Disclaimer

These slides are intended as presentation aids for the lecture. They contain information that would otherwise be to difficult or time-consuming to reproduce on the board. But they are incomplete, not self-explanatory, and are not always used in the order they appear in this presentation. As a result, these slides should not be used as a script for this course. I recommend you take notes during class, maybe on the slides themselves. It has been shown that taking notes improves learning success.

Reading for this set of slides

- Planning Algorithms (Steve LaValle)
  - 6 Combinatorial Motion Planning (6.1 – 6.3)
  - 8 Feedback Motion Planning (8.1, 8.2)
- Please refer to the slides for potential fields and vehicle kinematics

Please note that this set of slides is intended as support for the lecture, not as a stand-alone script. If you want to study for this course, please use these slides in conjunction with the indicated chapters in the text books. The textbooks are available online or in the TUB library (many copies that can be checked out for the entire semester. There are also some aspects of the lectures that will not be covered in the text books but can still be part of the homework or exam. For those It is important that you attend class or ask somebody about what was covered in class.

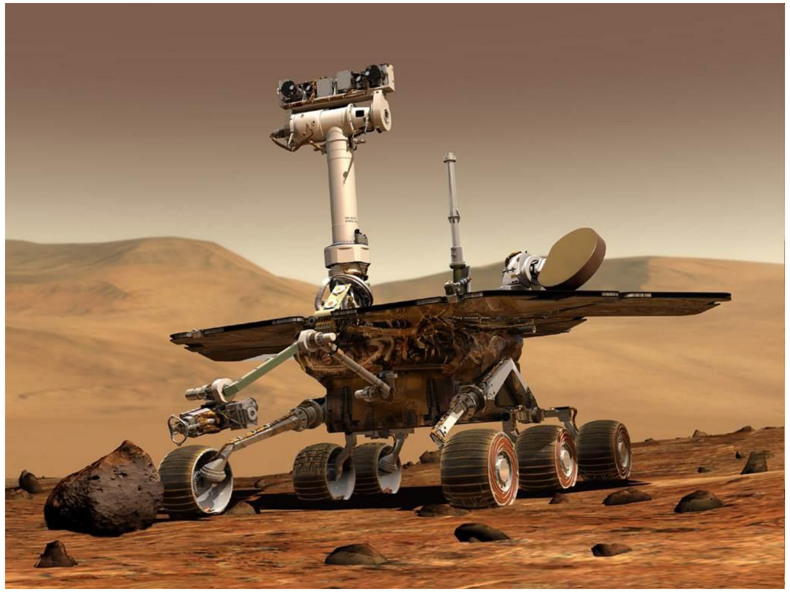
Robotics

Mobile Robots

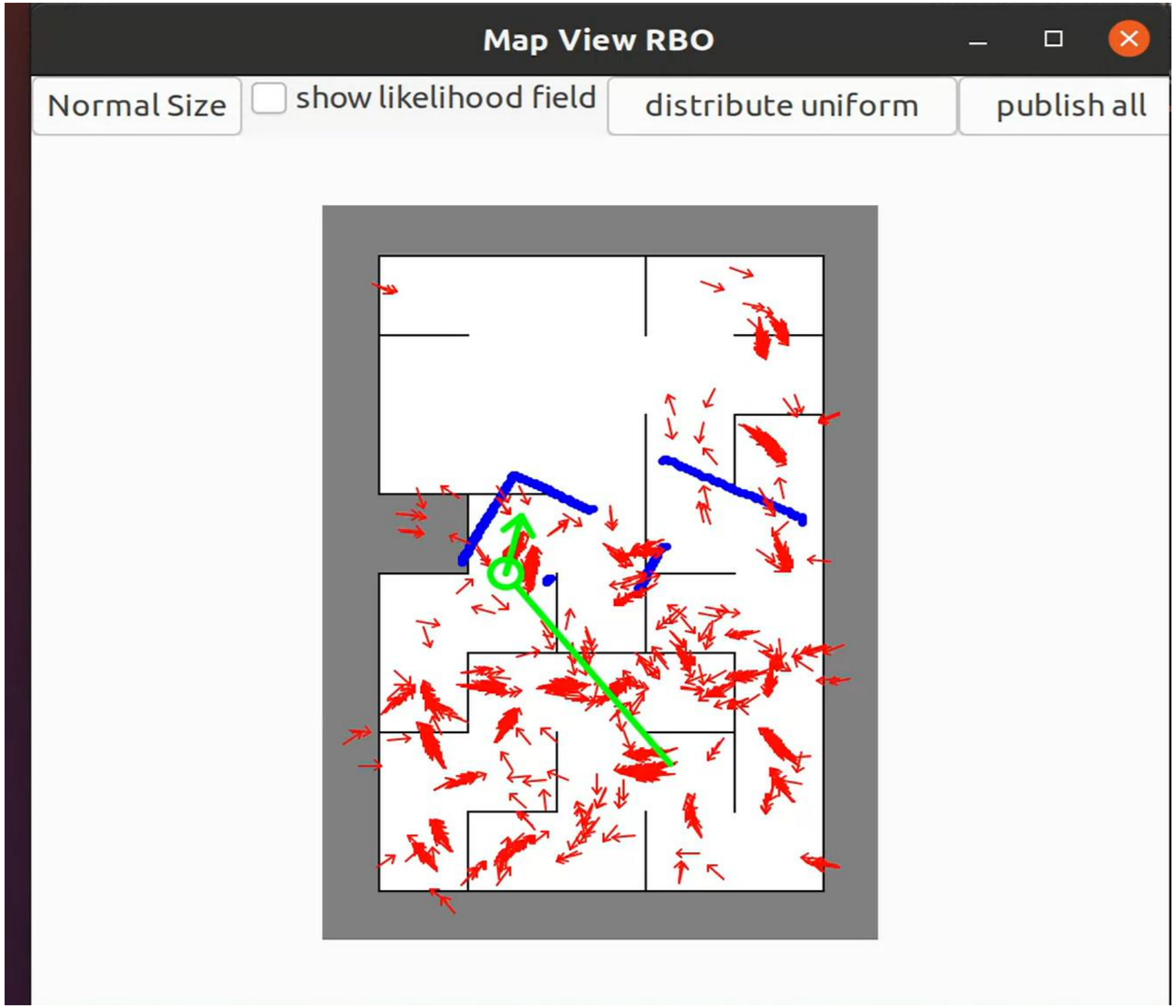
TU Berlin

Oliver Brock

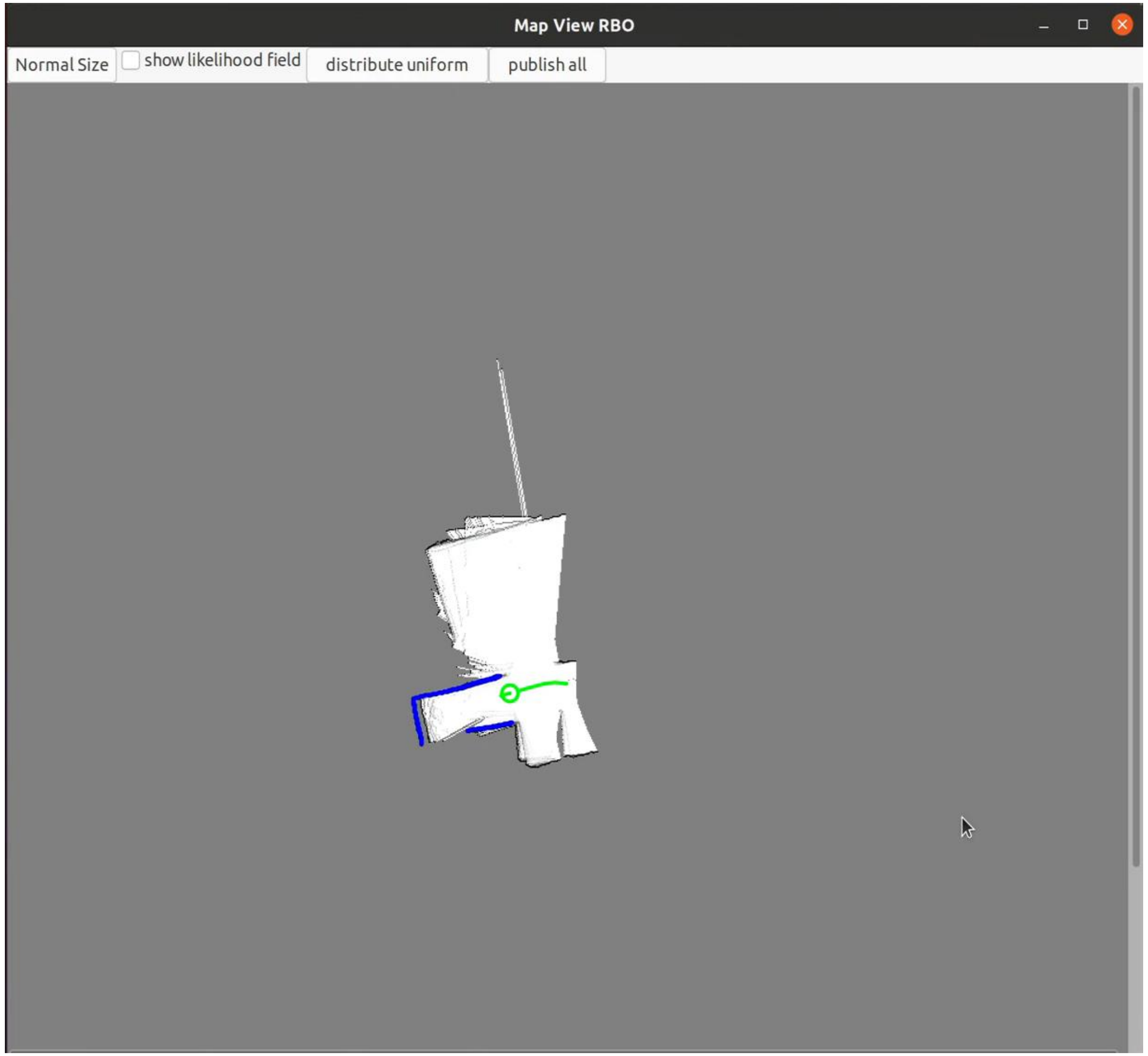
What is the Difference?



Where am I? And what is “where”?



Where am I? And what is “where”?



SLAM

Nature-Made Mobility



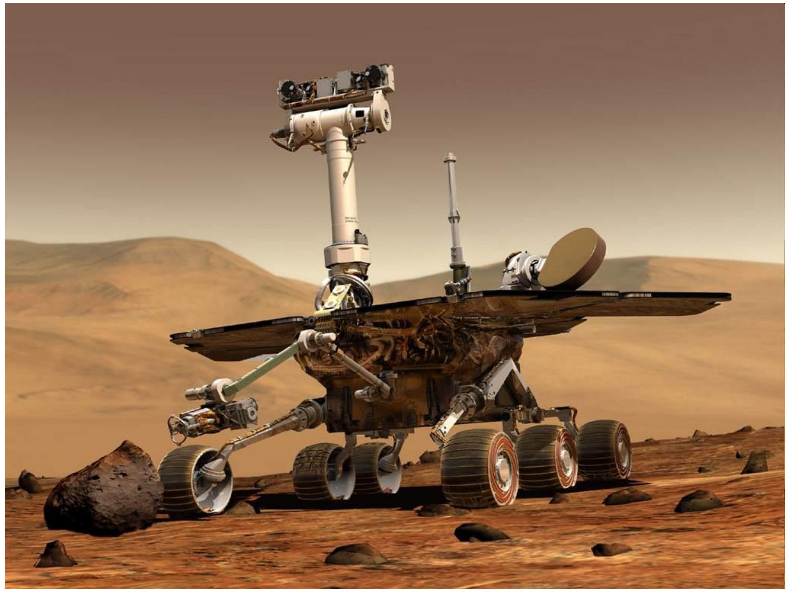


Human-Made Mobility


Boston Dynamics


MASSACHVSETTS INSTITVTE OF TECHNOLOGY

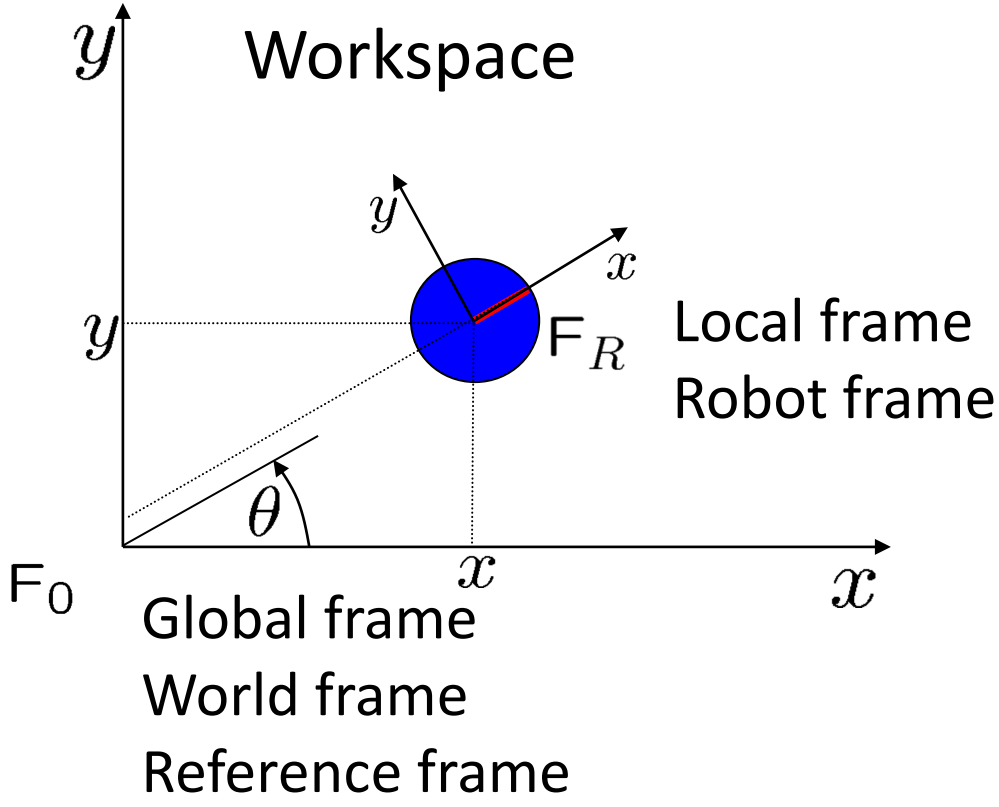
What is the Difference?



LOCALIZATION

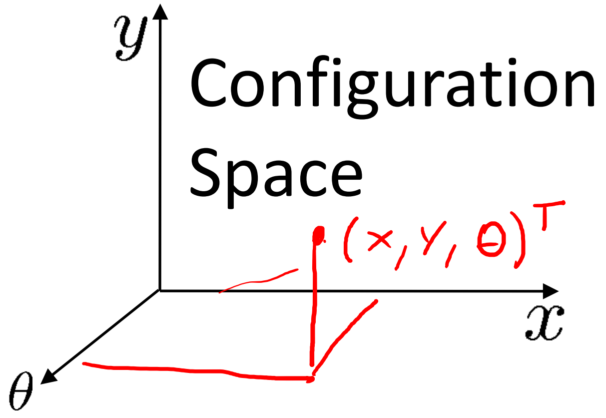
SLAM

Representation

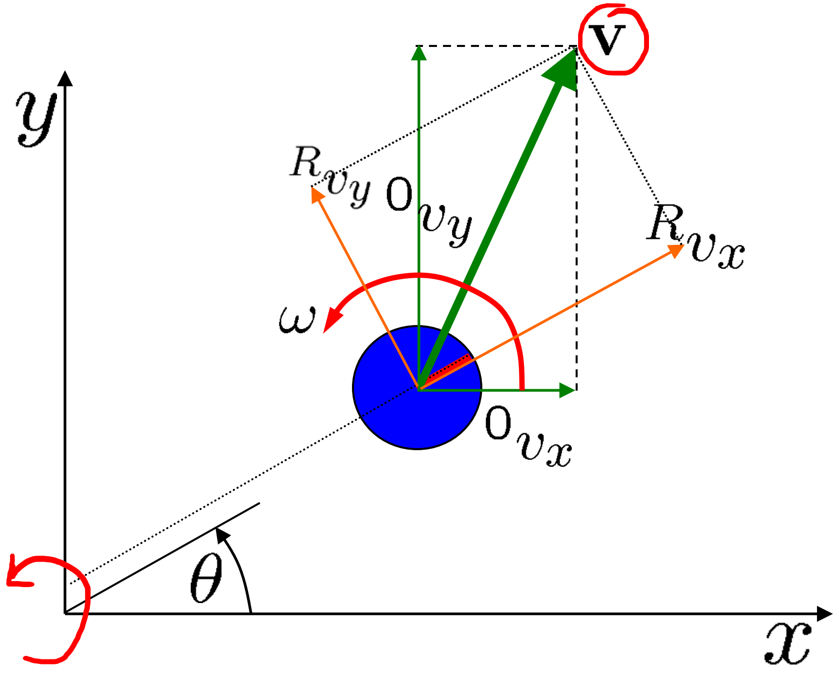


$$q = (\overbrace{x, y}^{\text{position}}, \theta)^T$$

orientation
or heading



Representation cont.



3 DOF

$$q = (x, y, \theta)^T$$

$$\dot{q} = (\dot{x}, \dot{y}, \dot{\theta})^T$$

$$\dot{q} = (v_x, v_y, \omega)$$

$$\dot{q} = (\mathbf{v}, \omega)$$

Encoders for Odometry

Odometer measures how far we go…

We can use odometry to estimate the robot’s configuration based on the motor commands we sent.

Encoders!

Dead Reckoning

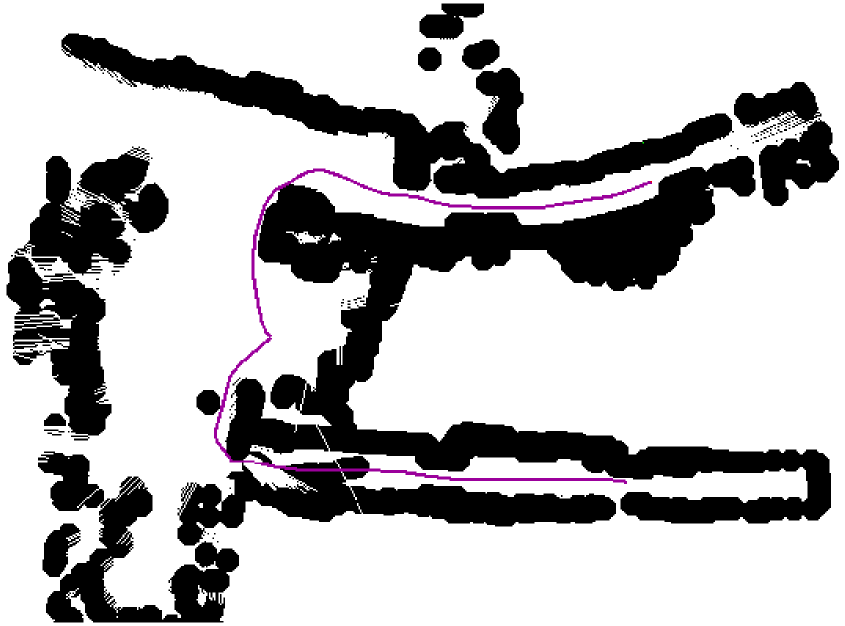
DEDUCTIVE

Estimating the Postition

- How can we estimate the behavior of the robot based on the command we send?
- Time: $\hat{s} = t \cdot v_{\text{desired}}$
- Error: $s - \hat{s} = t \cdot \underbrace{(v_{\text{actual}} - v_{\text{desired}})}_{e}$
- Error accumulates with time!
- Error can be very large!

Error Sources

- Controller
- Mapping to motor command
- Performance of motor command
  - slippage!
  - actuation limits



Motion Constraints

2 DOF OF ACTUATION:
ACCELERATOR, STEERING

$(x, y, \theta)^T$

1 DOF

nonholonomic

Nonholonomic equality constraint:

$$-\sin\theta\,\dot{x} + \cos\theta\,\dot{y} = 0$$

$$\overset{\theta=0^\circ}{\Rightarrow} \quad 0\,\dot{x} + 1\,\dot{y} = 0$$

$$\overset{\theta=45^\circ}{\Rightarrow} \quad -\frac{1}{\sqrt{2}}\,\dot{x} + \frac{1}{\sqrt{2}}\,\dot{y} = 0$$

$$\overset{\theta=90^\circ}{\Rightarrow} \quad -1\,\dot{x} + 0\,\dot{y} = 0$$

holonomic