Challenges in Robotics Software Engineering

Ivano Malavolta (Vrije Universiteit Amsterdam)
Sara Ljungblad (University of Gothenburg)
Goal

To elaborate together on the MAIN CHALLENGES of robotics software engineering

1. Provided initial set of challenges
2. Clustered the identified challenges
3. Identify the design space of selected challenges (challenges + candidate solutions)
RoSE 2018 Workshop Survey

Dear workshop participant,

to foster discussion of robotics challenges and solutions during our 4th session (cf. http://www.tinyurl.com/rose2018), we’d like to collect your thoughts and ideas beforehand.

Please note that all fields are optional.

Thank you very much!

Federico Ciccozzi
Davide Di Russo
Ivano Mavolot
Patrizio Pelliccione
Andreas Wortmann

What are the most important challenges in robotics?
Your answer

How can software engineering solve the important challenges in robotics?
Your answer

What is still missing for SE in robotics (a tool with specific functionalities, a method that supports specific challenges, ...)?
Your answer
Demographics

10 participants

In which kind of organization are you working?
- Both: 10.0%
- Industry: 20.0%
- Research: 70.0%

What is your main area of expertise?
- Software Engineering: 60.0%
- The intersection: 30.0%
- Robotics: 10.0%
The “corpus”

Which challenges of robotics software projects are fundamentally different from traditional software projects?
6 responses

How can software engineering solve the important challenges in robotics?
9 responses

What challenges of robotics software projects will software engineering maybe never able to solve?
5 responses

How can we foster the integration of the SE community and the robotics community?
8 responses

What are the most important challenges in robotics?
10 responses

integrating software, mechanical, electrical engineering, etc.

trustworthiness, compositability, expense of development

Evaluating the environment

Integrating modern software deployment technologies (virtualization, cloud, etc.) in robotics.

See robotics community roadmap (for challenges and opportunities)

I don’t know. I guess some relate to feature interaction and context data. Lots of ad hoc development in robotics; not so much standardization.

What is still missing for SE in robotics (a tool with specific functionalities, a method that supports specific challenges, ...)?
7 responses

only recently have the SE platforms matured to be widespread and open (ROS). We are still missing good descriptions of middleware technologies.

Better understanding of SE pain points within robotics. Deeper semantic connections between code and the real world, better abstractions to represent possible (likely?) geometric configurations of systems within environments.

Hardware abstraction frameworks; sensor fusion algorithms/architectures; hardware/environment simulation frameworks
Emerging challenges

- Uncertainty management
- (Legacy) code reuse
- Virtualization/containers
- Integration with hardware
- Analysis (model-based?)
- Tooling + integration with IDEs
- Testing and debugging
- Systems integration, middleware
- Project management, estimation
What are DESIGN considerations for Robotics Software Engineering?
Sara Ljungblad

Assistant professor, Dept. of Applied IT, University of Gothenburg.

- Design methodology
- Design skills
- Inclusive design
- Human Robot Interaction
- Human Computer Interaction

Critique towards robotic research: **Robot Cargo Cult**, Workshops on **user-centred design** of robotic artefact. Artist perspectives on robotic materials.

**Workpackage** leader (year 2010) in LIREC (Living with Robots and Interactive Companions, EU project 2008-2011) **User studies**

Publications in the area of Human Robot Interaction (2004-2016) Ro-Man, HRI, NordiChi, CSCW.
Studies of how robotic products are experienced:
- Long term perspectives of robotic toys adopted in families (Pleo)
- Hospital robot at work (Robcab)
- Experiences of eating aids and assisted meals (Bestic)
Some DESIGN considerations:
Keep a holistic approach
”How difficult it can be to train AI software to be consistent and robust”
See through the eyes of others
“This is not for me”

Negative and stereotype views of elderly in robot use scenarios

(Lee et al, 2016).
Question the question
Do you:

• Keep a holistic approach
• See through the eyes of others
• Question the question

How? ..or why not?
THANKS!

Applied Robotics in Gothenburg

- Newly started research group!
- User, stakeholder perspectives in robotic research
- Critical perspectives
  - Robot cargo cult
  - Empirically grounded ethics
Go to the whiteboard &
- put your name below the most interesting challenge for you
- add new challenges

Link to this presentation: https://goo.gl/3WGAr4
Raw data about the answers
What are the most important challenges in robotics?

10 responses

- Integrating software, mechanical, electrical engineering, + the ethics and awareness questions
- Trustworthiness, composability, expense of development

**Evaluating the environment**

- Integrating modern software deployment technologies (virtualization, containers), software defined networks and robotics.
- See robotics community roadmap (for challenges and opportunities)
- I don’t know. I guess some relate to feature interaction and correctness. Also mostly relate to control and less to data. Lots of ad hoc development in robotics; not so much systematic testing, reuse, automation of SE tasks.

**Reliable perception under wide operational conditions**

- The integration between existing components and legacy solution and new algorithms and approaches. Also, the development of complex and high level application exploiting the existing and consolidated robotic platforms.
- Code reusability; system interoperability
- Uncertainty is fundamental
How can software engineering solve the important challenges in robotics?

9 responses

- Systems thinking, leveraging requirements and modeling
- Code synthesis, formal methods, models, architectures
- Machine vision, object recognition, pattern recognition, sensor fusion, data analysis, machine learning, natural language processing
- Code generation, re-usable components, modern standardized APIs between sensors/actuators and applications
- Identify real needs of robotic field that our techniques are not handling well
- Providing effective tools with RAMS properties to roboticists
- Robotics could use a lot of existing SE techniques and technologies to improve the development of robotic applications. However, most of them needs to be adapted or adjusted to take in account specific challenges of robotics (i.e., the importance of the hardware layer)
- Create tools that can guarantee some properties of the system
- By defining design patterns and standard architectural approaches for robotics (maybe for specific robotic domains)
What is still missing for SE in robotics (a tool with specific functionalities, a method that supports specific challenges, ...)?

7 responses

- only recently have the SE platforms matured to be widespread and open (ROS). We are still missing good descriptions of middleware technologies.
- Better understanding of SE pain points within robotics. Deeper semantic connections between code and the real world. better abstractions to represent possible (likely?) geometric configurations of systems within environments.
- Hardware abstraction frameworks; sensor fusion algorithms/architectures; hardware/environment simulation frameworks
- ROS and Gazebo could be better integrated with development tools like Eclipse
- Analysis that can better handle environments, configurations, and uncertainty.
- A proper way to demonstrate how tools and framework at model level can be really deployed in a real robotics application ... possibly industrial or on a service robot
- Tooling for SE in robotics is particularly weak. For example, even debugging still is a difficult and complex task in robotics. Project management, code-base reuse and organization are heavily underused, too.
How can we foster the integration of the SE community and the robotics community?

8 responses

Teaching: code quality analysis; student clubs and projects; outreach via ROSE and others. Attending ROSCon was great for this.

On the SE side: build robot specific tools within existing infrastructures (like LLVM). On the robotics side, build SE modeling/analysis into existing tools (like Gazebo or MuJoCo).

Pursue the development of hardware-control libraries and hardware/environment-simulation libraries. The former will make it easier for the robotics community to leverage a wider pool of SE knowledge/talent, while the latter will aid the SE community in providing control/test systems for the robotics community. Over time, the distance between these two communities should close. In essence, take a similar approach to how Google, Amazon, Microsoft, etc. are making it easier for the wider SE community to develop Machine Learning applications, and also to help improve ML algorithms as a result.

Standardization and Open Source tools

Events like this workshop, but that can also be present at robotic conferences like iros and icra

One thing that could help would be if we have a set of projects whose code is open source (robotics projects for some open robotics platforms, not SE tools projects for robotics) that could be used as case studies and benchmarks.

Participate to robotics conferences submitting papers and participating to the debate this would provide real insights to the SE community of what the robotics challenges are

Both roboticists and software engineers should stop doing each other jobs. A roboticist can recognize the problems of his field, but he doesn’t have the knowledge and the skill to solve them efficiently. At the same time, software engineers cannot provide useful solution without the right knowledge of the underlying problems. Collaboration between experts of the two fields is the key.
Which challenges of robotics software projects are fundamentally different from traditional software projects?

<table>
<thead>
<tr>
<th>6 responses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real time aspects, low level programming</td>
</tr>
<tr>
<td>noise, uncertainty, interaction with physical environment</td>
</tr>
<tr>
<td>The physical interaction (sensing, perception, decision making) with a physical dynamic, underspecified, unknown environment.</td>
</tr>
<tr>
<td>Traditional software projects tend to abstract from the underlying hardware implementation. While this is partially possible in robotics (i.e., during the early stages of development), in the end it is always necessary to take in account the robotic platform used when deploying the software. In a way this is closer to embedded software development, but this tie between hardware and software may propagate to high level applications.</td>
</tr>
<tr>
<td>integration between software and hardware;</td>
</tr>
<tr>
<td>Uncertainty, safety, real-time control</td>
</tr>
</tbody>
</table>
What challenges of robotics software projects will software engineering maybe never able to solve?

5 responses

<table>
<thead>
<tr>
<th>Ethics and AI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Robotics include many element besides software, we cannot expect to handle them all.</td>
</tr>
<tr>
<td>Robotics aspects, indeed SE should never pretend to be Robotics. So SE will never solve any of the Robotics aspects of robotics, and vice versa.</td>
</tr>
<tr>
<td>All those challenges that require specific robotic knowledge. For those they will need the help of roboticists.</td>
</tr>
<tr>
<td>enable tools that will enable robot performances that are similar in simulation and on real robots</td>
</tr>
</tbody>
</table>