Robotic Exploration and Mapping with Pharo

Johann Dichtl, Luc Fabresse, and Noury Bouraqadi

IMT Lille – Douai, France

Mai 18, 2017
# Table of Contents

1. Introduction to Localization and Mapping
2. ROS & PhaROS
3. Developing in Pharo
4. Conclusion
Introduction to Localization and Mapping
What is SLAM?

Simultaneous Localization And Mapping (SLAM)

A chicken–and–egg problem:

- to locate ourself, we need a map
- to create/update a map, we need to know our location
Robots in our lab:
Introduction to Localization and Mapping
ROS & PhaROS
Developing in Pharo
Conclusion

What is SLAM?
Our robots
ROS & PhaROS
What is the Robot Operation System (ROS)?

ROS is:
- the de-facto standard middleware used in robotics
- written in C++ and Python
- provides an architecture with a central control structure
- has a modular design, build on nodes
- organizes data flow via topics
- supports distribute systems
Example architecture of a ROS system
Inter–node communication
Two systems with one common master
PhaROS allows us to use Pharo within the ROS framework to:

- create nodes (written in Pharo)
- use topics
- to transmit and receive data
Developing in Pharo
Developing in Pharo

Our setup:
- Pharo 5 (+latest PhaROS)
- Linux (Ubuntu 16.04 LTS)
- ROS kinetic
Developing in Pharo

What is different when working in robotics?

- working with hardware (robots) and simulations
- heterogenous systems
- distributed systems
Developing in Pharo

Limitations in Pharo and how to deal with them

- multithreading support
- linear algebra / matrices
- no 32bit floats
Developing in Pharo

Multithreading support

Pharo supports multiple threads, but only uses a single CPU core.

- split the node into multiple smaller nodes, running each node in its own image
- communication between nodes requires (de-)serialization of objects
Developing in Pharo

Linear algebra & matrices

Only basic support for linear algebra.

- DhbMatrix as basis
- Extensions by us, for example to compute eigen vectors & values
- Geometry utility classes in PhaROS
Developing in Pharo

Testing

- visual debugging
- also test on real robots
Developing in Pharo

Visual debugging
Bugs

Bugs in Pharo

- Athens: random freezes
- Rectangle #center message
Rectangle bug

Rectangle #center message

Code example:

```small
rect := Rectangle origin: 0.2@0.2 corner: 0.8@0.8.
center := rect center.
rect containsPoint: center. "false"
```
What happened?

#center answers with the coordinates floored to the next integer. In our example the result is 0@0, which is indeed outside of the rectangle.
How to fix it?

First try: fix the #center message.
Result: broke some GUI code with the bugfix
Second try: derive new class from Rectangle and fix there

Result: works, but requires some refactoring in our code

Tracker link:
https://pharo.fogbugz.com/f/cases/20015
Conclusion
robotics with Pharo is quite doable
performance can be limiting in pharo
Future work

- automated tests with the full setup
- working with multiple robots at once
Thanks for listening!

Any questions?