User-friendly specification of robotic missions

Claudio Menghi, Sergio Garcia, Patrizio Pelliccione, Thorsten Berger, Carlo Ghezzi, Christos Tsigkanos

{claudio.menghi,sergio.garcia,patrizio.pelliccione,thorsten.berger}@gu.se
{carlo.ghezzi,christos.tsigkanos}@polimi.it,
Robot evolution

Robotics Timeline

1960 - 1970

1970 - 1985

1975 - 1990

1990 - 2000

1980 - 1990

1990 - 2000

1995 - 2010

2000 - 2010

https://www.slideshare.net/robotsalive/introduction-to-mobile-robotics
Robot market

- Sales of professional service robots grows
- There are emerging markets for indoor logistics robotics (offices, hotels and hospitals)

Robotic missions specification
Robotic missions specification

• Precisely define the mission requirement a (team of) robot(s) should achieve

• Transform the mission requirement into a formal mission specification
  (mission specification problem)
Robotic missions specification

- Precisely define the mission requirement (team of) robot(s) should achieve.

- Transform the mission requirement into a formal mission specification (mission specification problem).

---

   Usability evaluation of high-level user assistance for robot mission specification.
   Transactions on Systems, Man, and Cybernetics, 2004

2. Federico Ciccozzi, Davide Di Ruscio, Ivano Malavolta, Patrizio Pelliccione
   Adopting MDE for Specifying and Executing Civilian Missions of Mobile Multi-Robot Systems
   IEEE Access, 2016
SE goals

• Providing systematic procedures and tools that help robotic developers in
  • defining the mission requirements
  • solving the mission specification problem
A simple example

• We want a robot to bring coffee to a user in an office

• How can we systematically support the specification of mission requirements?

• How can we convert a mission requirement into a formal mission specification?
Patterns: a software engineering solution

- Patterns are a SE solution for solving the mission specification problem

- **Mission Specification Pattern**: a pattern is a template solution for a recurrent mission specification problem
  - it maps a recurrent mission requirement to a mission specification
Identifying Specification Patterns: Methodology

- Collect mission requirements from literature
- Extract concerns capturing mission aspects from description
- Identify recurrent specification problems
- Provide model solutions to specification problems in the form of a pattern catalog
  - ✓ Linear Temporal Logic as specification language
An example of mission specification pattern

**Name:** Strict Ordered Patrolling

**Intent:** A robot must patrol a set of locations following a strict sequence ordering. Such locations can be e.g. a set of areas of a building that must be surveyed.

**Template:** The following formula encodes the mission in LTL for two locations and a robot $r$:

$$
\mathcal{G}((\mathcal{F}(l_1 \land \mathcal{F}(l_2))) \land ((\neg l_2) U l_1) \land \mathcal{G}(l_2 \rightarrow X((\neg l_2) U l_1)) \land \mathcal{G}(l_1 \rightarrow X((\neg l_1) U l_2))
$$

where $l_1$ and $l_2$ are expressions that indicate that a robot $r$ is in locations 1 and 2, respectively.

**Variations:** A developer may want to allow traces in which sequences of consecutive $l_1$ ($l_2$) are allowed, that is strict ordering is applied on sequences of non consecutive $l_1$ ($l_2$). In this case, traces in the form $l_1 \rightarrow (\neg l_1 \rightarrow l_1 \rightarrow l_3 \rightarrow l_2)^o$ are admitted, while traces in the form $l_1 \rightarrow (\neg l_1 \rightarrow l_1 \rightarrow l_3 \rightarrow l_2)^o$ are not admitted. This variation can be encoded using the following specification:

$$
\mathcal{G}((\mathcal{F}(l_1 \land \mathcal{F}(l_2))) \land ((\neg l_2) U l_1) \land \mathcal{G}(l_2 \land X(\neg l_2)) \land X((\neg l_2) U l_1)) \land \mathcal{G}(l_1 \land X(\neg l_1)) \rightarrow X((\neg l_1) U l_2))
$$

This specification allows for sequences of consecutive $l_1$ ($l_2$) since the left side of the implication $l_1 \land X(\neg l_1)$ ($l_2 \land X(\neg l_2)$) is only triggered when 1 (2) is exited.

**Examples and Known Uses:** A common usage example of the Strict Ordered Patrolling pattern is a scenario where a robot is performing surveillance in a building during night hours.

Strict Sequence Patrolling and Avoidance often go together. Avoidance patterns are used to force robots to avoid obstacles as they guard a location. Triggers can also be used in combination with the Strict Sequence Patrolling pattern to specify conditions upon which Patrolling should start or stop.

**Relationships:** The Strict Ordered Patrolling pattern is a specialisation of the Ordered Patrolling pattern, in which the robot should keep visiting a set of locations in a given strict order.

**Occurrences:** Smith et. al. [44] proposed a mission specification that forces a robot to not visit a location twice in a row before a final target location is reached. This can be considered as an example of strict patrolling.

**Büchi Automaton:**

![Büchi Automaton](http://178.62.206.217/patterns/)
A simple example

• We want a robot to bring coffee to a user in an office

• The robot should can we systematically support the specification of mission requirements?

• How can we convert a mission requirement into a formal mission specification?
A simple example

- We want a **robot** to **bring** coffee to a **user** in an office

- Visit(coffe, patrizio_office)

- The robot should be able to systematically support the specification...
A simple example

- We want a **robot** to **bring** coffee to a **user** in an office

- **Instantaneous_reaction**(coffee, ask_load_coffee)

- **Instantaneous_reaction**(patrizio_office, ask_unload_coffee)

- The robot should be able to systematically support the specification...
A simple example

• We want a robot to bring coffee to a user in an office

Visit(coffee, patrizio_office) AND
Instantaneous_reaction(coffee, ask_load_coffee) AND
Instantaneous_reaction(patrizio_office, ask_unload_coffee)
A simple example

- We want a **robot** to **bring** coffee to a **user** in an office

Visit(coffee, patrizio_office) AND
Instantaneous_reaction(coffee, ask_load_coffee) AND
Instantaneous_reaction(patrizio_office, ask_unload_coffee)
A simple example

- We want a robot to bring coffee to a user in an office

Ordered_Visit(coffe,patrizio_office) AND
Instantaneous_reaction(coffe,ask_load_coffe) AND
Instantaneous_reaction(patrizio_office,ask_unload_coffe)
PsAlMISt

- https://github.com/claudiomenghi/PsAlMISt
PsAlMiSt
Exploiting the patterns: a DSL for robotic missions
Conclusions

• We discussed specification patterns for robotic mission

• The demo showed how the pattern system can be used in a simple real scenario

Coffe delivery

Fika organizer

Fire robot
Conclusions

• The main idea is to collect recurrent specification problem and "agree" on well known solution of proved validity

• We believe that the proposed work is a base building block in this direction (papers/practical experience)

• The patterns should belong to the community (occurrences in literature are indicated on the website)

• Help us in collecting and maintaining and updating them!
Thanks

• More information about the performed evaluation is presented in the following papers

• Poster: Property Specification Patterns for Robotic Missions
  40th International Conference on Software Engineering (ICSE), Poster Track
  Claudio Menghi, Christos Tsigkanos, Thorsten Berger, Patrizio Pelliccione, Carlo Ghezzi