

# Machine Learning for Probabilistic Robotics with Webots

Joan Gerard<sup>1</sup>

Promotor: Prof. Gianluca Bontempi <sup>1</sup>

<sup>1</sup>Université Libre de Bruxelles

June 7, 2019

# Table of Contents

[Probabilistic Robotics](#)

[Webots](#)

[Project Objectives](#)

[Preliminary Results](#)

# Table of Contents

Probabilistic Robotics

Webots

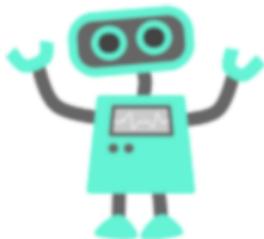
Project Objectives

Preliminary Results

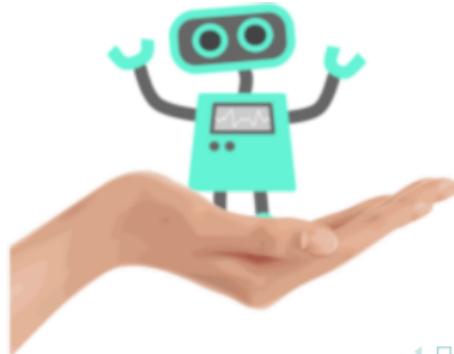
# Where am I?

- ▶ Robot knows where it is
- ▶ Robot is kidnapped
- ▶ Robot does not know where it is

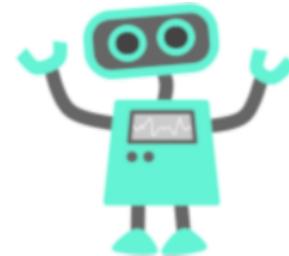
I know where I am!



I was kidnapped.  
HELP!

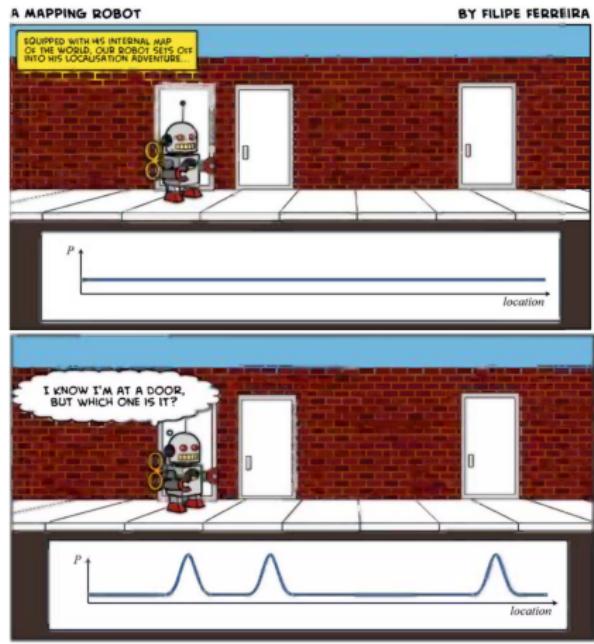


Now I am lost  
How to recover?

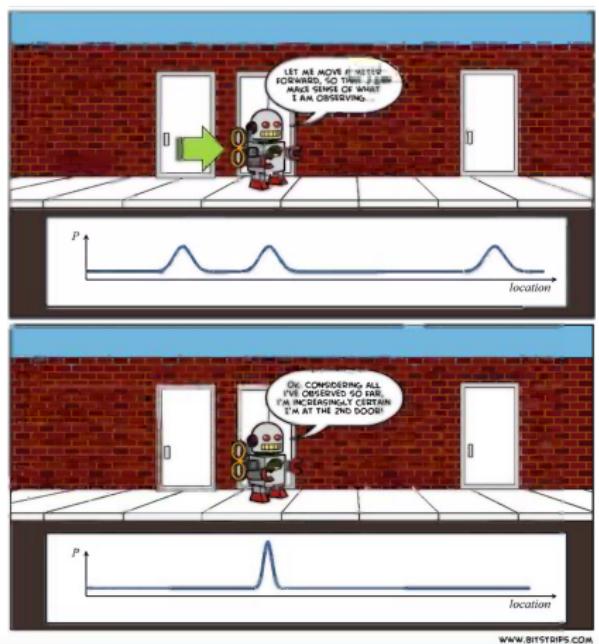


# Let's figure it out using Bayes Filter!

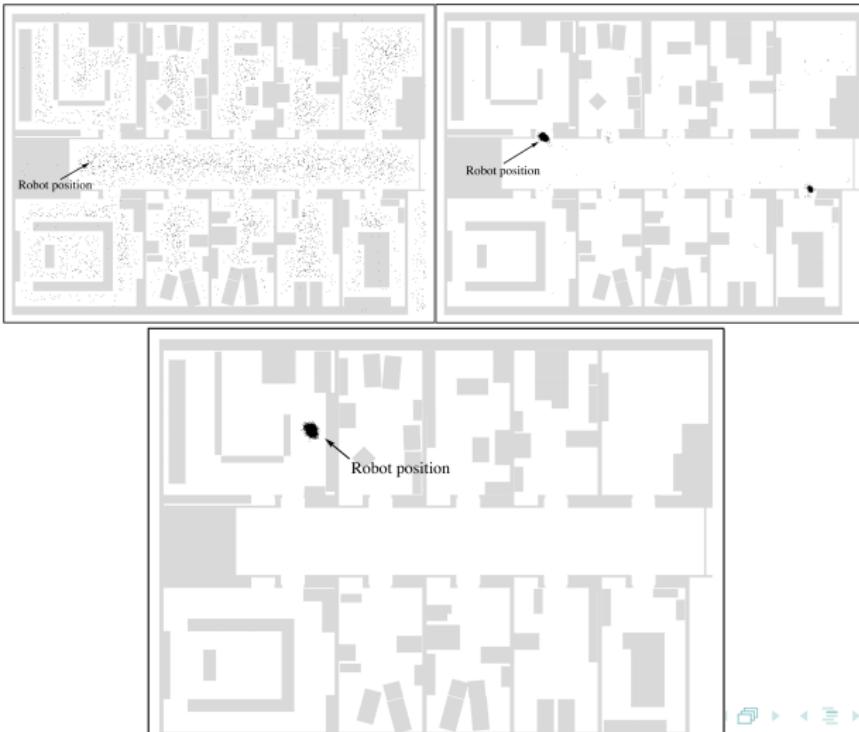
## ► Prediction



## ► Error Correction



# What about Particles Filter?



# Particles Filter Algorithm

- ▶ Train/test a prediction model: robot state → distance sensor data
- ▶ Compare it with the true sensor data and determine a weight for each particle
- ▶ More close to the real sensor data more weight will have

---

```
input : particles, controlAction, sensorData
output: nextParticles
1 nextParticles ← ∅
2 foreach particle ∈ particles do
3     particle.state ← predictState(particle.state, controlAction)
4     particle.weight ← calculateWeight(particle.state, sensorData)
5 end
6 for m = 1 to m = |particles| do
7     newParticle ← draw i from particles with probability ∝ particles[i].weight
8     nextParticles.add(newParticle)
9 end
```

# Table of Contents

Probabilistic Robotics

Webots

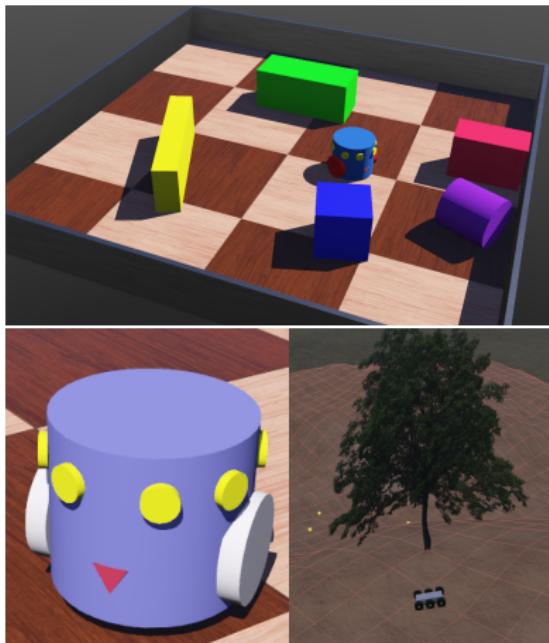
Project Objectives

Preliminary Results

# What tool can we use for simulating that problem?

Webots!

- ▶ Open Source
- ▶ Python, C++, Java, etc.
- ▶ 44 robot models
- ▶ Robot model creation/customization
- ▶ Robust documentation



# Table of Contents

Probabilistic Robotics

Webots

Project Objectives

Preliminary Results

# Main Objective

- ▶ The objective will be to exploit the simulation benefits of Webots to introduce Machine Learning techniques together with non-parametric filters (such as Particles Filter) for robot positioning estimation, independently from the kind of robot used for in-door environments.

# How do we plan to do it?

- ▶ Using Webots
  - ▶ Create a custom robot
  - ▶ Randomize control actions while moving avoiding obstacles
  - ▶ Use a realistic in-door environment
- ▶ Using Machine Learning and Particles Filters
  - ▶ Positioning estimation
  - ▶ Error correction
- ▶ Test the developed technique with other robots

# Table of Contents

Probabilistic Robotics

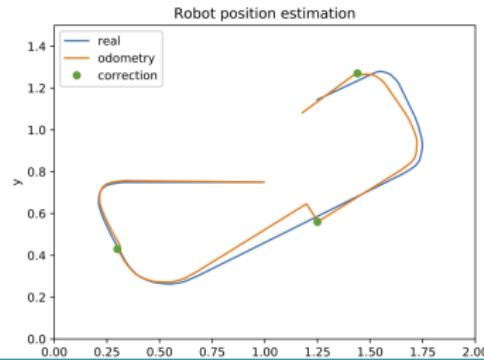
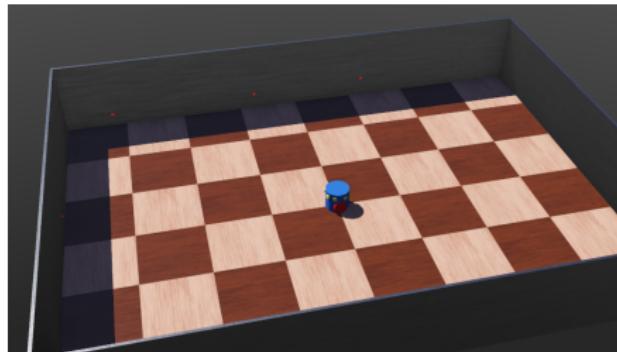
Webots

Project Objectives

Preliminary Results

# What have we done so far?

- ▶ Robot creation
- ▶ Dummy environment creation
- ▶ Collect distance sensor data + real position coordinates
- ▶ Bayes Filter algorithm implementation
  - ▶ *Position prediction:* Odometry techniques
  - ▶ *Error correction:* Training a Random Forest model using collected data



# Are the results good?

- ▶ Yes! but there are improvements to be done:
  - ▶ Collect more data (+ Simulations)
  - ▶ Randomize robot moves
  - ▶ Train/test with other Machine Learning models
  - ▶ Improve performance of the error correction algorithm

# Any question?

