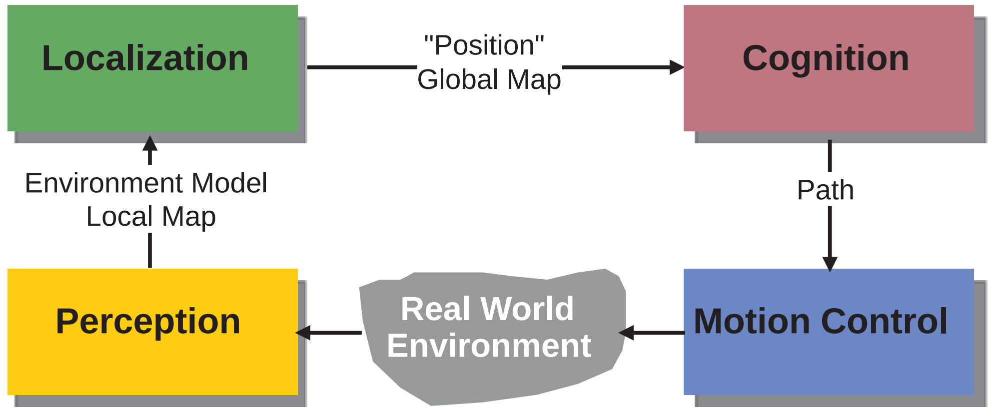
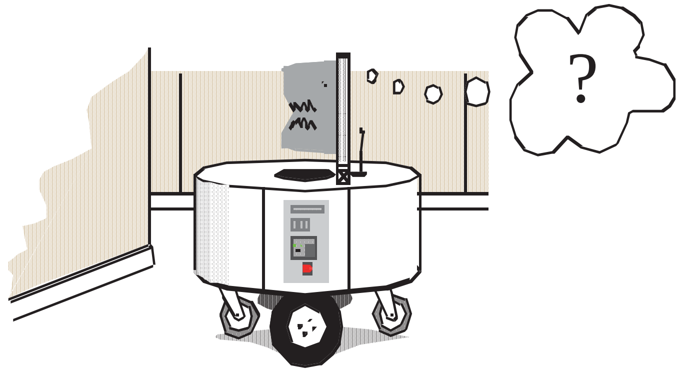
Autonomous Mobile Robots



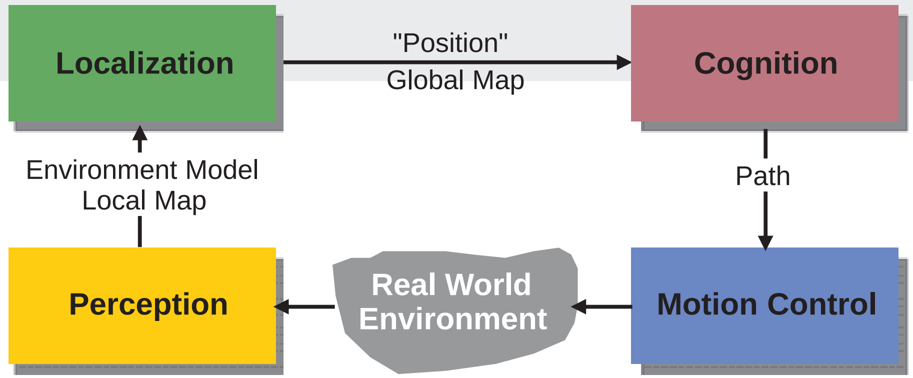
Localization



ETH Zürich

Autonomous Systems Lab

Where are we?

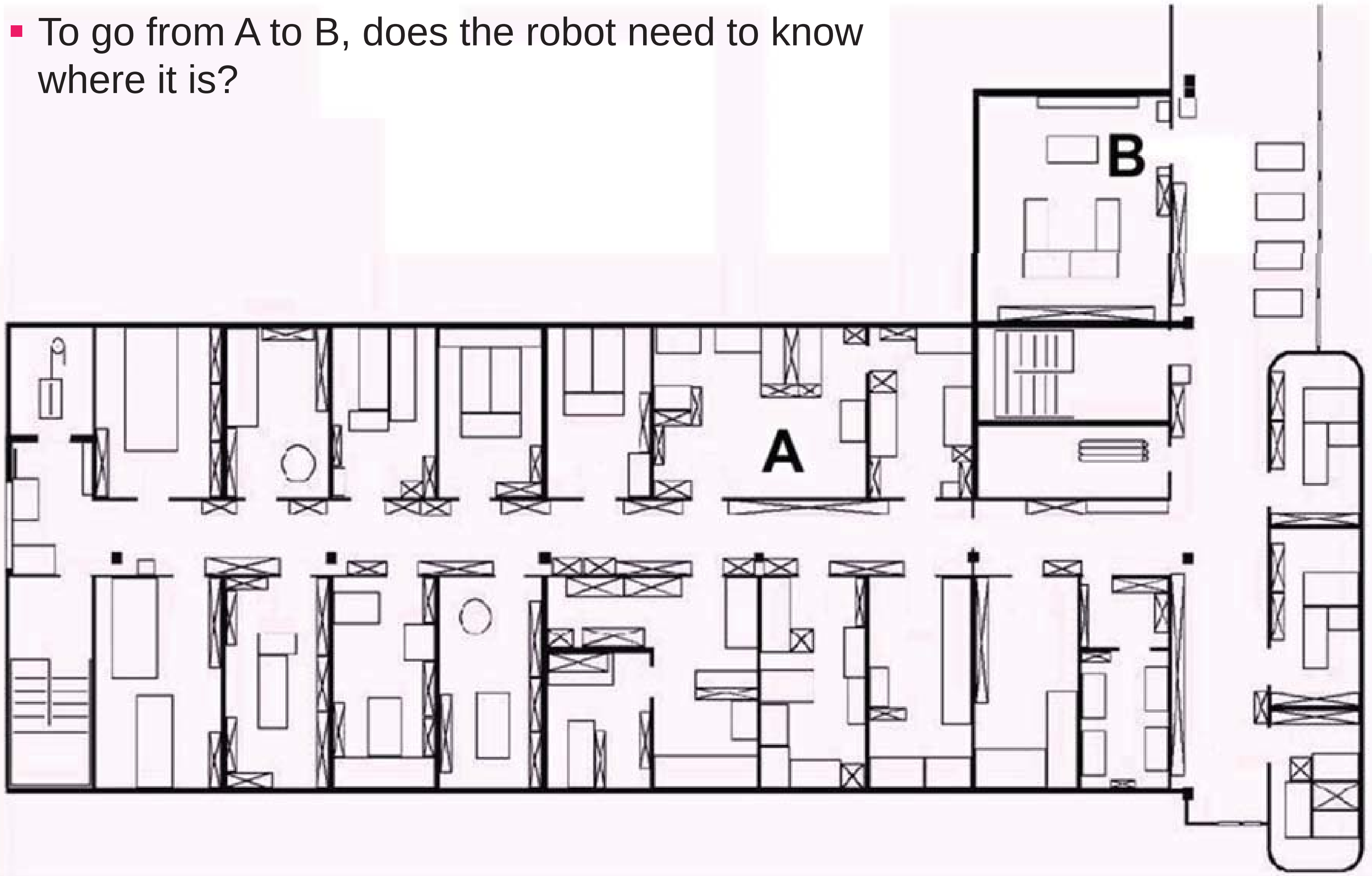


- Overview of what we have seen so far
  - Mobile robot kinematics and motion control
  - Proprioceptive and exteroceptive sensors; sensor uncertainty
  - Feature extraction from laser and camera images
    - Lines (from laser and camera)
    - Planes (just as an example)
    - Points (Ex. corners: Harris, SIFT)

- What will we see next?
  - Robot localization (today and next two lecture)
  - Autonomous map building (SLAM)
  - Path planning

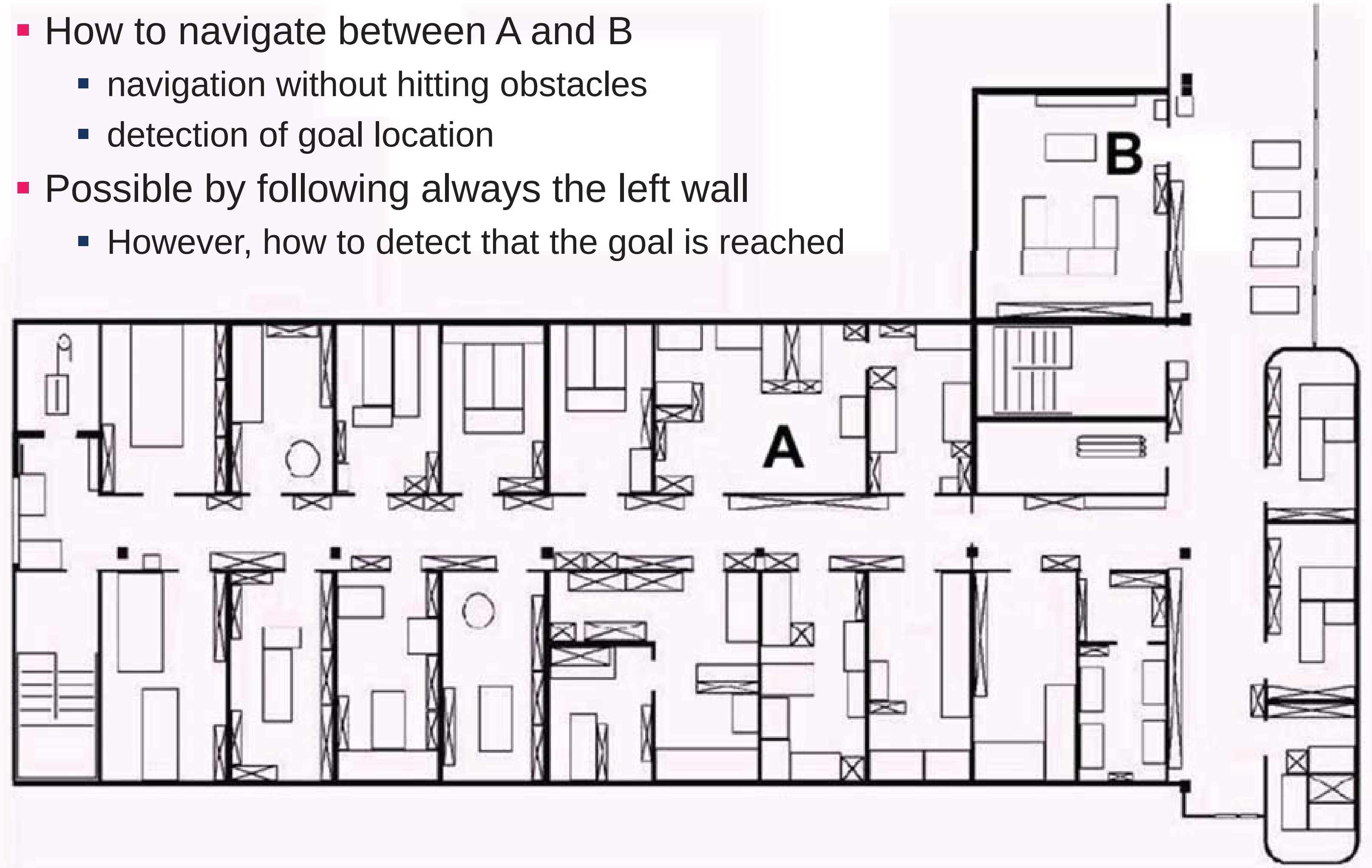
Do we need to localize or not?

- To go from A to B, does the robot need to know where it is?



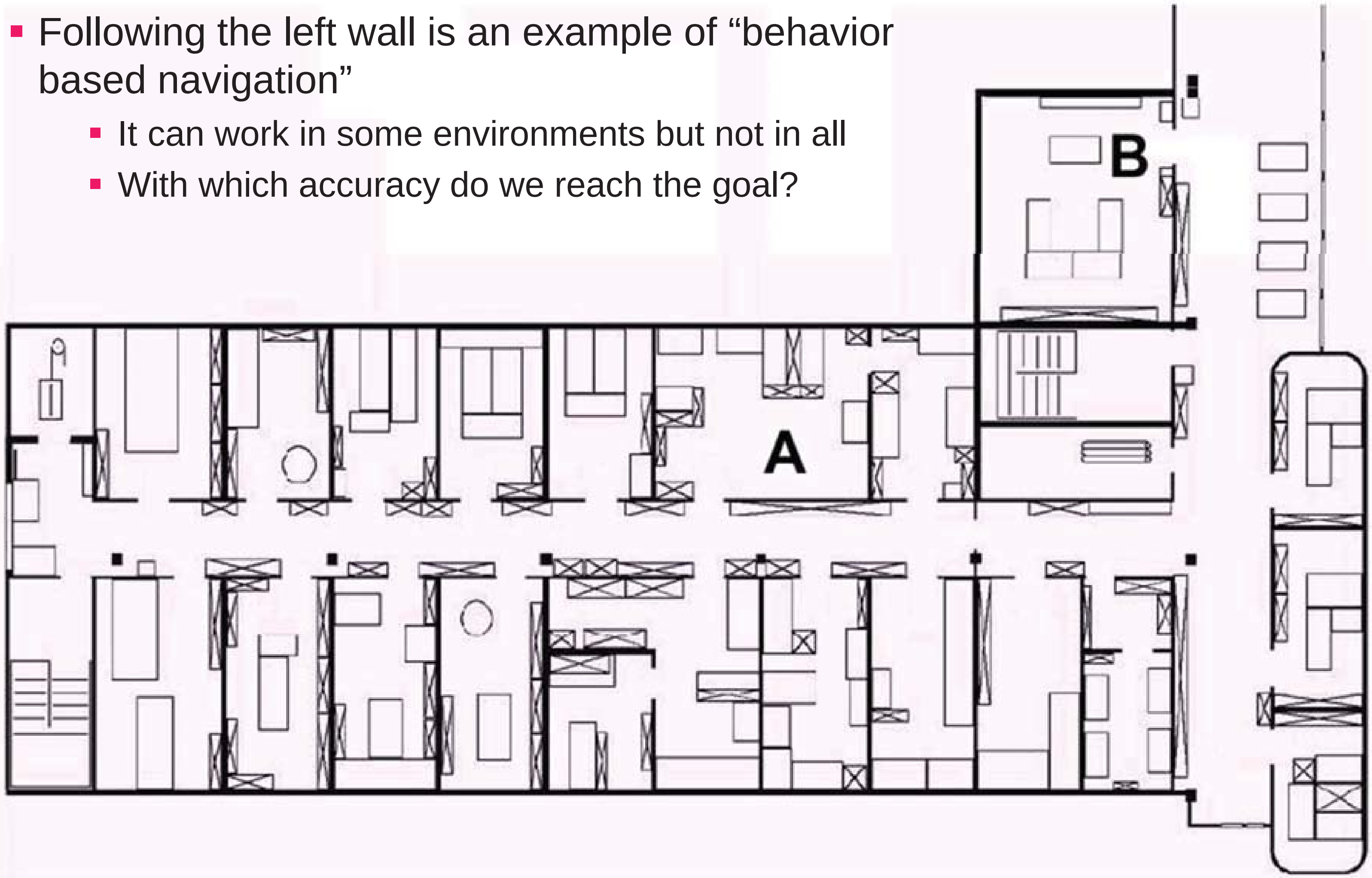
4 Do we need to localize or not?

- How to navigate between A and B
  - navigation without hitting obstacles
  - detection of goal location
- Possible by following always the left wall
  - However, how to detect that the goal is reached



5 Do we need to localize or not?

- Following the left wall is an example of "behavior based navigation"
  - It can work in some environments but not in all
  - With which accuracy do we reach the goal?



6 Do we need to localize or not?

- In contrast to behavior based navigation, ***map based navigation*** relies on a map
- Assuming that the map is known, at every time step the robot has to localize itself in the map.
  - How?
  - If we know the start position, we can use wheel odometry or dead reckoning.
    - Is this enough?
    - What else can we use?
- But how do we represent the map for the robot?
- And how do we represent the position of the robot in the map?

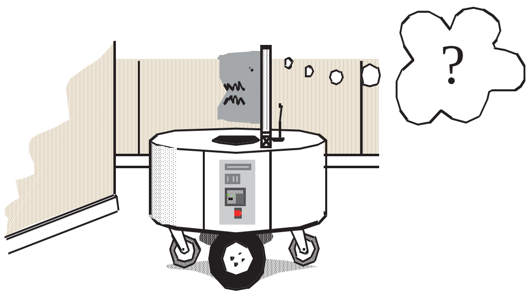
Today's lecture overview

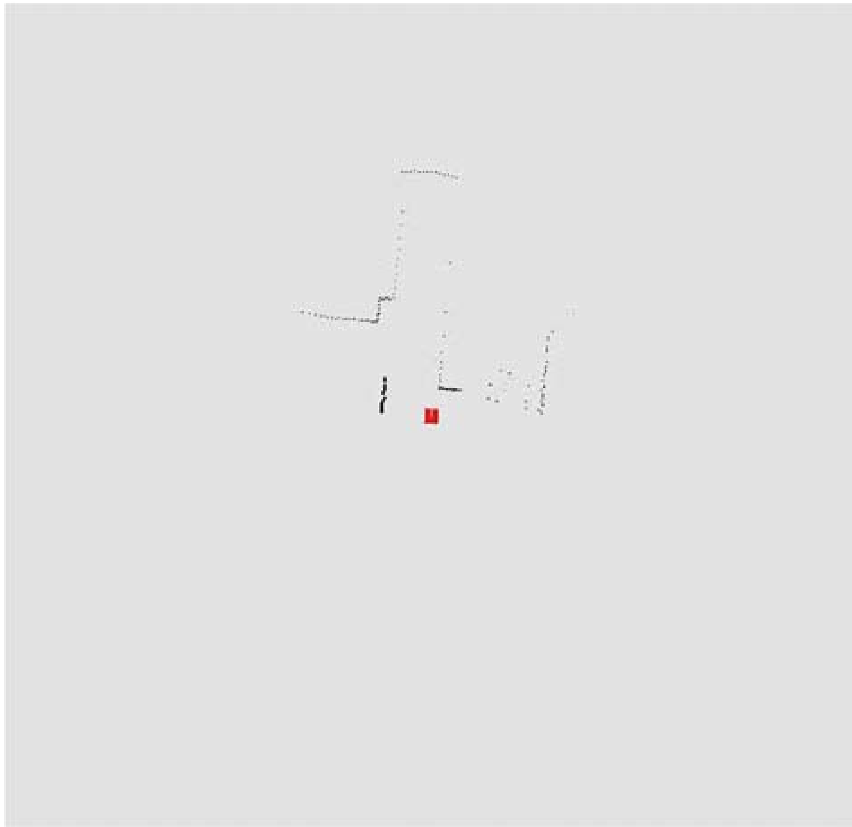
- Types of localization and examples of localization systems
- Noise: odometric position estimation
- Belief representation: how to represent the robot position
- Map representation: continuous, discrete, topological
- Introduction to probabilistic map based localization

8 Localization

- Global localization
  - The robot is not told its initial position
  - Its position must be estimated from scratch

- Position Tracking
  - A robot knows its initial position and “only” has to accommodate small errors in its odometry as it moves



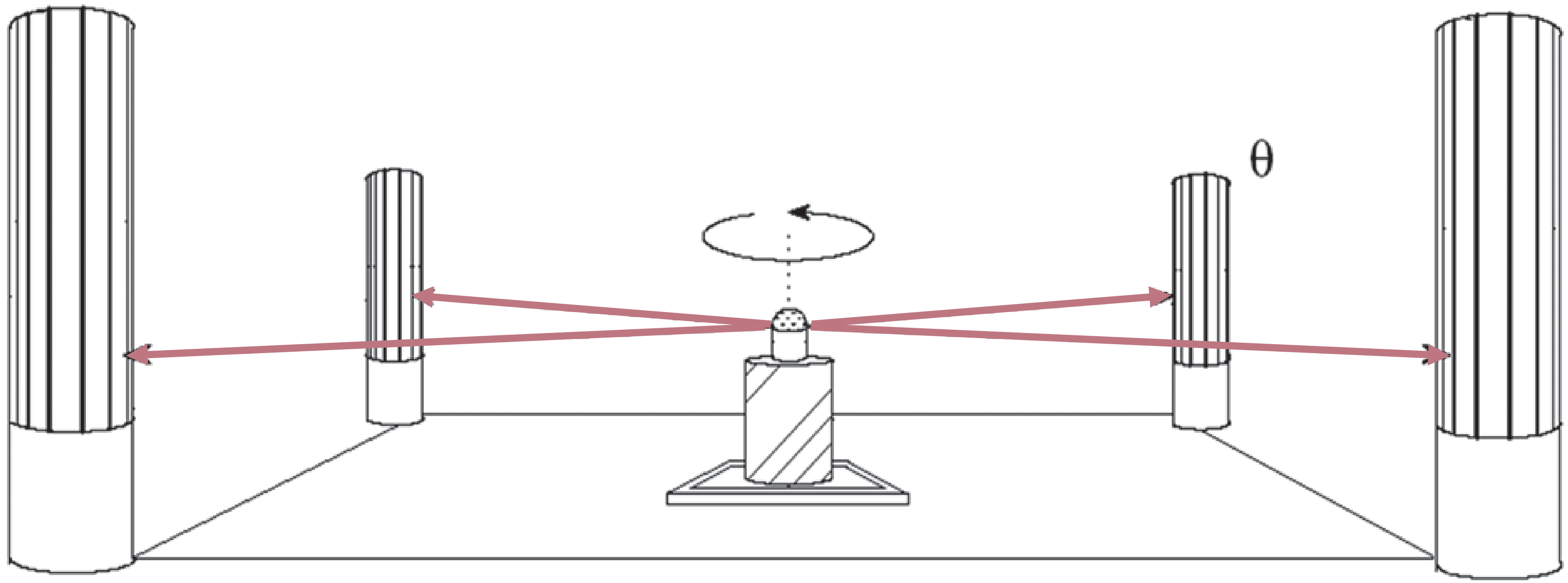


How to localize?

- Localization based on external sensors, beacons or landmarks
- Odometry
- Map Based Localization
  - without external sensors or artificial landmarks, just use robot onboard sensors
  - Example: Probabilistic Map Based Localization

10 Beacon Based Localization Systems: Triangulation

- Ex 1: Poles with highly reflective surface and a laser for detecting them
- Ex 2: Coloured beacons and an omnidirectional camera for detecting them (example: robocup or autonomous robots in tennis fields)



11 Beacon Based Localization Systems: KIVA Systems

- KIVA Systems, Boston (MA)

http://www.youtube.com/watch?v=lWsMdN7HMuA

12 The SmartTer Platform – Map and GPS Based Localization

- Three navigation SICK laser scanners
  - Obstacle avoidance and local navigation
- Two rotating laser scanners (3D SICK)
  - 3D mapping of the environment
  - Scene interpretation
- Omnidirectional camera
  - Texture information for the 3D terrain maps
  - Scene interpretation
- Monocular camera
  - Scene interpretation

Motion Estimation / Localization

- Differential GPS system *(Omnistar 8300HP)*
- Inertial measurement unit *(Crossbow NAV420)*
- **Optical Gyro**
- Odometry (wheel speed, steering angel)
  - Motion estimation
  - Localization

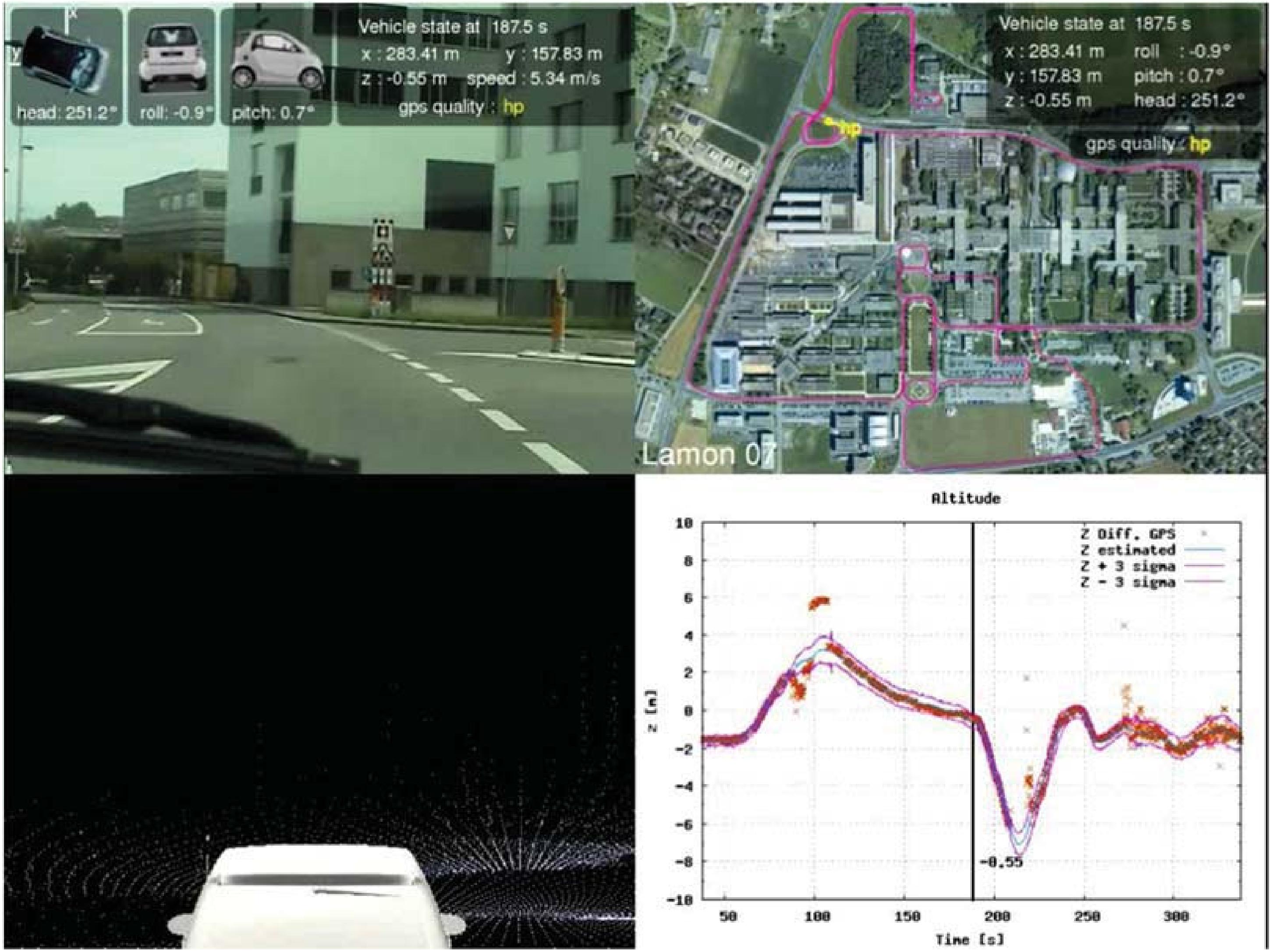
Internal car state sensors

- Vehicle state flags (engine, door, etc.)
- Engine data, gas pedal value

Camera for life video streaming

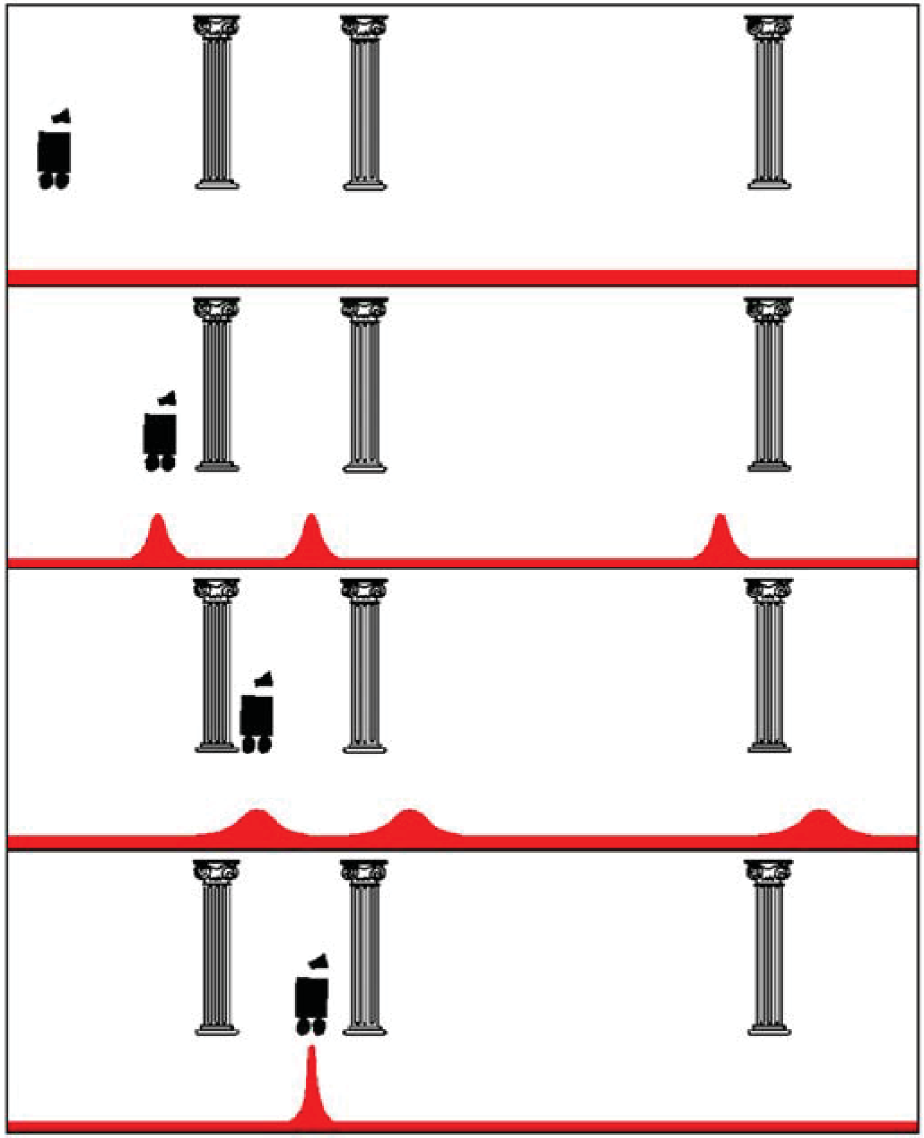
- Transmission range up to 2 km

13 Autonomous Navigation and 3D Mapping



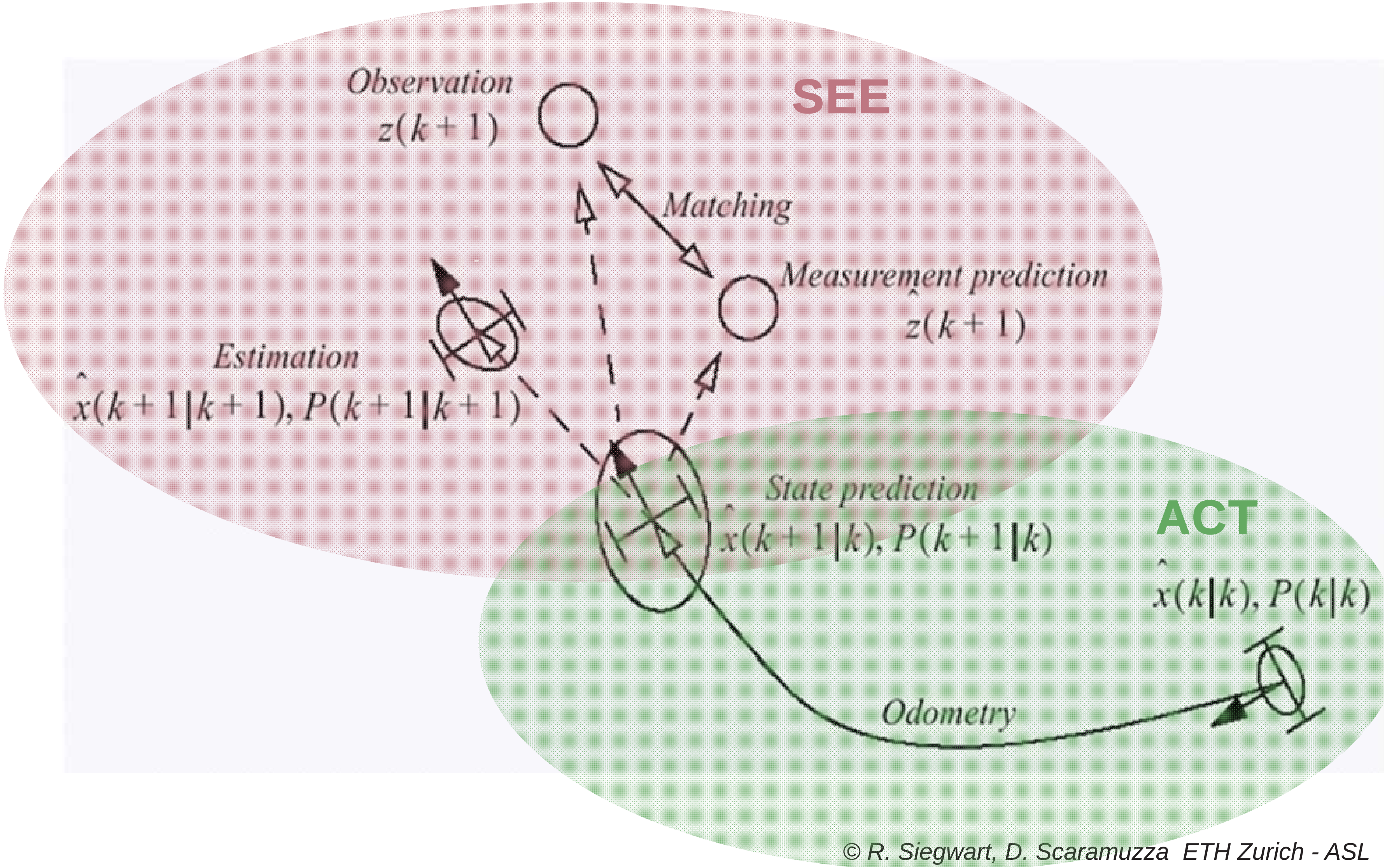
Localization Cycle

- Improving belief state
  by moving (SEE and ACT)

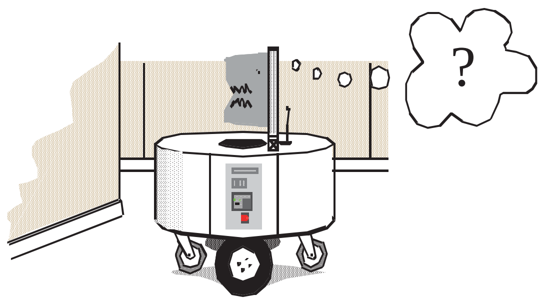


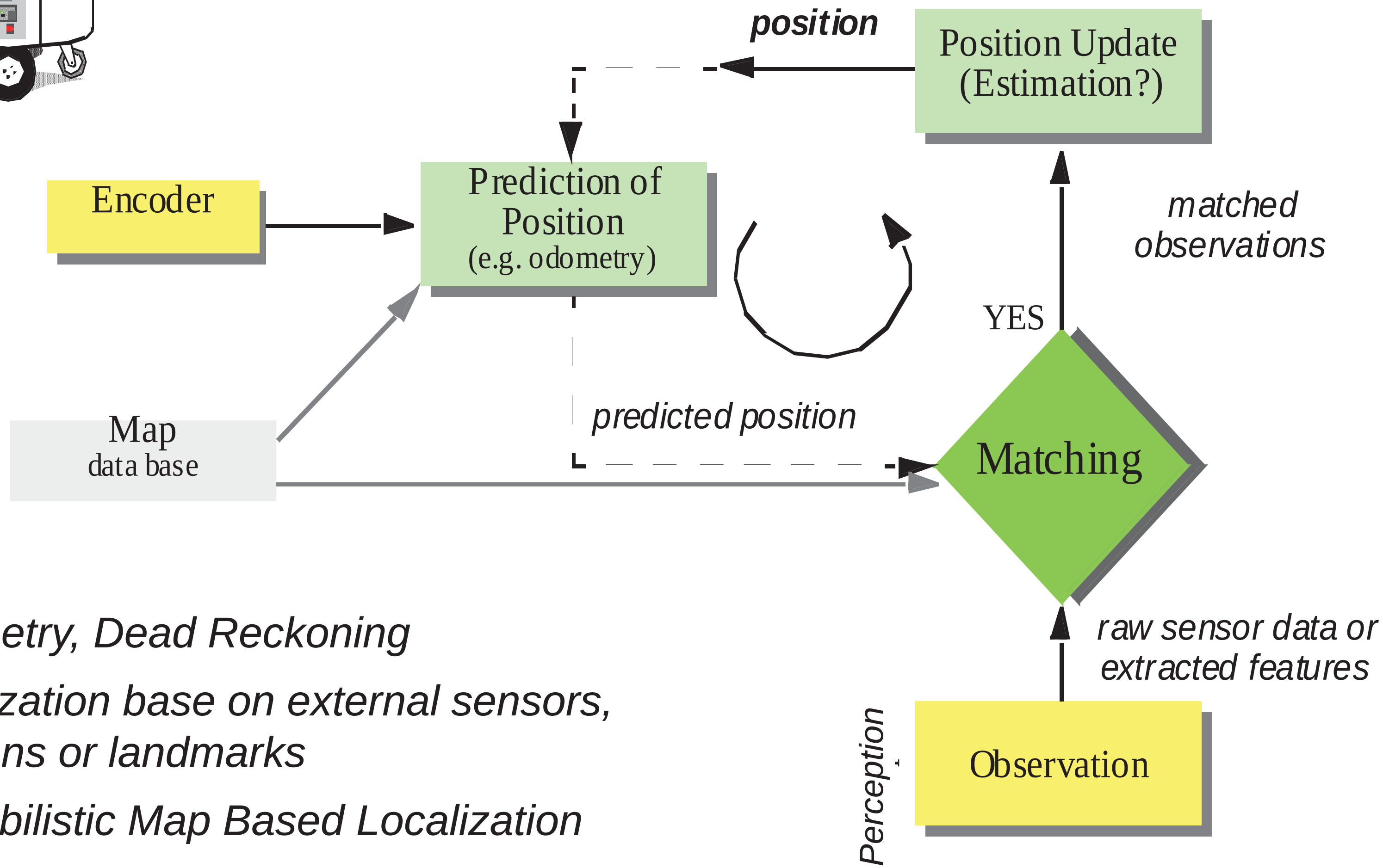
15 Probabilistic Localization (example Kalman Filter)

- Continuous, recursive and very compact



16 Map based localization (to be covered in details later)





- *Odometry, Dead Reckoning*
- *Localization base on external sensors, beacons or landmarks*
- *Probabilistic Map Based Localization*

17 Challenges of Localization

- Knowing the absolute position (e.g. GPS) is not sufficient
- Localization in human-scale in relation with environment
- Planning in the *Cognition* step requires more than only position as input
- Perception and motion plays an important role
  - Exteroceptive sensor noise
  - Effector noise
  - Odometric position estimation

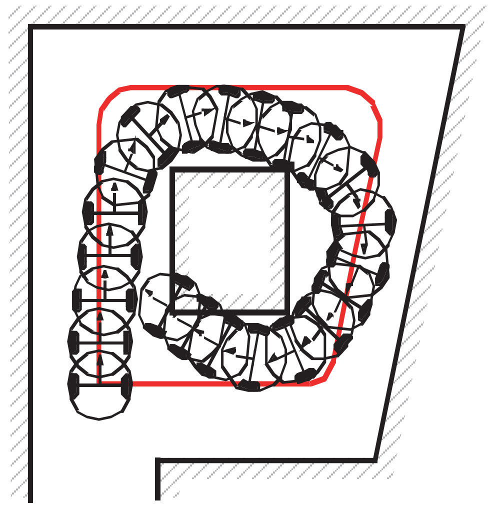
Exteroceptive Sensor Noise

- Sensor noise is mainly influenced by environment
  e.g. surface, illumination …

- and by the measurement principle itself
  e.g. interference between ultrasonic sensors

- Sensor noise drastically reduces the useful information of sensor readings.
  The solution is:
  - to model sensor noise appropriately
  - to take multiple readings into account
  - employ temporal and/or multi-sensor fusion

19 Effector Noise: Odometry, Deduced Reckoning



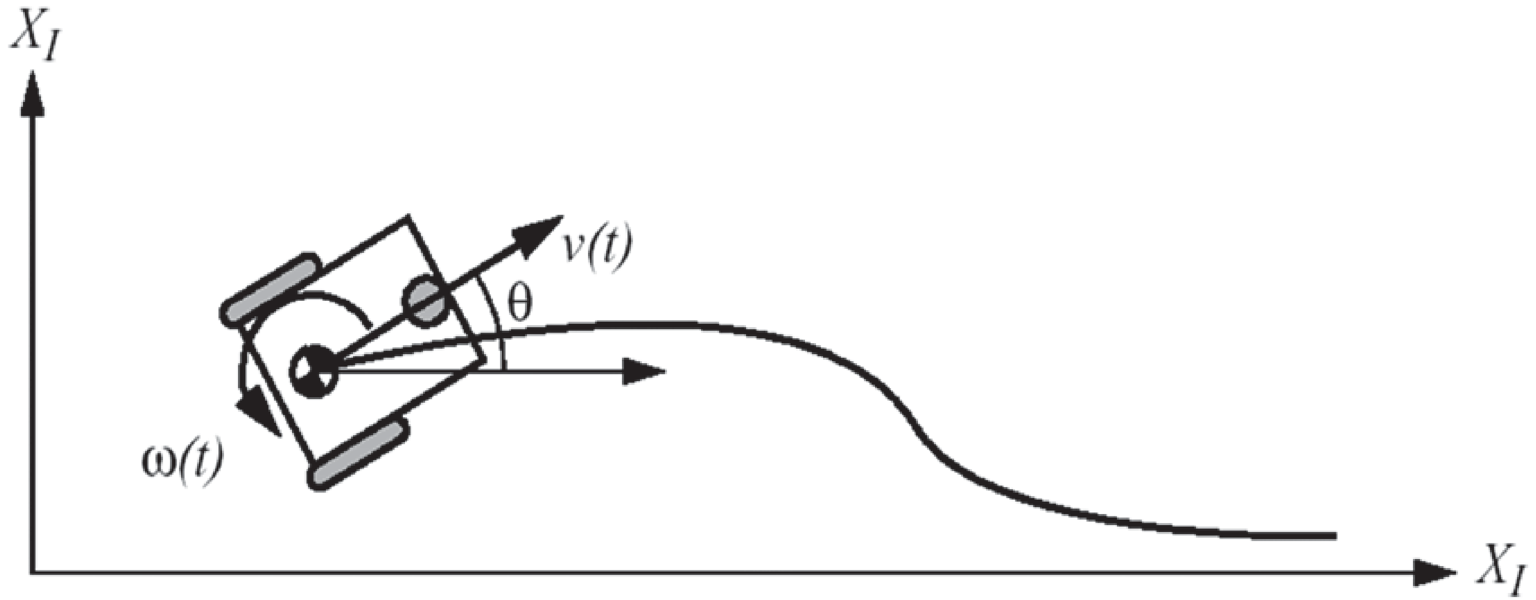
- Odometry and dead reckoning:
  Position update is based on proprioceptive sensors
  - Odometry: wheel sensors only
  - Dead reckoning: also heading sensors

- The movement of the robot, sensed with wheel encoders and/or heading sensors is integrated to the position.
  - Pros: Straight forward, easy
  - Cons: Errors are integrated -> unbound

- Using additional heading sensors (e.g. gyroscope) might help to reduce the cumulated errors, but the main problems remain the same.

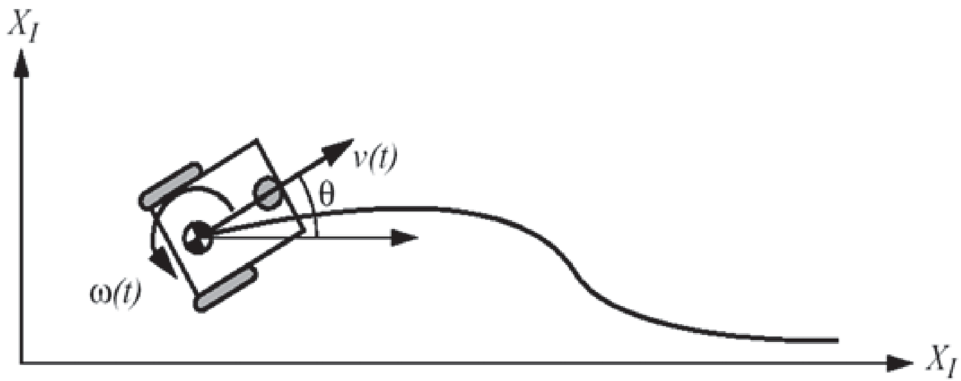
20 Odometry: The Differential Drive Robot (1)

$$p = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} \qquad p' = p + \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\theta \end{bmatrix}$$



21 Odometry: The Differential Drive Robot (2)

- Kinematics



$$\Delta x = \Delta s \cos(\theta + \Delta\theta / 2)$$

$$\Delta y = \Delta s \sin(\theta + \Delta\theta / 2)$$

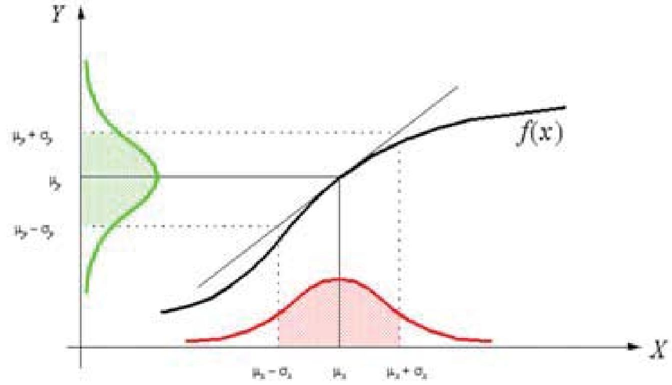
$$\Delta\theta = \frac{\Delta s_r - \Delta s_l}{b}$$

$$\Delta s = \frac{\Delta s_r + \Delta s_l}{2}$$

$$p' = f(x, y, \theta, \Delta s_r, \Delta s_l) = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix} + \begin{bmatrix} \frac{\Delta s_r + \Delta s_l}{2} \cos\left(\theta + \frac{\Delta s_r - \Delta s_l}{2b}\right) \\ \frac{\Delta s_r + \Delta s_l}{2} \sin\left(\theta + \frac{\Delta s_r - \Delta s_l}{2b}\right) \\ \frac{\Delta s_r - \Delta s_l}{b} \end{bmatrix}$$

22 Odometry: The Differential Drive Robot (3)

- Error model



$$\Sigma_\Delta = covar(\Delta s_r, \Delta s_l) = \begin{bmatrix} k_r|\Delta s_r| & 0 \\ 0 & k_l|\Delta s_l| \end{bmatrix}$$

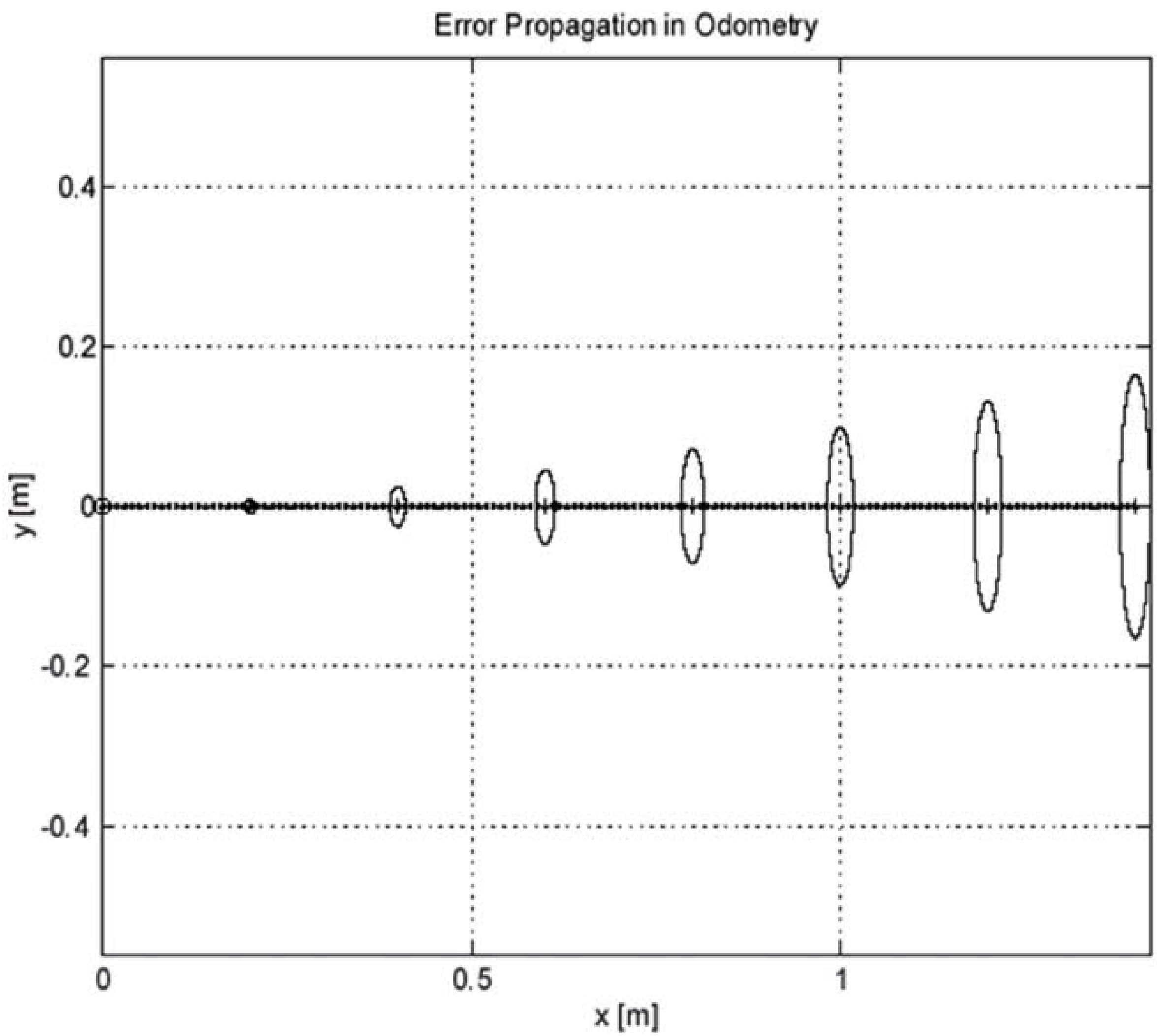
$$\Sigma_{p'} = \nabla_p f \cdot \Sigma_p \cdot \nabla_p f^T + \nabla_{\Delta_{rl}} f \cdot \Sigma_\Delta \cdot \nabla_{\Delta_{rl}} f^T$$

$$F_p = \nabla_p f = \nabla_p(f^T) = \begin{bmatrix} \frac{\partial f}{\partial x} & \frac{\partial f}{\partial y} & \frac{\partial f}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\Delta s \sin(\theta + \Delta\theta/2) \\ 0 & 1 & \Delta s \cos(\theta + \Delta\theta/2) \\ 0 & 0 & 1 \end{bmatrix}$$

$$F_{\Delta_{rl}} = \begin{bmatrix} \frac{1}{2}\cos\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b}\sin\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2}\cos\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b}\sin\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{2}\sin\left(\theta + \frac{\Delta\theta}{2}\right) + \frac{\Delta s}{2b}\cos\left(\theta + \frac{\Delta\theta}{2}\right) & \frac{1}{2}\sin\left(\theta + \frac{\Delta\theta}{2}\right) - \frac{\Delta s}{2b}\cos\left(\theta + \frac{\Delta\theta}{2}\right) \\ \frac{1}{b} & -\frac{1}{b} \end{bmatrix}$$

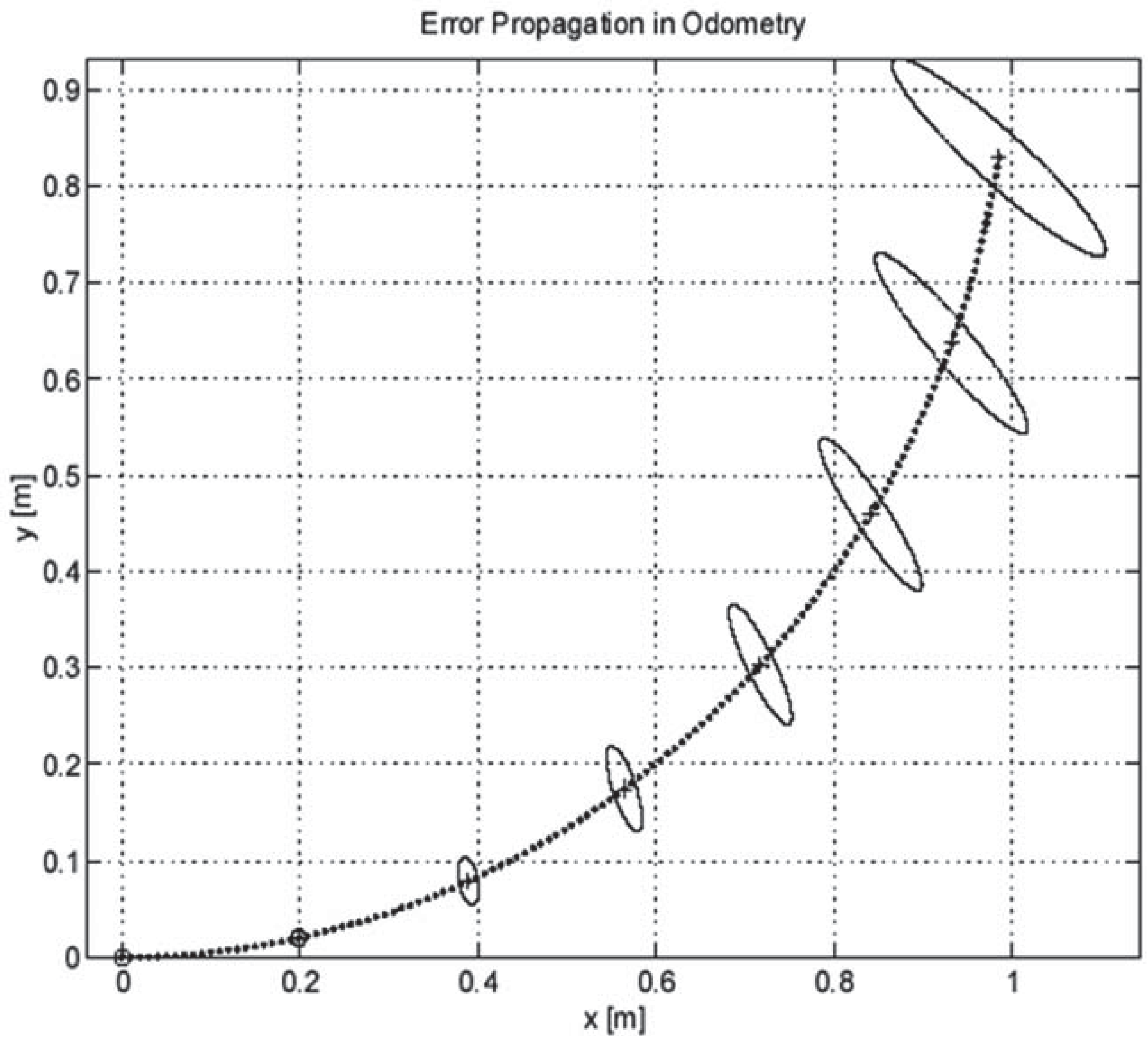
23 Odometry: Growth of Pose uncertainty for Straight Line Movement

- Note: Errors perpendicular to the direction of movement are growing much faster!



Odometry: Growth of Pose uncertainty for Movement on a Circle

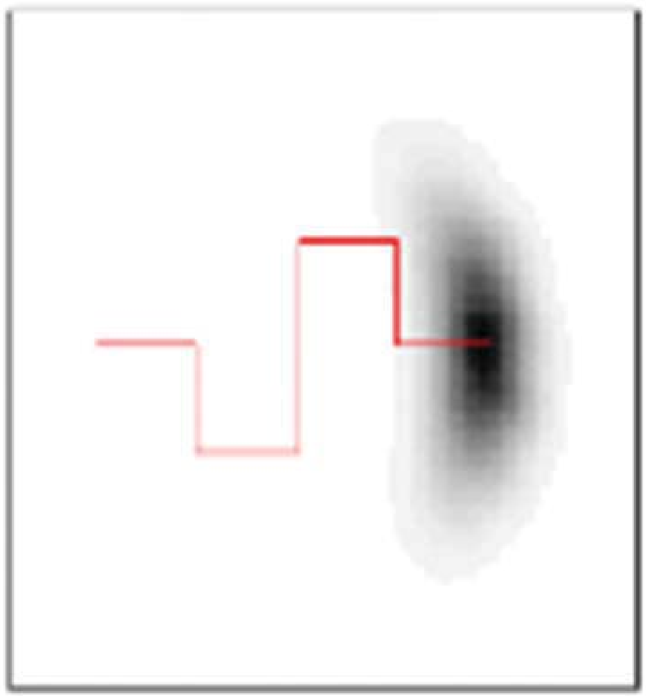
- Note: Errors ellipse in does not remain perpendicular to the direction of movement!



25 Odometry: example of non-Gaussian error model

- Note: Errors are not shaped like ellipses!

Courtesy AI Lab, Stanford



[Fox, Thrun, Burgard, Dellaert, 2000]

26 Odometry: Error sources

deterministic (systematic) ⟷ non-deterministic (non-systematic)

- deterministic errors can be eliminated by proper calibration of the system.
- non-deterministic errors have to be described by error models and will always lead to uncertain position estimate.

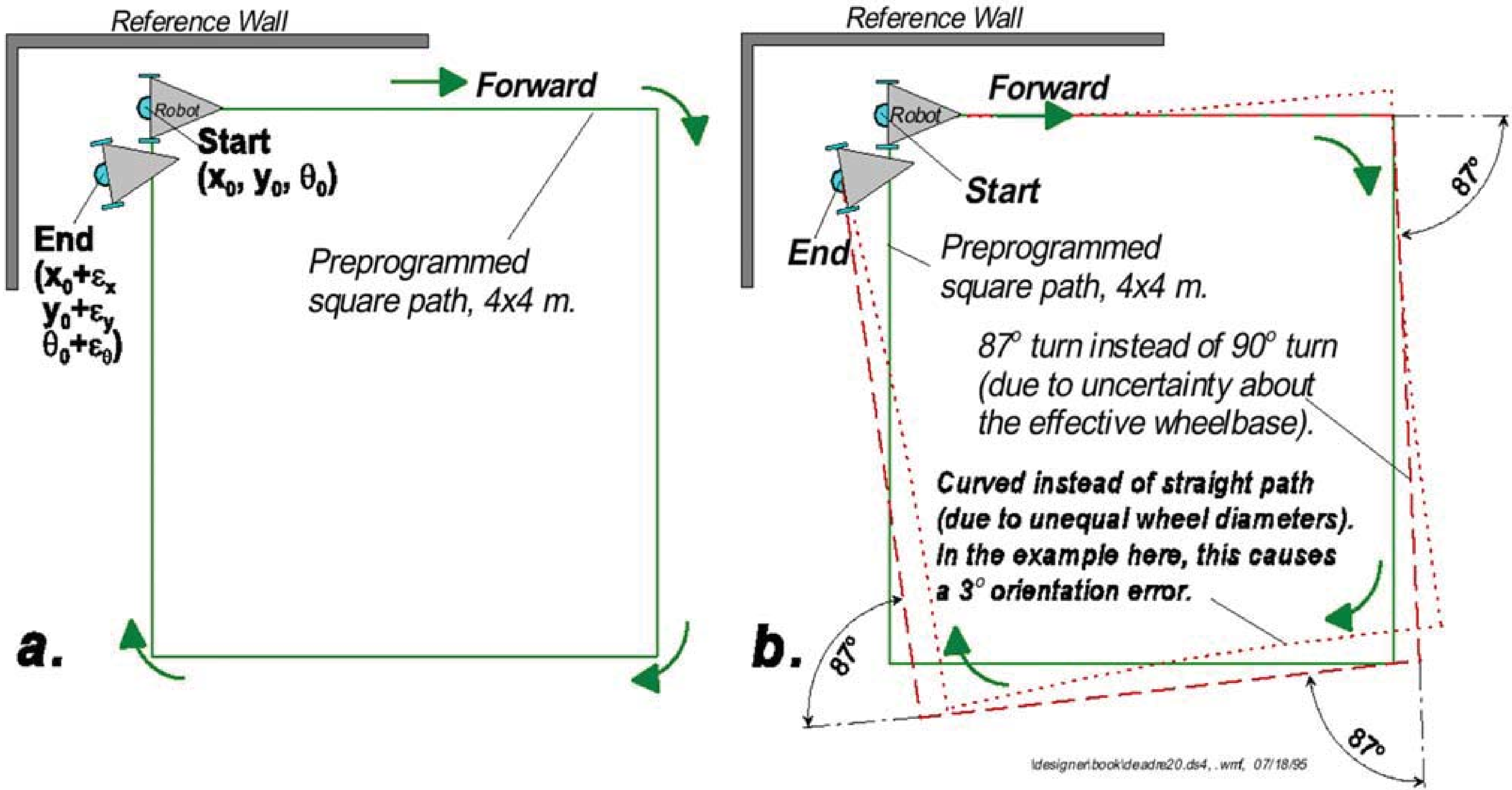
- Major Error Sources in Odometry:
  - Limited resolution during integration (time increments, measurement resolution)
  - Misalignment of the wheels (deterministic)
  - Unequal wheel diameter (deterministic)
  - Variation in the contact point of the wheel
  - Unequal floor contact (slipping, not planar …)

27 Odometry: Classification of Integration Errors

- Range error: integrated path length (distance) of the robots movement
  - sum of the wheel movements
- Turn error: similar to range error, but for turns
  - difference of the wheel motions
- Drift error: difference in the error of the wheels leads to an error in the robots angular orientation
- **Over long periods of time, turn and drift errors** far outweigh range errors!
  - Consider moving forward on a straight line along the x axis. The error in the y-position introduced by a move of d meters will have a component of $d \sin \Delta q$, which can be quite large as the angular error Δq grows.

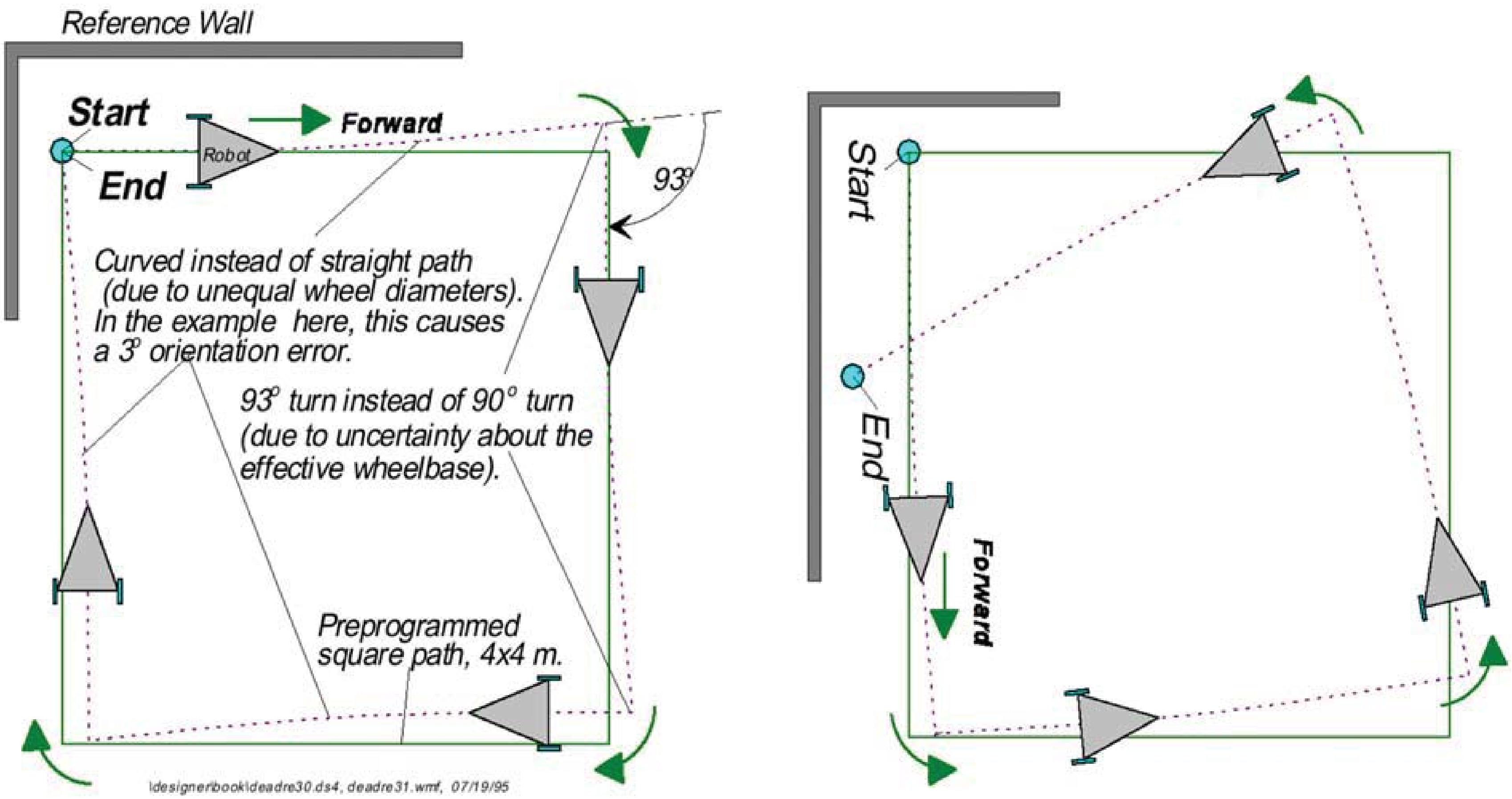
28 Odometry: Calibration of Errors I (Borenstein [5])

- The unidirectional square path experiment



29 Odometry: Calibration of Errors II (Borenstein [5])

- The bi-directional square path experiment

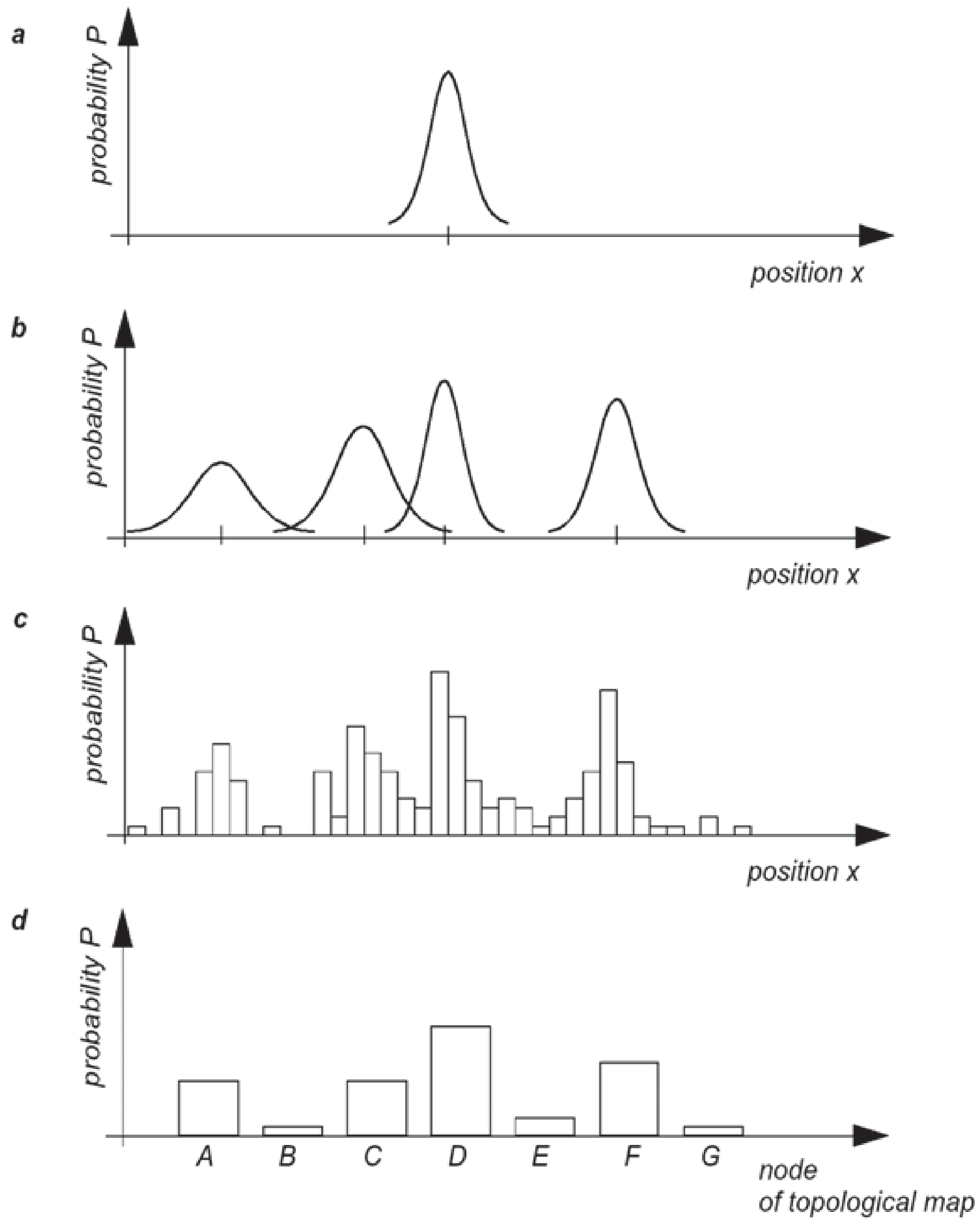


Belief representation

How do we represent the robot position,
where the robot ‘believes” to be?

31 Belief Representation

- a) Continuous map
  with single hypothesis
  probability distribution
- b) Continuous map
  with multiple hypothesis
  probability distribution
- c) Discretized map
  with probability distribution
- d) Discretized topological
  map with probability
  distribution

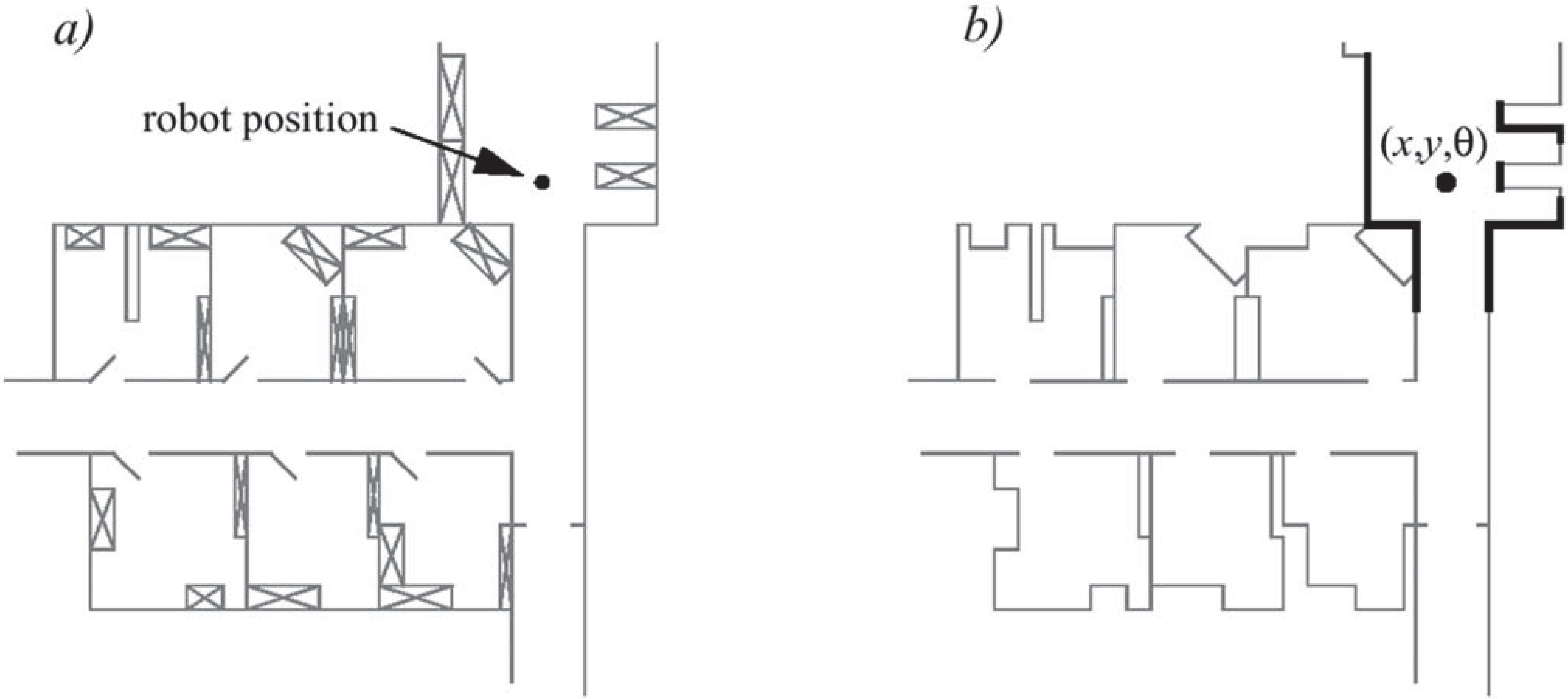


Belief Representation: Characteristics

- Continuous
  - Precision bound by sensor data
  - Typically single hypothesis pose estimate
  - Lost when diverging (for single hypothesis)
  - Compact representation and typically reasonable in processing power.

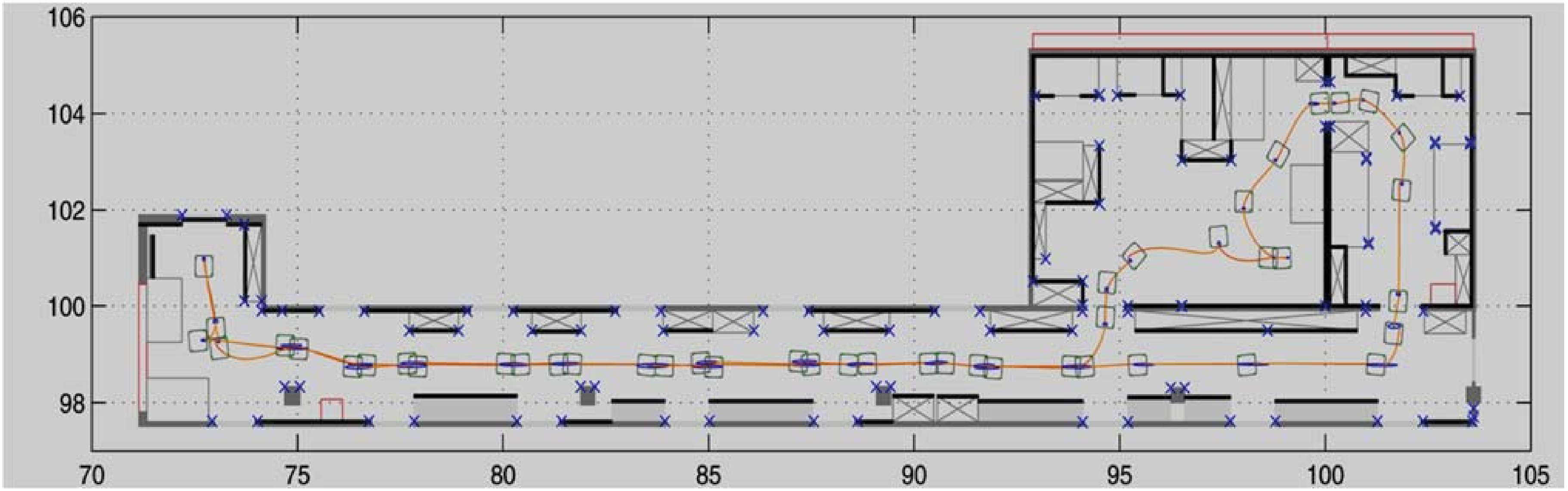
- Discrete
  - Precision bound by resolution of discretisation
  - Typically multiple hypothesis pose estimate
  - Never lost (when diverges converges to another cell)
  - Important memory and processing power needed. (not the case for topological maps)

33 Single-hypothesis Belief – Continuous Line-Map

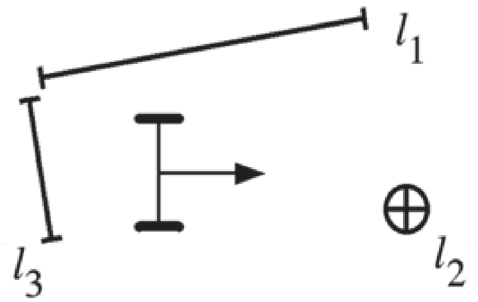


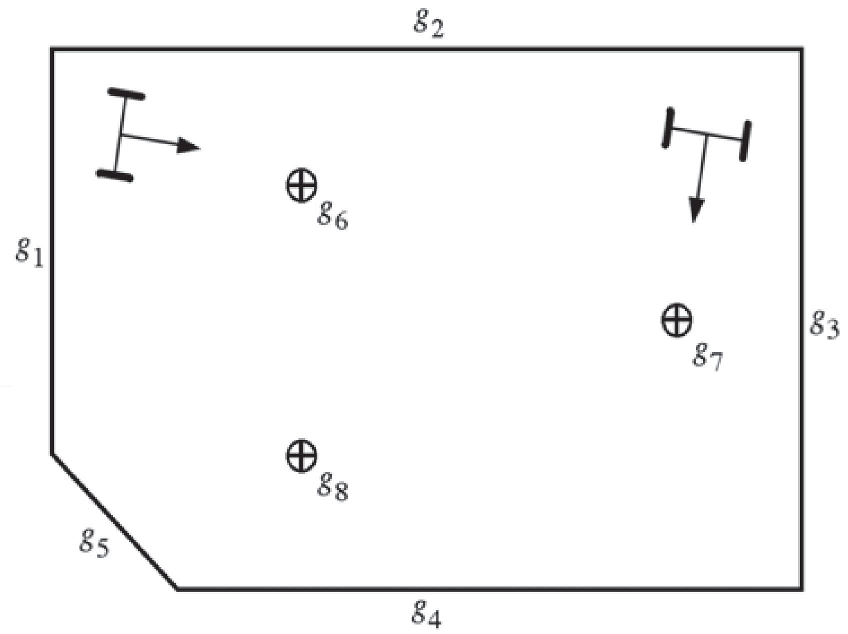
34 Autonomous Indoor Navigation (Pygmalion EPFL)

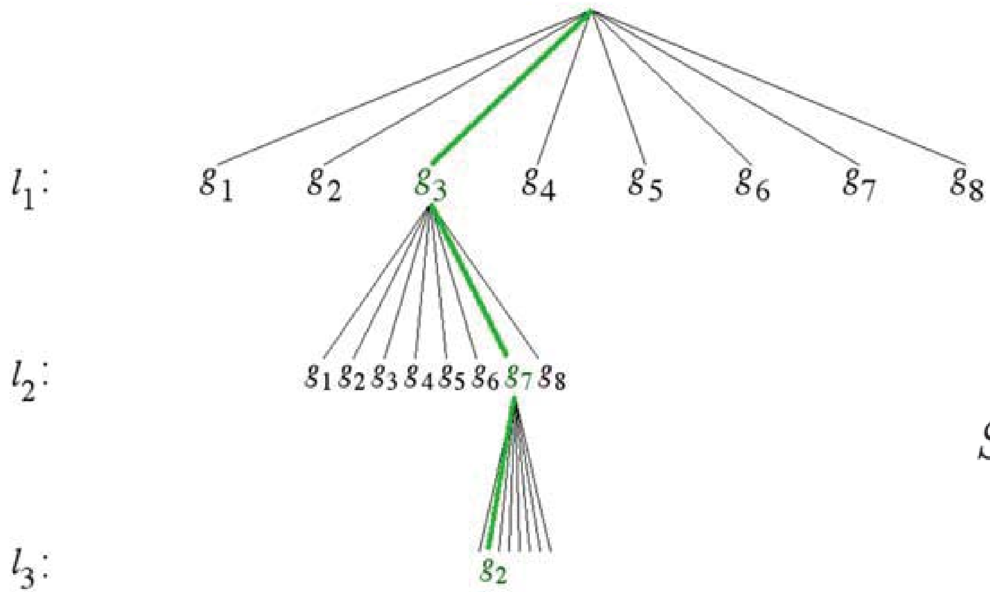
- very robust on-the-fly localization
- one of the first systems with probabilistic sensor fusion
- 47 steps,78 meter length, realistic office environment,
- conducted 16 times > 1km overall distance
- partially difficult surfaces (laser), partially few vertical edges (vision)



35 Multiple-hypothesis Belief – Continuous Line-Map

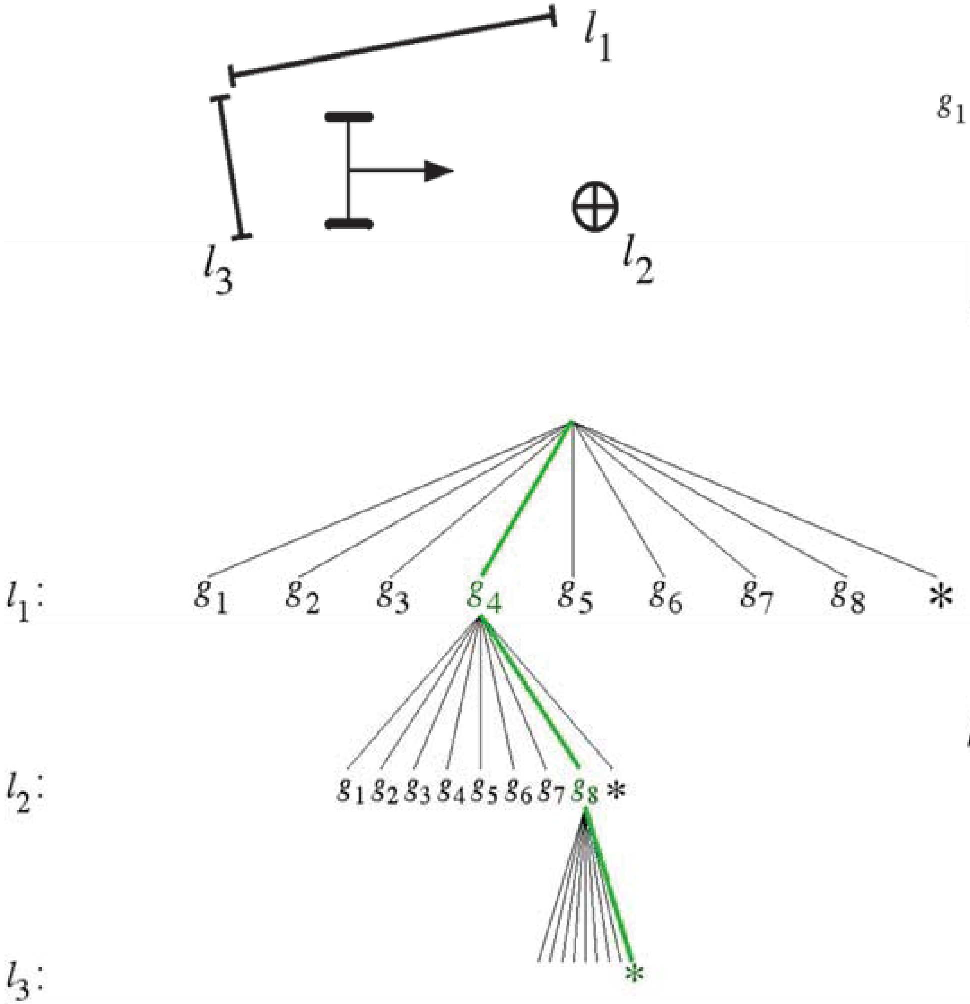


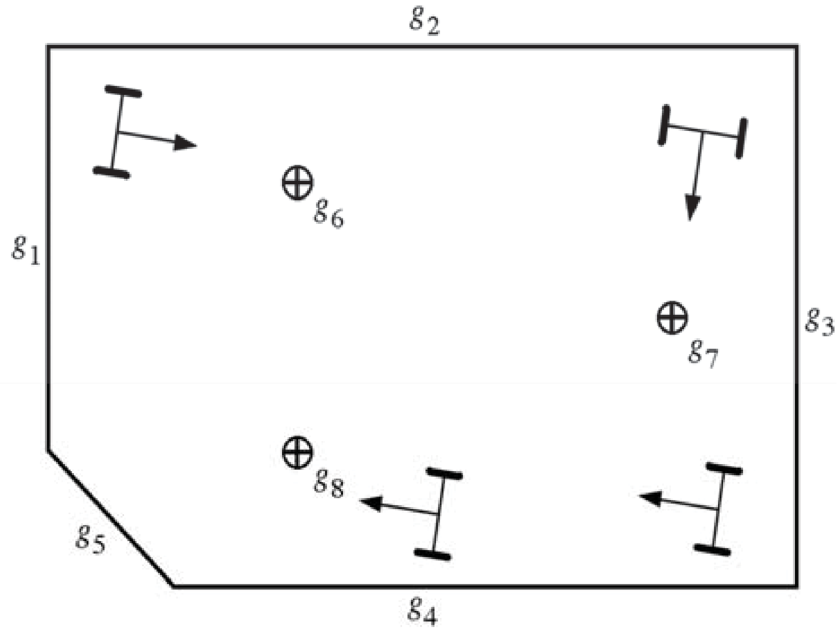




$$S_{h_2} = \{\{l_1, g_3\}, \{l_2, g_7\}, \{l_3, g_2\}\}$$

36 Multiple-hypothesis Belief – Continuous Line-Map



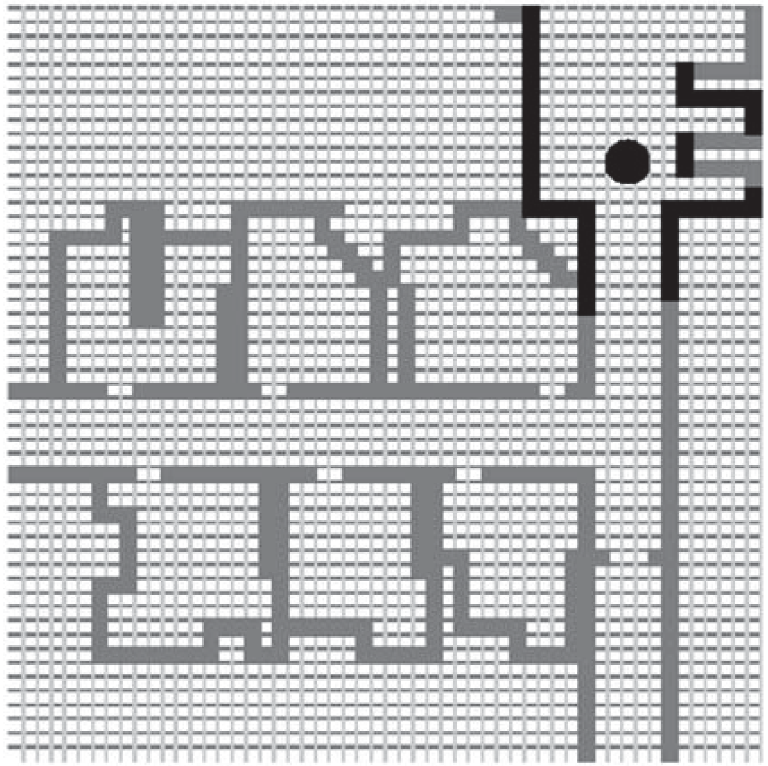


Environment dynamics

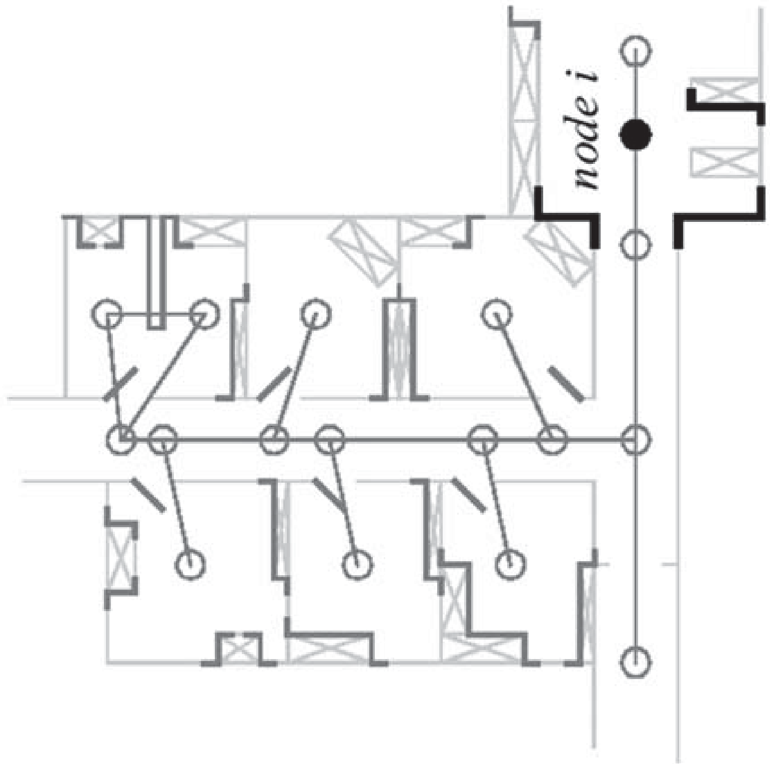
$$S_h = \{\{l_1, g_4\}, \{l_2, g_8\}, \{l_3, *\}\}$$

37 Single-hypothesis Belief – Grid and Topological Map

c)



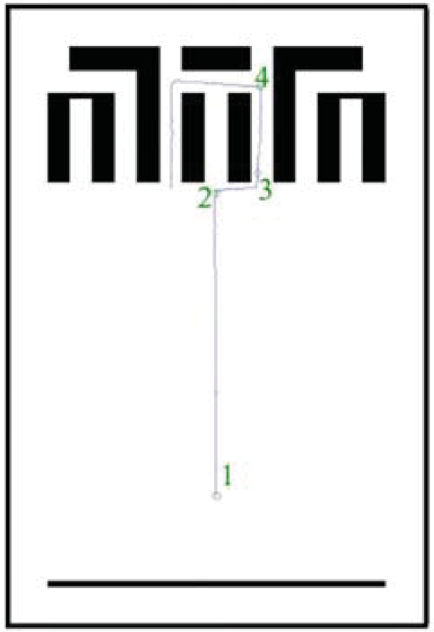
d)



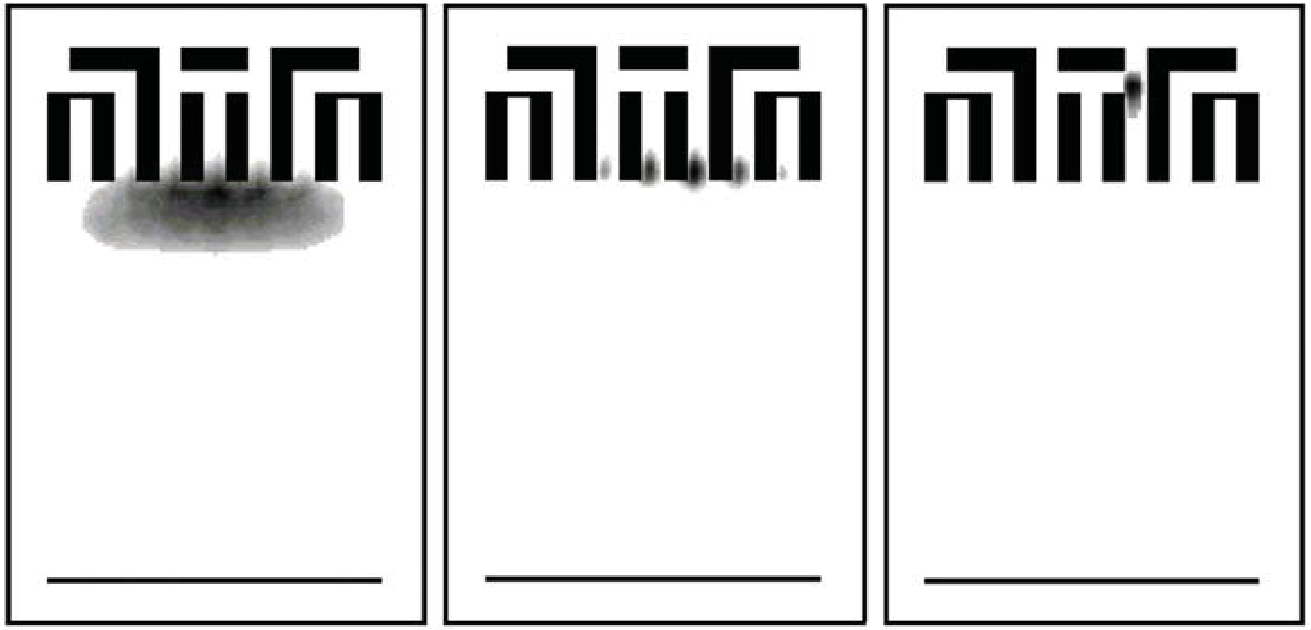
38 Grid-based Representation - Multi Hypothesis

- Grid size around 20 cm^2.

Courtesy of W. Burgard



Path of the robot

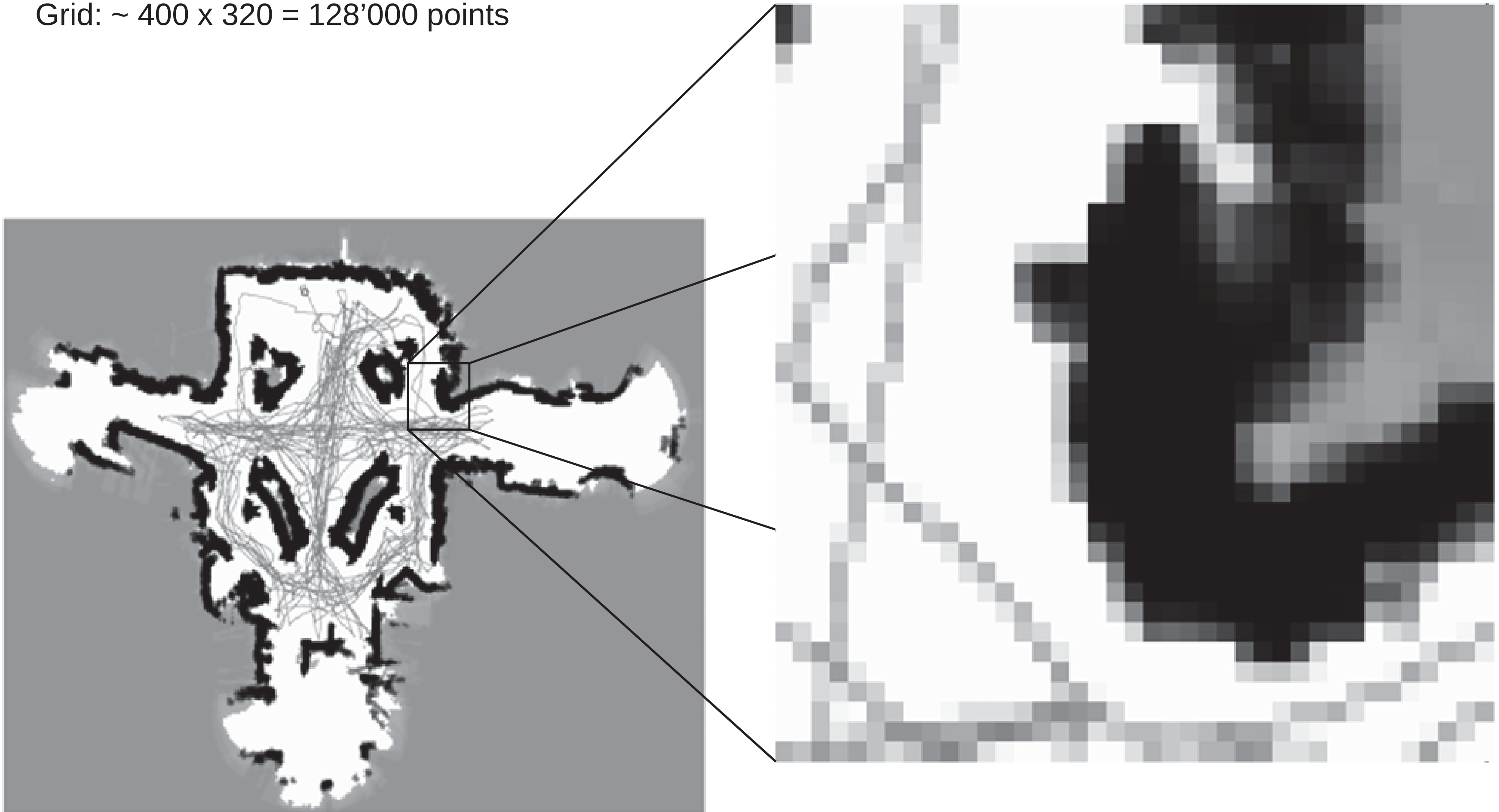


Belief states at positions 2, 3 and 4

39 Grid-Based Metric Approach

- Grid Map of the Smithsonian's National Museum of American History in Washington DC. (Courtesy of Wolfram Burger et al.)

Grid: ~ 400 x 320 = 128'000 points



Courtesy S. Thrun, W. Burgard

Map Representation

1. Map precision vs. application
2. Features precision vs. map precision
3. Precision vs. computational complexity

- Continuous Representation
- Decomposition (Discretisation)

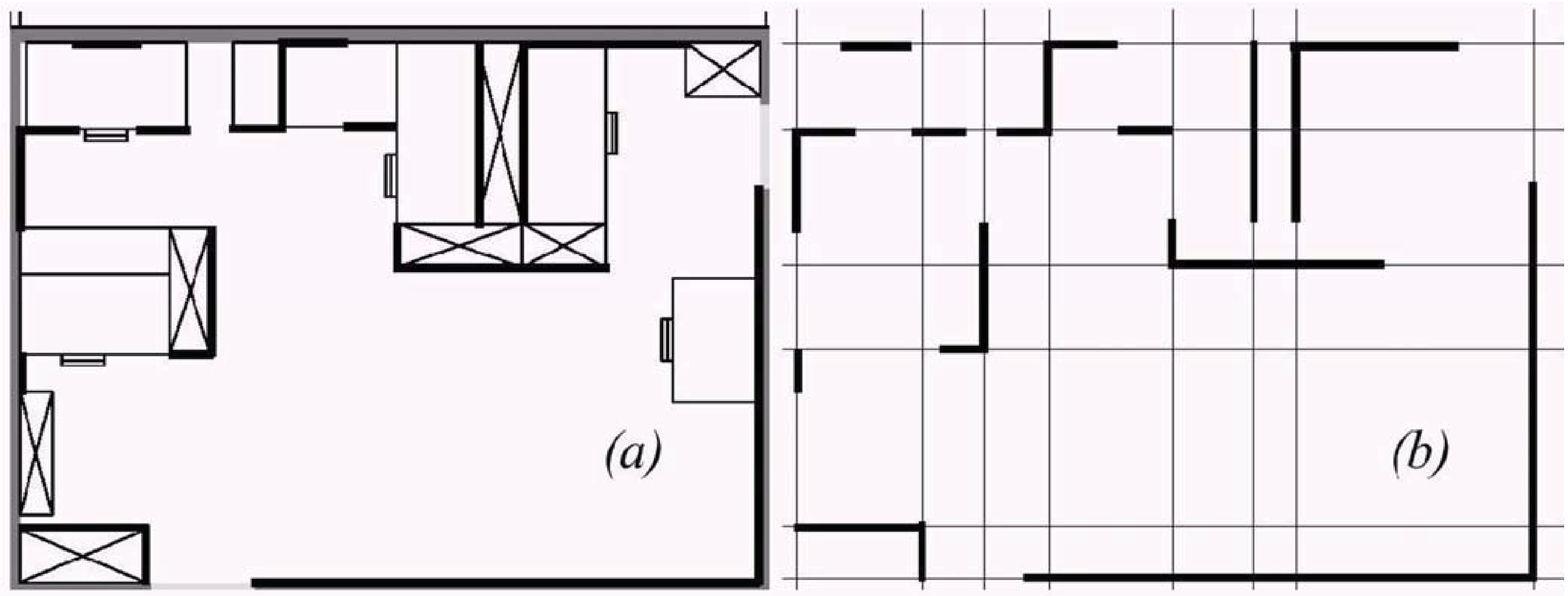
Representation of the Environment

- Environment Representation
  - Continuous Metric $\rightarrow$ x,y,θ
  - Discrete Metric $\rightarrow$ metric grid
  - Discrete Topological $\rightarrow$ topological grid

- Environment Modeling
  - Raw sensor data, e.g. laser range data, gray-scale images
    - large volume of data, low distinctiveness on the level of individual values
    - makes use of all acquired information
  - Low level features, e.g. line other geometric features
    - medium volume of data, average distinctiveness
    - filters out the useful information, still ambiguities
  - High level features, e.g. doors, a car, the Eiffel tower
    - low volume of data, high distinctiveness
    - filters out the useful information, few/no ambiguities, not enough information

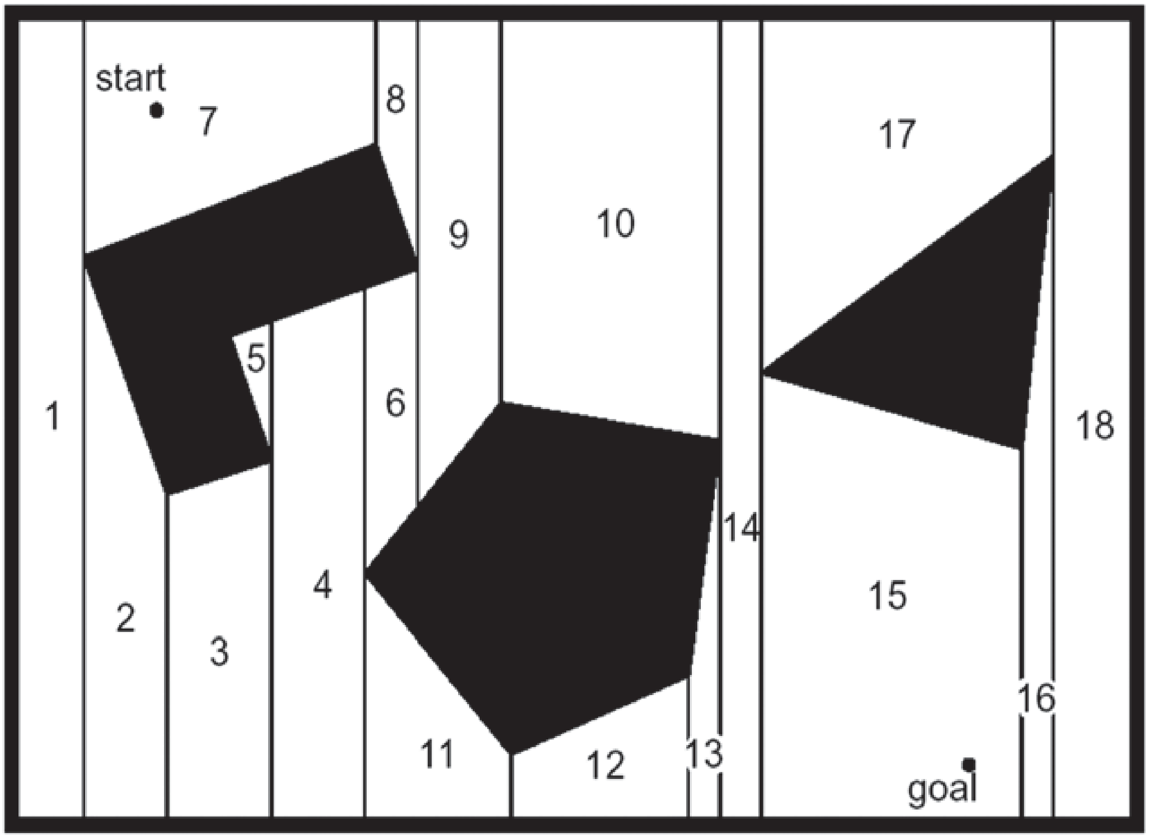
Map Representation: Continuous Line-Based

a) Architecture map
b) Representation with set of finite or infinite lines



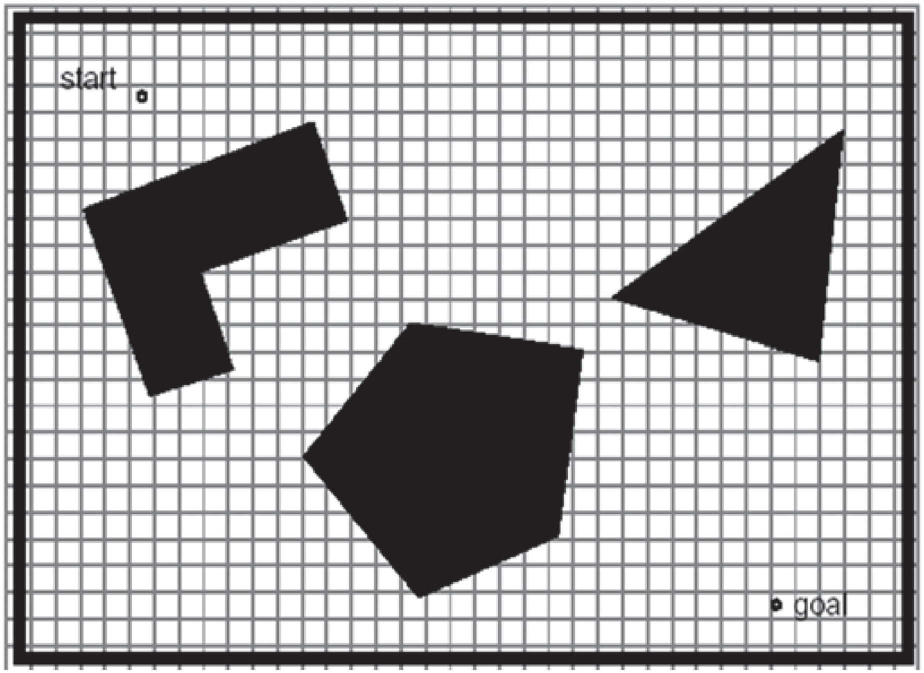
43 Map Representation: Exact cell decomposition

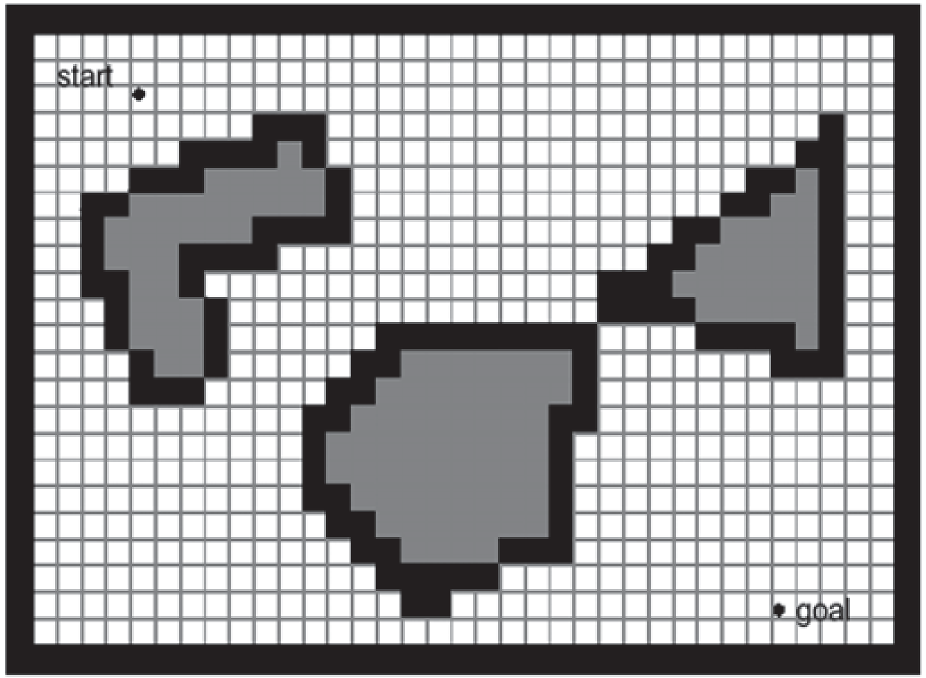
- Exact cell decomposition - Polygons



44 Map Representation: Approximate cell decomposition (1)

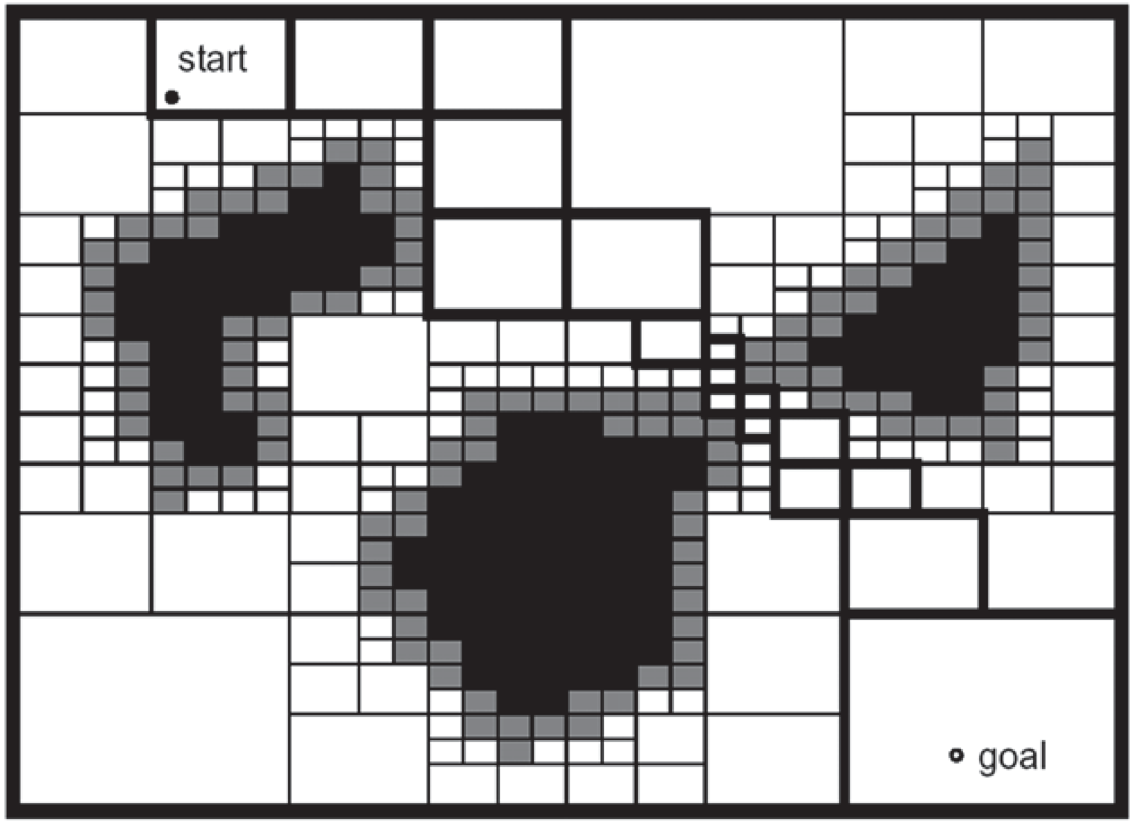
- Fixed cell decomposition
  - Narrow passages disappear





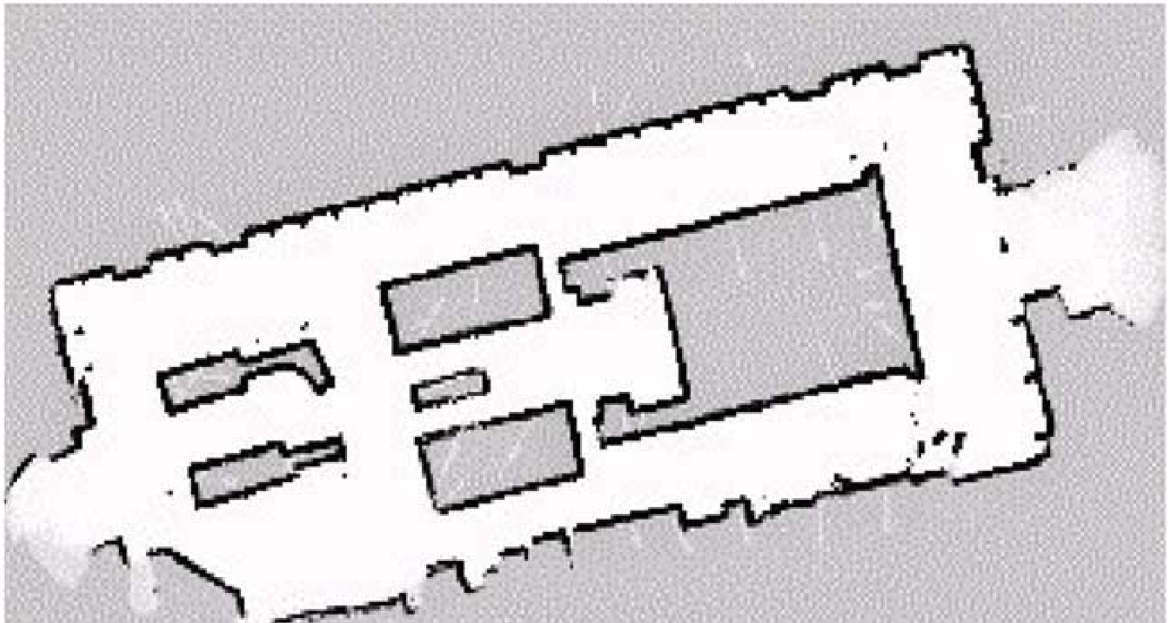
45 Map Representation: Adaptive cell decomposition (2)

- Exercise: how do we implement an adaptive cell decomposition algorithm?



46 Map Representation: Occupancy grid

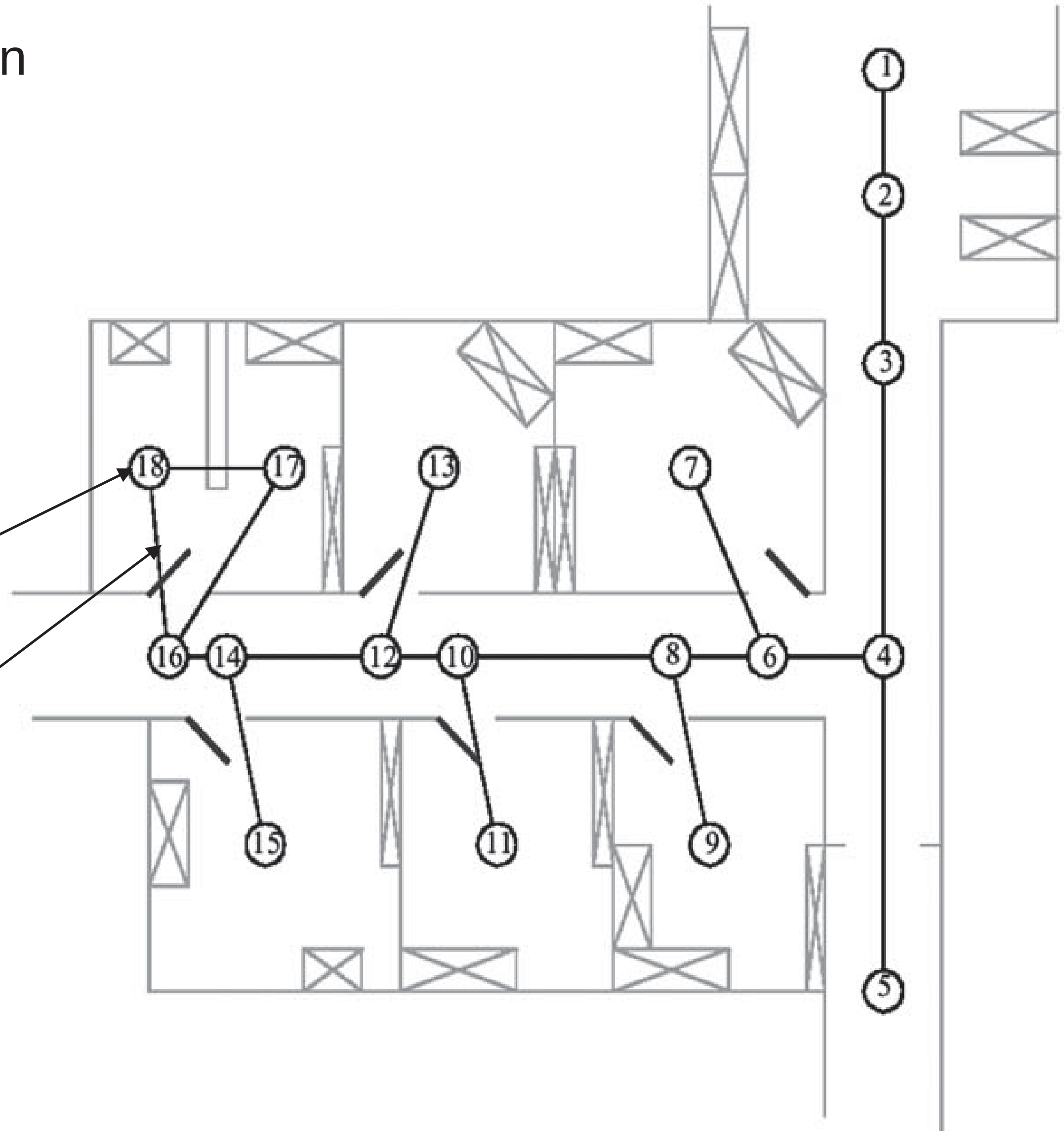
- Fixed cell decomposition: occupancy grid example
  - In occupancy grids, each cell may have a counter where 0 indicates that the cell has not been hit by any ranging measurements and therefore it is likely free-space. As the number of ranging strikes increases, the cell value is incremented and, above a certain threshold, the cell is deemed to be an obstacle
  - The values of the cells are discounted when a ranging strike travels through the cell. This allows us to represent “transient” (dynamic) obstacles



Courtesy of S. Thrun

47 Map Representation: Decomposition (6)

- Topological Decomposition

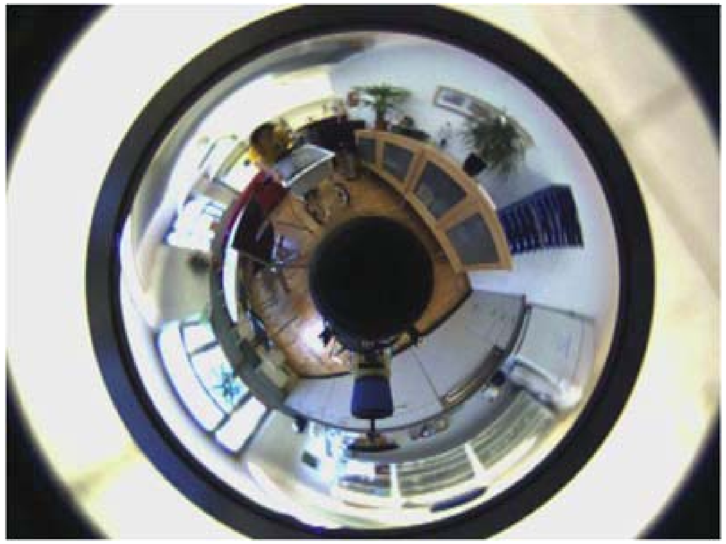


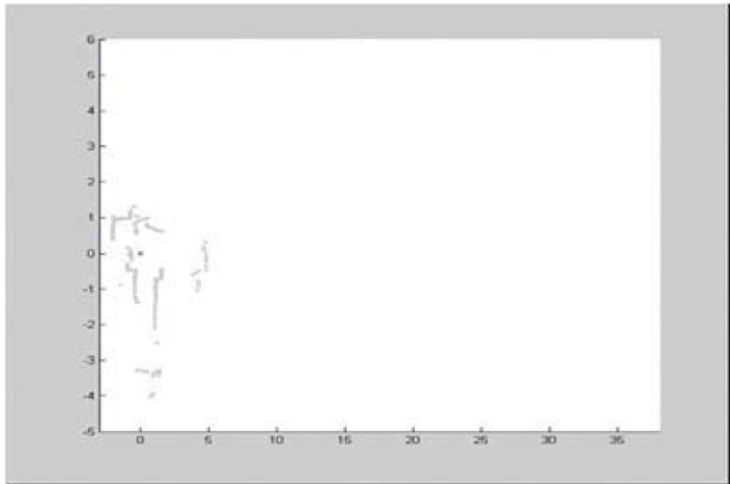
node
(location)

edge
(connectivity)

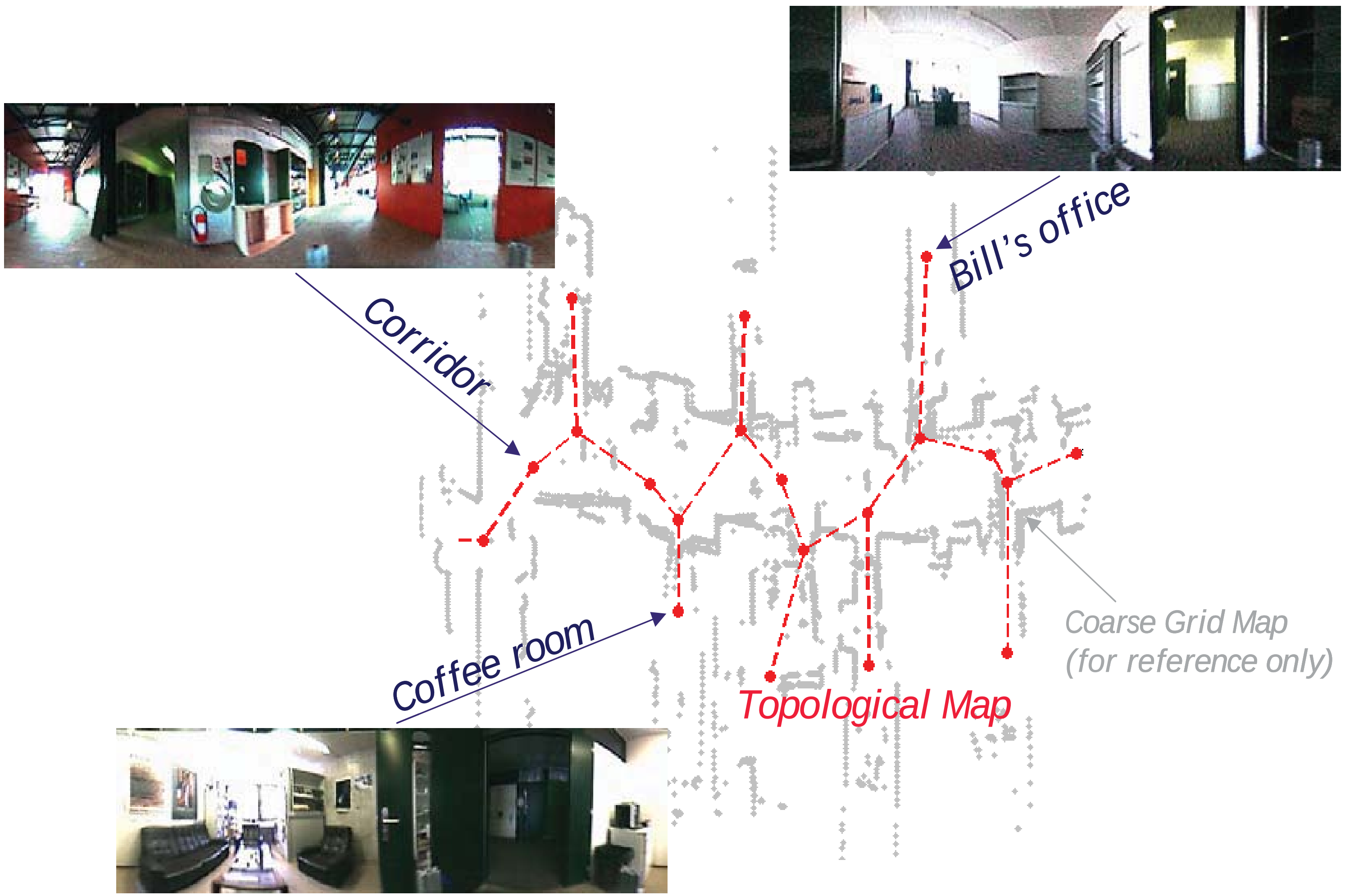
Example: Automatic Map building of topological map using vision

- Use SIFT features





49 Example: Automatic Map building of topological map using vision



Summary on Map representation

- Metric maps
  - Continuous
    - For example line based, point based, or plane based
  - Discrete
    - Exact cell decomposition
    - Approximate cell decomposition
      - Fixed cell decomposition (also occupancy grids)
      - Adaptive cell decomposition
- Topological
- Hybrid (mixture of metric and tolopological)

State-of-the-Art: Current Challenges in Map Representation

- Real world is dynamic
- Perception is still a major challenge
  - Error prone
  - Extraction of useful information difficult
- Traversal of open space
- How to build up topology (boundaries of nodes)
- Sensor fusion
- 2D...3D