expert cohort
• How user friendly and easy-to-use ViSpec is
ViSpec – Usability Study
ViSpec – Usability Study

Experimental Design:

• At most 45 minutes
ViSpec – Usability Study

Experimental Design:

• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
ViSpec – Usability Study

Experimental Design:
• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
ViSpec – Usability Study

Experimental Design:

• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
• Tasks asked the subject to formalize a natural language specification through ViSpec
ViSpec – Usability Study

Experimental Design:
• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
• Tasks asked the subject to formalize a natural language specification through ViSpec
• Tasks become more complex throughout the session
ViSpec – Usability Study

Experimental Design:

• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
• Tasks asked the subject to formalize a natural language specification through ViSpec
• Tasks become more complex throughout the session
• Computer screen and actions recorded for each session.
ViSpec – Usability Study

Experimental Design:

• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
• Tasks asked the subject to formalize a natural language specification through ViSpec
• Tasks become more complex throughout the session
• Computer screen and actions recorded for each session.
• Subjects completed a demographic and post-completion questionnaire
ViSpec – Usability Study

Experimental Design:
• At most 45 minutes
• Subjects received a one minute and thirty second tutorial
• Each user received ten tasks
• Tasks asked the subject to formalize a natural language specification through ViSpec
• Tasks become more complex throughout the session
• Computer screen and actions recorded for each session.
• Subjects completed a demographic and post-completion questionnaire

Example Task (Recurrence):
NL: At every point in time in the first 40 seconds, vehicle speed will go over 100 in the next 10 seconds.
MTL: $G_{[0,40]}F_{[0,10]}(speed>100)$
ViSpec – Usability Study
ViSpec – Usability Study

Two metrics used for performance evaluation:
• Task Completion: a binary measure.
Two metrics used for performance evaluation:

- **Task Completion**: a binary measure.
- **Task accuracy**: a value from 1 to 5. Answers graded by formal specification experts using the following criteria:
ViSpec – Usability Study

Two metrics used for performance evaluation:

• Task Completion: a binary measure.

• Task accuracy: a value from 1 to 5. Answers graded by formal specification experts using the following criteria:
  • How accurate the meaning of the natural language specification is captured.
ViSpec – Usability Study

Two metrics used for performance evaluation:

• Task Completion: a binary measure.

• Task accuracy: a value from 1 to 5. Answers graded by formal specification experts using the following criteria:
  • How accurate the meaning of the natural language specification is captured.
  • Whether the inaccuracies in the user submitted formula can be easily debugged and corrected in the testing and verification process.
ViSpec – Results

Average grade per task:

We test the hypothesis that:

Non-expert and Expert users can define formal requirements accurately using the Visual Specification Tool.
## ViSpec – Improvements

<table>
<thead>
<tr>
<th>#</th>
<th>Improve...</th>
<th>Prime Indicators</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>the process of creating child templates</td>
<td>Misclicks. User feedback</td>
</tr>
<tr>
<td>2</td>
<td>the tutorial by placing more emphasis on the difference between implication and conjunction when connecting the templates</td>
<td>User generated specifications, User feedback, Task accuracy grade</td>
</tr>
<tr>
<td>3</td>
<td>the visual representation of grouped templates</td>
<td>User generated specifications, User feedback, Task accuracy grade</td>
</tr>
<tr>
<td>4</td>
<td>the Template setup assistant</td>
<td>User Feedback, User thought map</td>
</tr>
</tbody>
</table>
ViSpec – Application

Serial link manipulator for robotic surgery

♦ Puncturing action

♦ NL: The force applied to the patient by the end effector is always less than a given threshold, except for the puncturing subtask.

♦ MTL:

\[ G_{[0,30]}(\neg \text{puncturing} \rightarrow f \leq f_{\text{max}}) \]

♦ ViSpec:

Conclusions
Conclusions

• We presented a graphical formalism and tool that enables users to easily develop formal specifications.
Conclusions

• We presented a graphical formalism and tool that enables users to easily develop formal specifications.

• The ViSpec tool enables users who have little to no mathematical training in formal logics to develop formal specifications, as indicated by a usability study.
Conclusions

• We presented a graphical formalism and tool that enables users to easily develop formal specifications.

• The ViSpec tool enables users who have little to no mathematical training in formal logics to develop formal specifications, as indicated by a usability study.

• The tool was utilized to formalize specifications for robotic applications.
Future Work
Future Work

• Expand the usability study to experts in formal languages.
  • We have some preliminary results
  • We need more data! Please participate through
    http://goo.gl/forms/MJtLAa6nNe
Future Work

• Expand the usability study to experts in formal languages.
  • We have some preliminary results
  • We need more data! Please participate through
    http://goo.gl/forms/MJtLAa6nNe

• Develop automatic debugging tools that would enable users to test for common problems in specification elicitation.

Metric Interval Temporal Logic Specification Elicitation and Debugging, Adel Dokhanchi, Bardh Hoxha and Georgios Fainekos, MEMOCODE 2015, Austin, Texas
Future Work

• Expand the usability study to experts in formal languages.
  • We have some preliminary results
  • We need more data! Please participate through
    
    http://goo.gl/forms/MJtLAa6nNe

• Develop automatic debugging tools that would enable users to test for common problems in specification elicitation.

  Metric Interval Temporal Logic Specification Elicitation and Debugging,
  Adel Dokhanchi, Bardh Hoxha and Georgios Fainekos,
  MEMOCODE 2015, Austin, Texas

• Expand the set of specifications supported by the graphical formalism
Acknowledgements

Awards:
CNS-1319560, CNS-1116136,
IIP-1454143, IIP-1361926

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Thank you!

Survey Link:
http://goo.gl/forms/MJtLAA6nNe