EMC 2015
Tooling and Infrastructure

Sjoerd van den Dries

Eindhoven University of Technology
Department of Mechanical Engineering

April 29, 2015

Introducing PICO

- Telepresence Robot from Aldebaran
  - Robot type: Jazz

Introducing PICO

- Telepresence Robot from Aldebaran
  - Robot type: Jazz
- Sensors:
  - Laser Range Finder (LRF)
  - Wheel encoders (odometry)
  - Asus Xtion Depth sensor
  - 170° wide-angle camera
  - Sonar

Introducing PICO

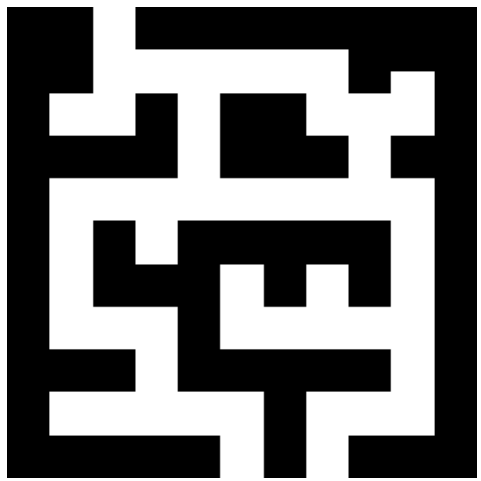
- Telepresence Robot from Aldebaran
  - Robot type: Jazz
- Sensors:
  - Laser Range Finder (LRF)
  - Wheel encoders (odometry)
  - Asus Xtion Depth sensor
  - 170° wide-angle camera
  - Sonar
- Actuators:
  - Holonomic base (omni-wheels)
  - Pan-tilt unit for head

Introducing PICO

- Telepresence Robot from Aldebaran
  - Robot type: Jazz
- Sensors:
  - Laser Range Finder (LRF)
  - Wheel encoders (odometry)
  - Asus Xtion Depth sensor
  - 170° wide-angle camera
  - Sonar
- Actuators:
  - Holonomic base (omni-wheels)
  - Pan-tilt unit for head
- Computer:
  - Intel I7
  - Running Ubuntu 14.04

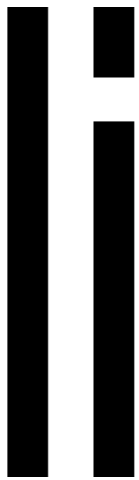
The Assignment

- Let PICO navigate through a maze and find and go to the exit.
- You have to:
  - try to be as fast as possible
  - deal with dynamics in the environment (a moving door)
- You can use:
  - The Laser Range Finder to detect walls and doors
  - The encoder data from the wheels
- Competition day: June 17



Intermediate Assignment

- Corridor Competition: Let PICO drive through a corridor and go through the side exit.
- You have to:
  - try to be as fast as possible
- You can use:
  - The Laser Range Finder to detect walls
  - The encoder data from the wheels
- Competition day: May 13



Ubuntu

- Linux-based operating system
- Use version **14.04** (not 14.10!)
- 32- and 64-bit (64-bit recommended)
- Easy dual boot installation with *e.g.*, Windows
- Download: `www.ubuntu.com`
  - Any problems? → Check the wiki.
  - No info? → Ask the ICT Servicedesk or contact us.

ROS?

- Robot Operating System
- Open-source meta-operating system for robots

ROS?

- Robot Operating System
- Open-source meta-operating system for robots
- Won't be using it!

ROS?

- Robot Operating System
- Open-source meta-operating system for robots

- Won't be using it!
- Instead, we will provide our own 'software layer'
  - It is simpler to understand, and 'cleaner' to use

ROS?

- Robot Operating System
- Open-source meta-operating system for robots

- Won't be using it!
- Instead, we will provide our own 'software layer'
  - It is simpler to understand, and 'cleaner' to use
- However, you are still allowed to use ROS!

C++

- We will use C++ as programming language
- C++ is object-oriented C
  - "C with Classes"
  - Encapsulate data and functionality within objects
- Is a powerful but complex programming language.
- However, we provide a software framework to get you started

Creating code: Qt Creator

- Integrated Development Environment
  - Advanced code editor
- Many advantages over 'simple editors':
  - Syntax highlighting
  - Code completion
  - Visual compiler feedback
  - Static code checking
  - Refactoring tools
  - Parenthesis matching
  - ...

Git Version Control

Git Version Control

- Version Control System:
  - *'Manages files and directories, and the changes made to them, over time'*
- Use to store and maintain your code on the server
  - (Like Dropbox)

Git Version Control

- Version Control System:
  - *'Manages files and directories, and the changes made to them, over time'*
- Use to store and maintain your code on the server
  - (Like Dropbox)
- Maintains version history
- Is distributed
  - You always have the full history on your pc
  - You can always go back to a version, show differences, even when off-line

Git Version Control

- Version Control System:
  - *'Manages files and directories, and the changes made to them, over time'*
- Use to store and maintain your code on the server
  - (Like Dropbox)
- Maintains version history
- Is distributed
  - You always have the full history on your pc
  - You can always go back to a version, show differences, even when off-line
- More info later

PICO Simulator

- You will have to work with the real robot, but we only have one. Therefore:

PICO Simulator

- You will have to work with the real robot, but we only have one. Therefore:
- PICO Simulator:
  - Simulates:
    - Sensors (Laser, odometry)
    - Actuators (Base)
    - Environment (maze)

PICO Simulator

- You will have to work with the real robot, but we only have one. Therefore:
- PICO Simulator:
  - Simulates:
    - Sensors (Laser, odometry)
    - Actuators (Base)
    - Environment (maze)
- Can easily create test environments using height maps

PICO Simulator

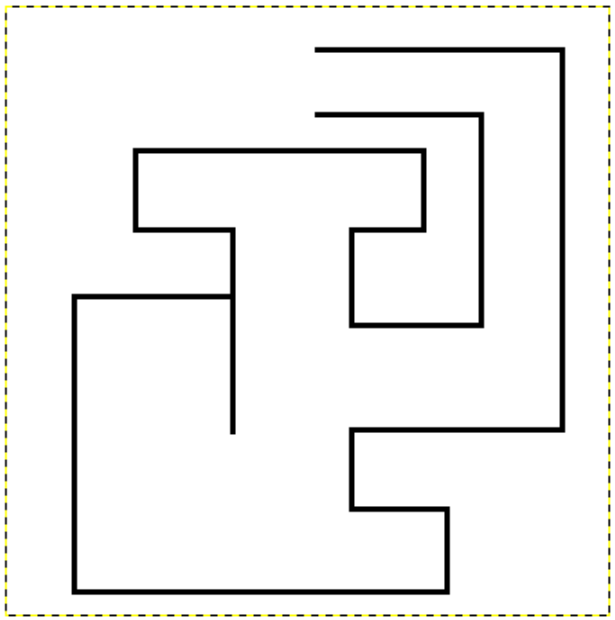
- You will have to work with the real robot, but we only have one. Therefore:
- PICO Simulator:
  - Simulates:
    - Sensors (Laser, odometry)
    - Actuators (Base)
    - Environment (maze)
- Can easily create test environments using height maps
- Integrates well with our provided software
  - If you your software runs on the simulator, it runs on the robot

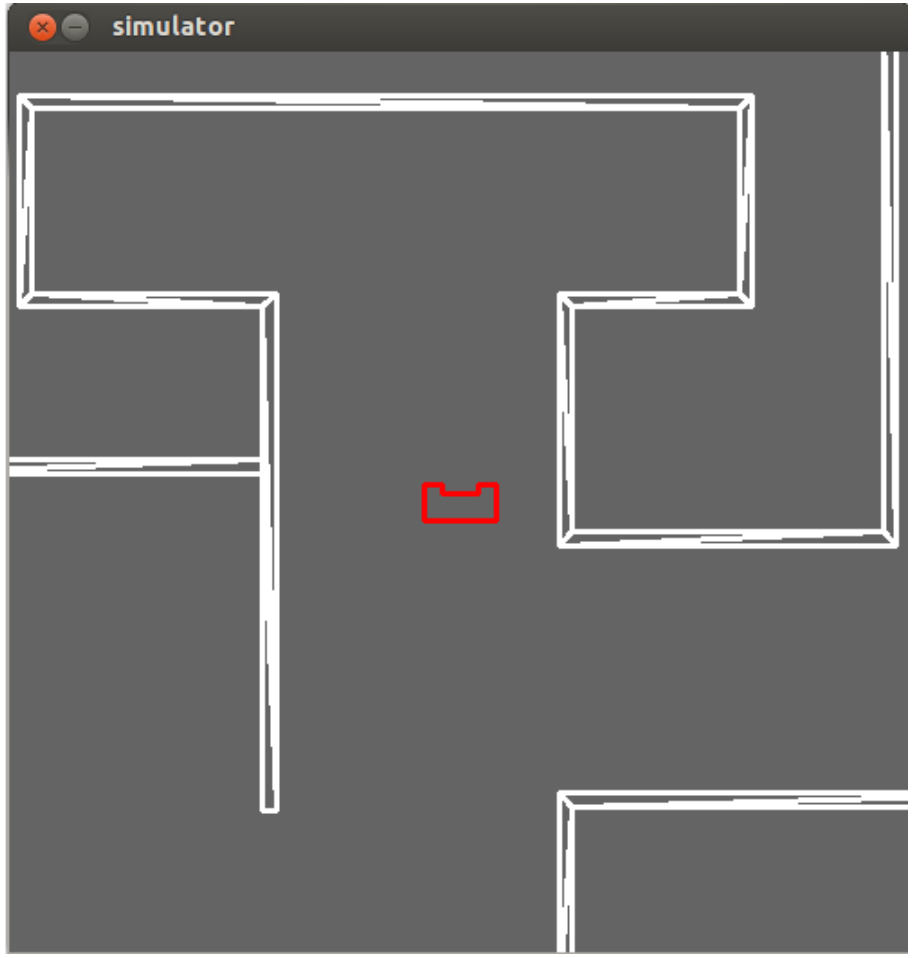
PICO Simulator

- You will have to work with the real robot, but we only have one. Therefore:
- PICO Simulator:
  - Simulates:
    - Sensors (Laser, odometry)
    - Actuators (Base)
    - Environment (maze)
- Can easily create test environments using height maps
- Integrates well with our provided software
  - If you your software runs on the simulator, it runs on the robot
  - This does not guarantee that it will also work...

You still need to test on the real system!

PICO Simulator





Wiki

- EMC Wiki:
  - http://cstwiki.wtb.tue.nl/index.php?title=Embedded_Motion_Control
  - Info on practical assignment, installation, getting started
  - Log-in: student account

Wiki

- EMC Wiki:
  - http://cstwiki.wtb.tue.nl/index.php?title=Embedded_Motion_Control
  - Info on practical assignment, installation, getting started
  - Log-in: student account
- Group pages on EMC Wiki:
  - Each group its own page
  - Update at least weekly

Wiki

- EMC Wiki:
  - http://cstwiki.wtb.tue.nl/index.php?title=Embedded_Motion_Control
  - Info on practical assignment, installation, getting started
  - Log-in: student account
- Group pages on EMC Wiki:
  - Each group its own page
  - Update at least weekly
- Overall use:
  - Everyone can edit
  - If you see a mistake: correct it
  - If you had a problem and know how to fix it: add it

Recap

- Assignment: solve maze with PICO robot

Recap

- Assignment: solve maze with PICO robot
- OS: Ubuntu 14.04
- Programming language: C++
- Code editor: Qt Creator
- Version control: git
- Simulation: PICO simulator
- Documentation: Wiki

Getting Started

- Check the wiki:
  - http://cstwiki.wtb.tue.nl/index.php?title=Embedded_Motion_Control
- Follow the tutorials on the wiki:
  - Ubuntu, C++, Qt Creator
  - Coming soon: simulator, software framework, git

Tutor name will be sent to you

It is *your* responsibility to get in touch with your tutor