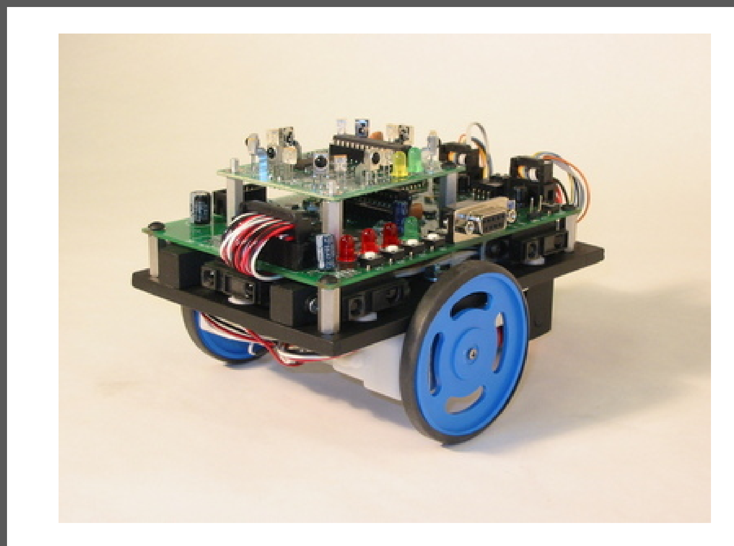
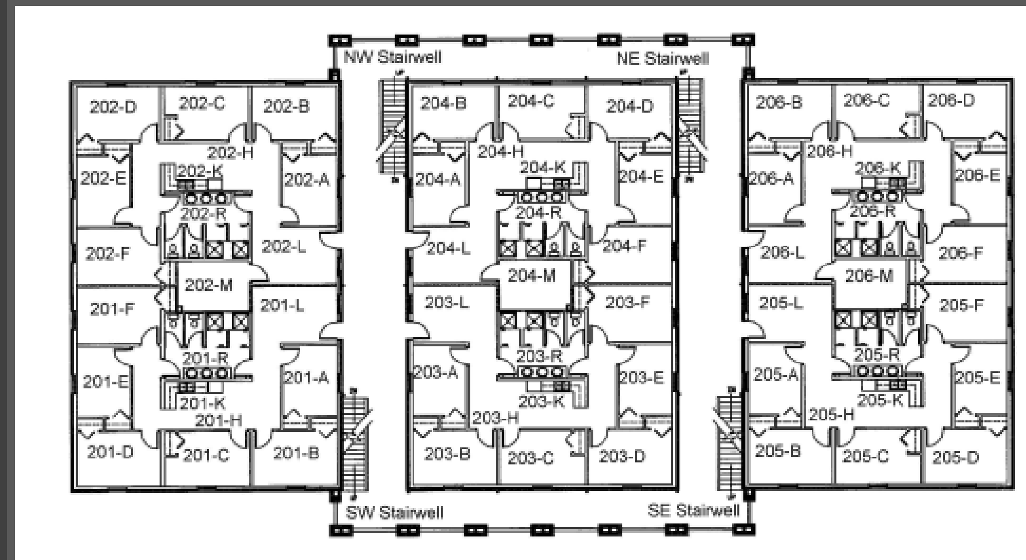
Mapping With Limited Sensing

Kris Beevers and Wes Huang

Algorithmic Robotics Laboratory
Department of Computer Science
Rensselaer Polytechnic Institute



The tool

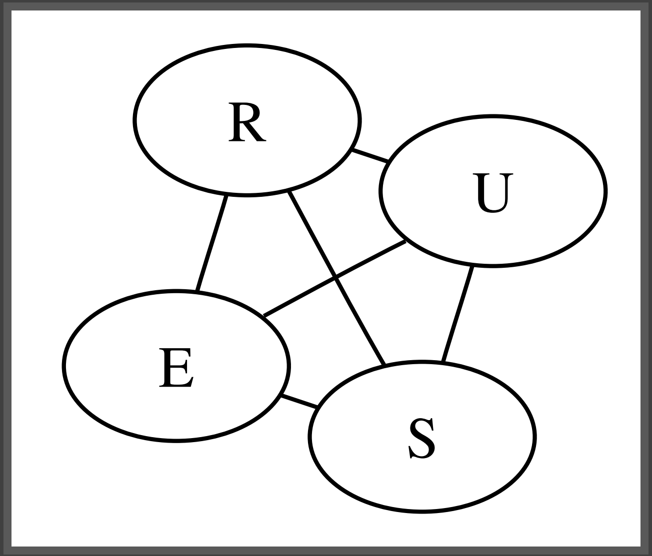


The objective

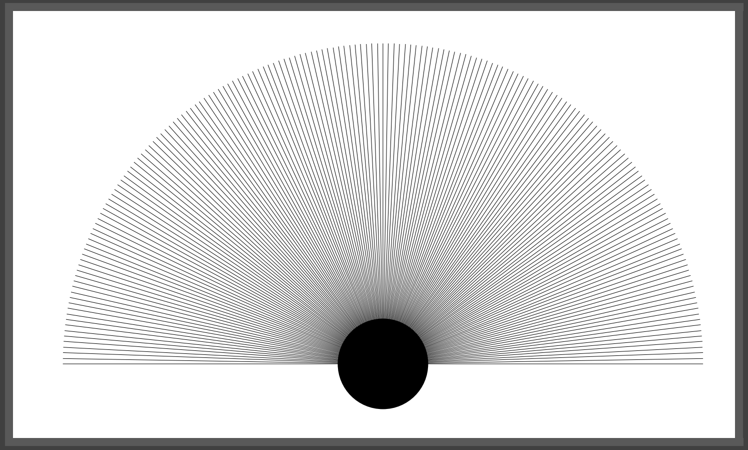
Goal: build an environment model with a robot

Issues in robot mapping

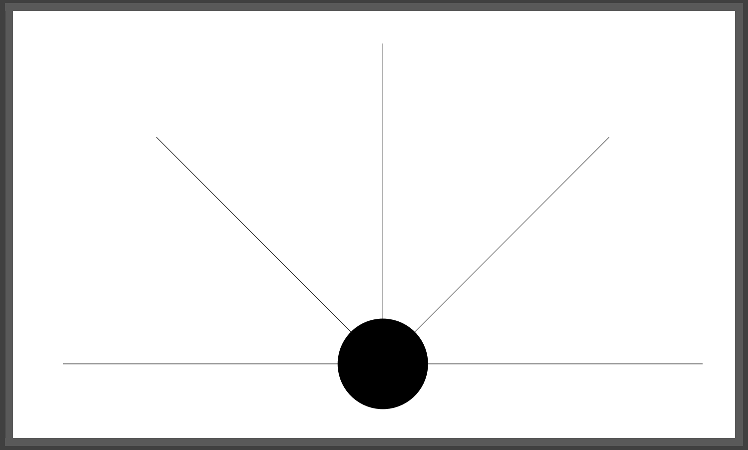
- **E**xploration strategies
- **R**epresentation
- **U**ncertainty management
- **S**ensing limitations



Sensing limitations



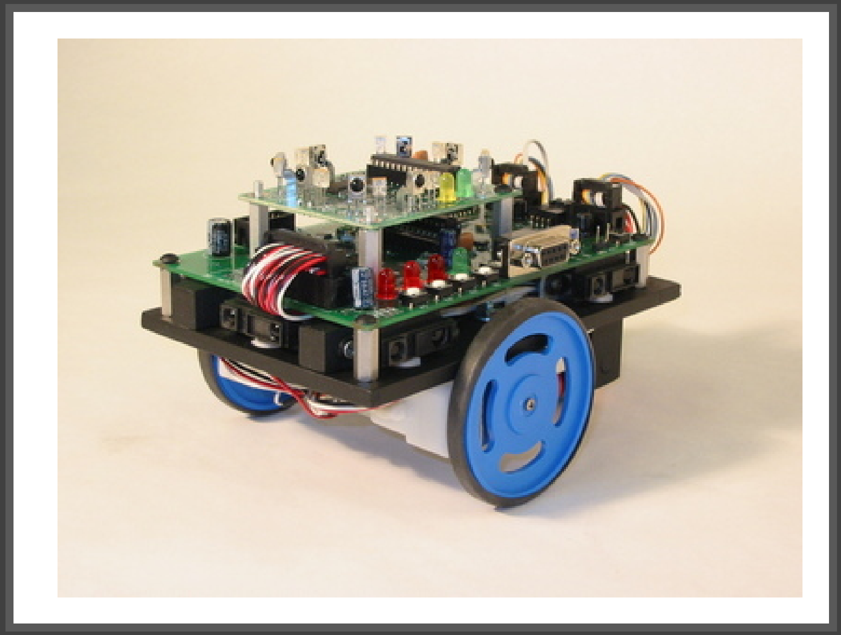
$\sim \$5000$

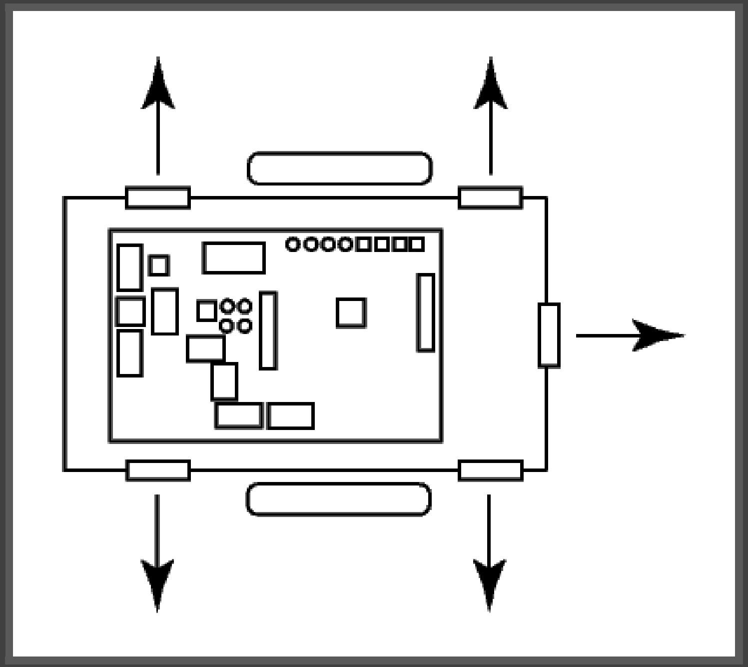


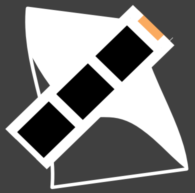
$\sim \$40$

- Sparse, short range
- Cheaper, lighter, less power
- For example: an array of a few IR sensors

Our robots







Our work

- Topological mapping with limited sensing
- Topological map merging
- SLAM with sparse sensing

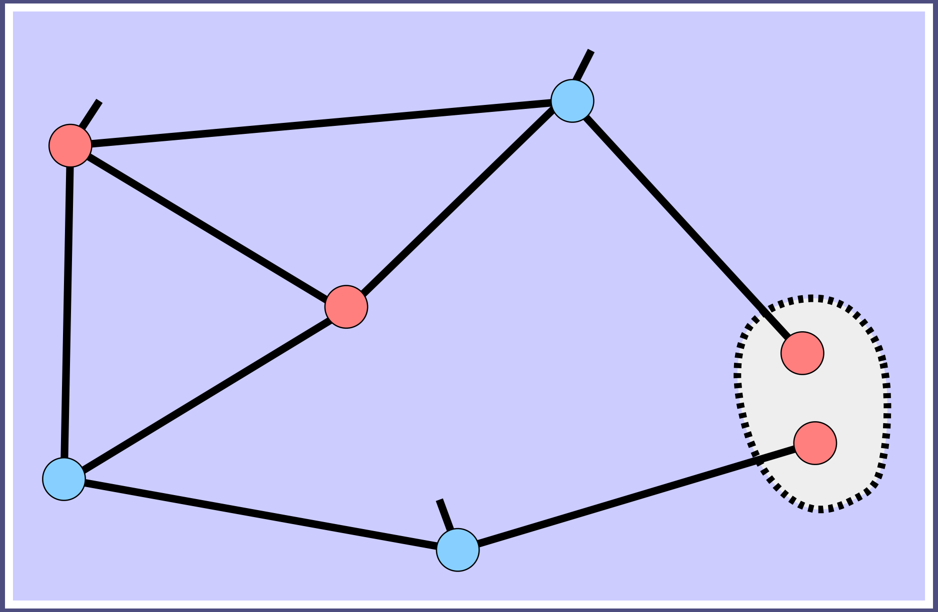
Topological mapping with limited sensing

- Map representations
- Exploration strategies

$\hookrightarrow$ WAFR 2004
$\hookrightarrow$ RPICS-TR 2005

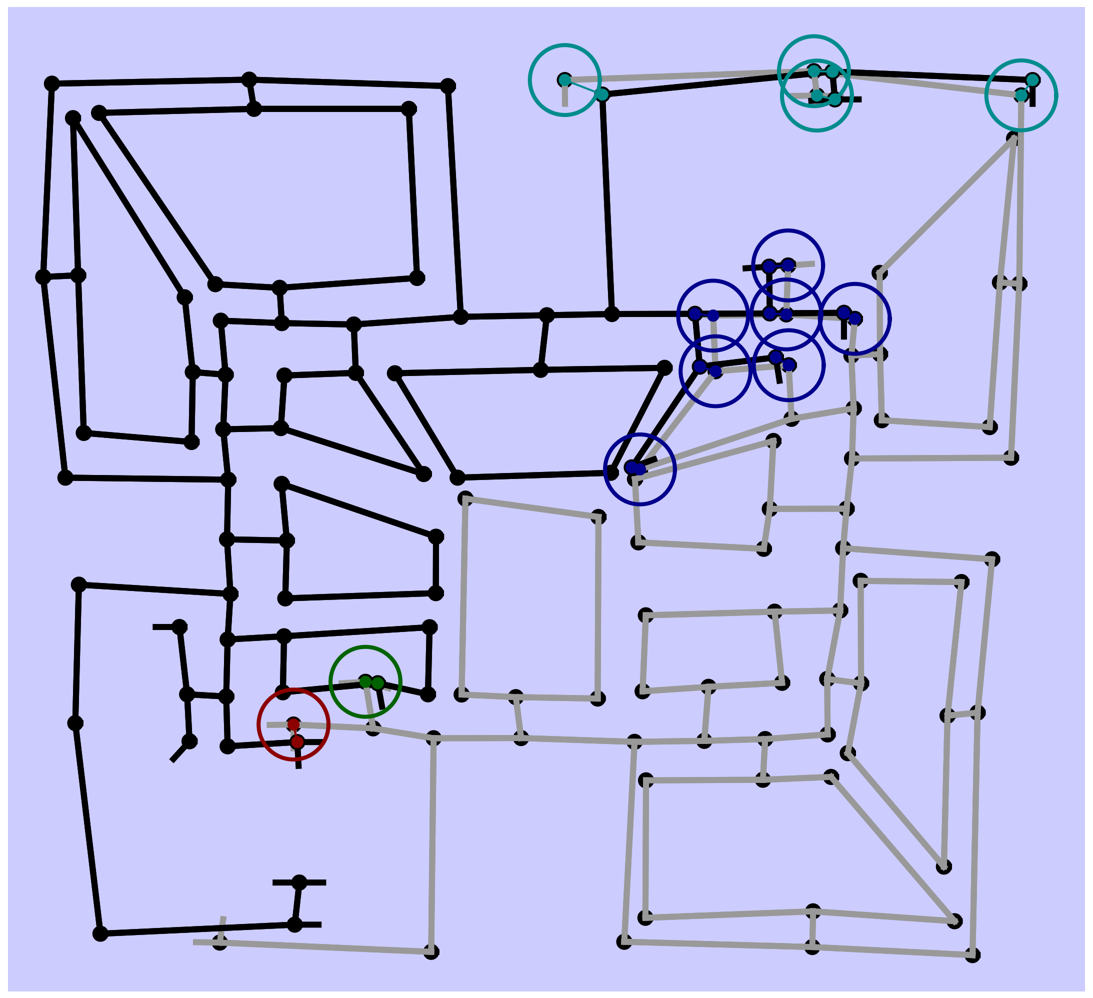
- Loop closing

$\hookrightarrow$ ICRA 2005



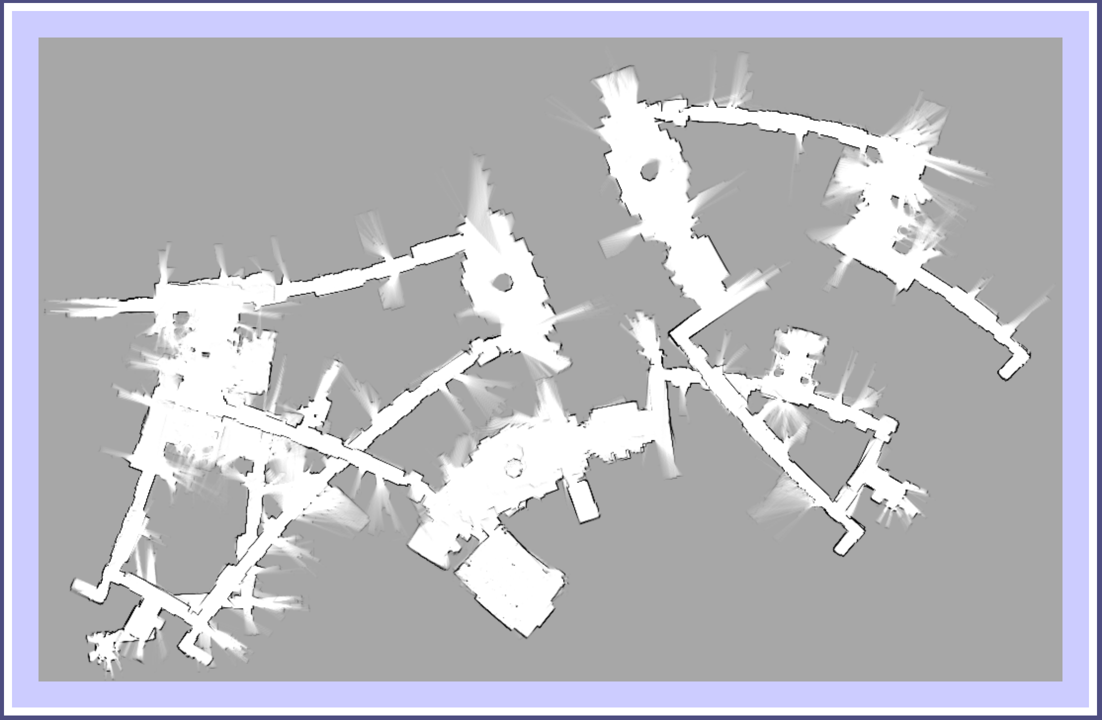
Topological map merging

- DARS 2004
- IJRR 2005

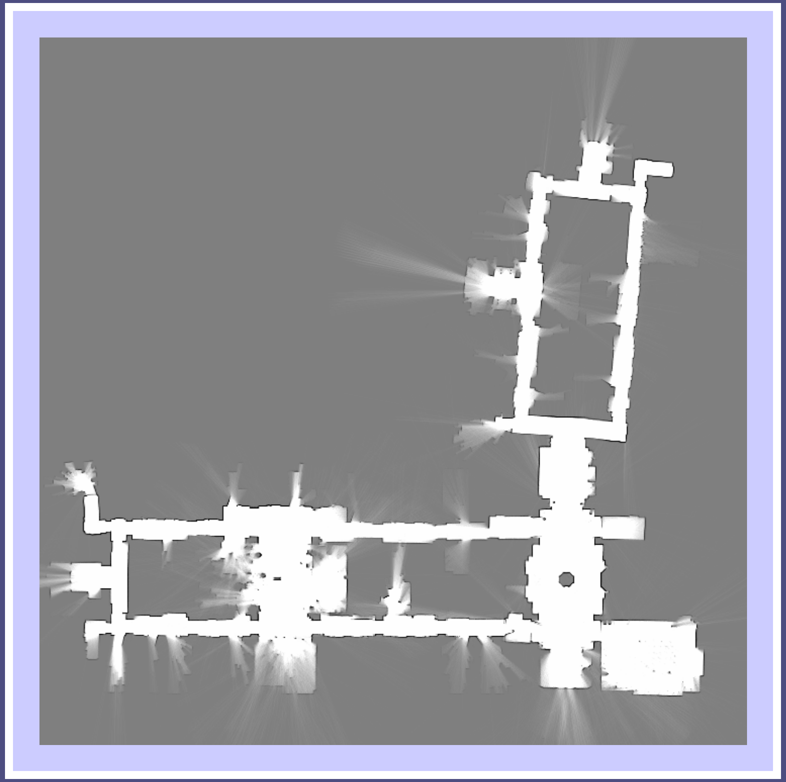


SLAM with sparse sensing
(ICRA 2006, submitted)

Courtesy B. Gerkey



The problem



The goal

A brief SLAM tutorial

The SLAM model

1. Move
2. Predict
3. Sense
4. **Update** pose/map

$$\boxed{x_k = [x_r(k)\ x_f(k)]}$$

- (Usually) *no* exploration strategy
- Maps: **landmarks** or occupancy grids

SLAM is hard!

$$P(x_k | z_k, u_k, n_k) \approx$$

$$P(z_k | x_k, \theta_{n_k}, n_k) \times$$

$$\int dx_{k-1} \left(\begin{array}{l} P(x_k | x_{k-1}, u_k) \times \\ \quad P(x_{k-1} | z_{k-1}, u_{k-1}, n_{k-1}) \end{array} \right)$$

Computing the SLAM update

- **Extended Kalman Filter (EKF)**

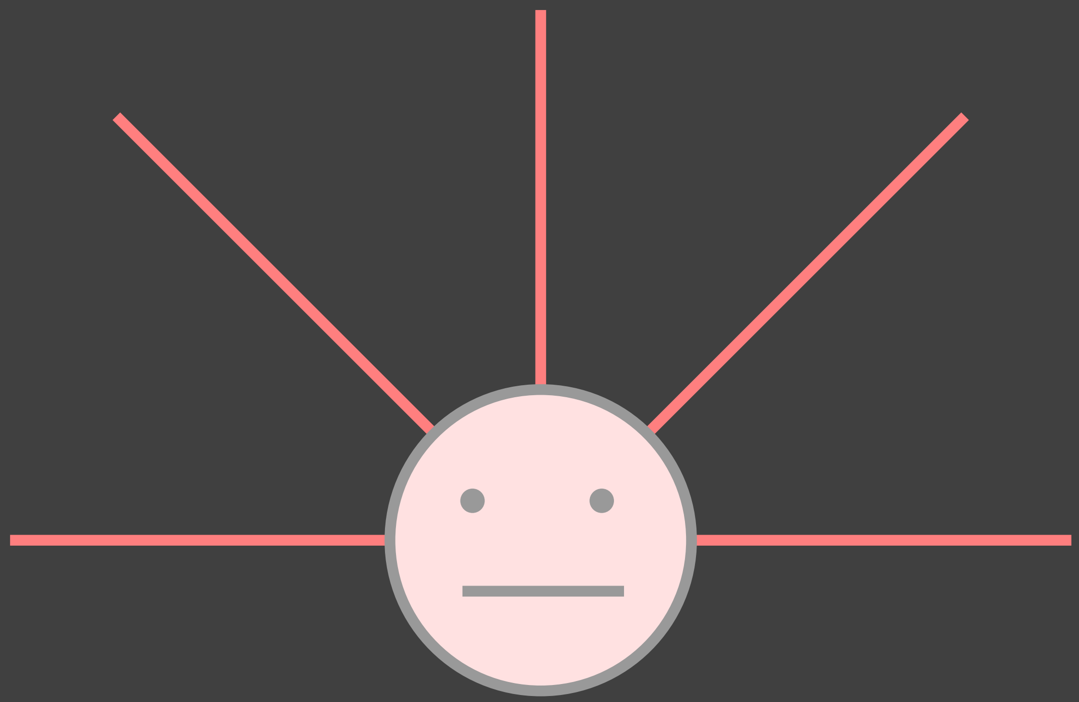
$\hookrightarrow$ Assume distributions are Gaussian

$\hookrightarrow$ $O(n^2)$

- **Particle Filter**

$\hookrightarrow$ Monte Carlo integration

$\hookrightarrow$ $O(N \log n)$



Move, predict, sense, update

SLAM with multiple poses

$$\begin{array}{ll} k=1 & \text{Move, Predict, Sense} \\ k=2 & \text{Move, Predict, Sense} \\ & \quad\quad \cdots \\ k=m & \text{Update} \end{array}$$

- Extract features using a window of m poses
- Trade off feature uncertainty for scan density

Using multi-pose data

- Now we've got pose error to deal with
- Add the "pose history" to the state:

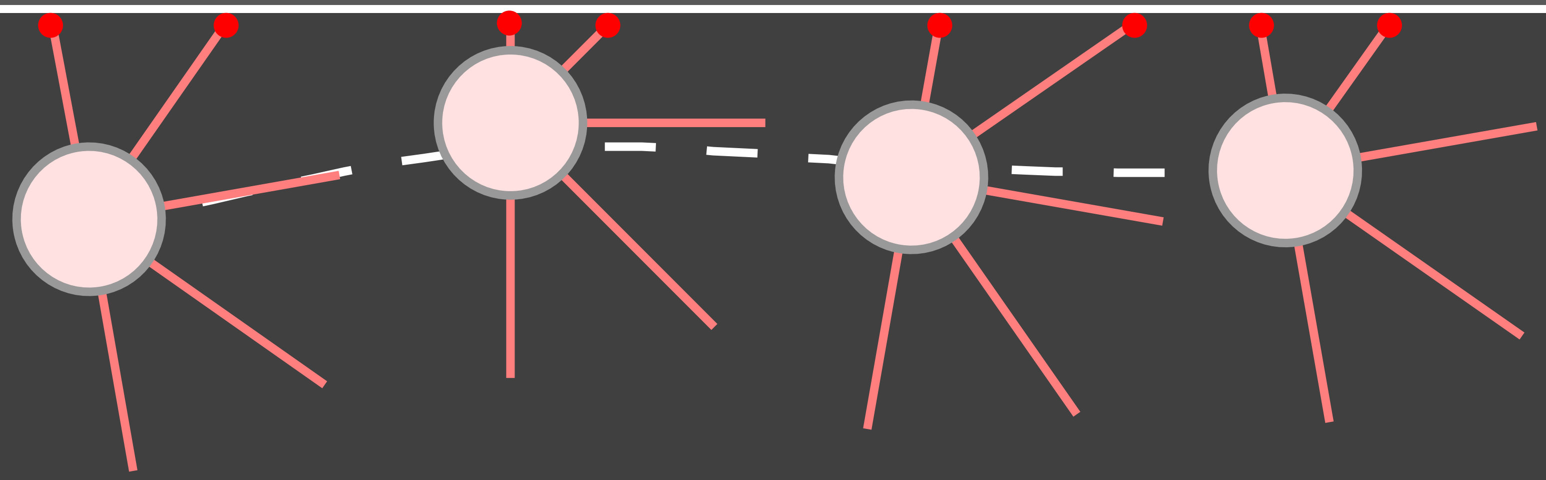
$$\boxed{x_r(k) = \left[x_{t_k}\ x_{t_{k-1}}\ \ldots\ x_{t_{k-m+1}}\right]}$$

Leonard, *et al.* 2002

- Too expensive! $\left(\int dx_{k-1}\right)$

SLAM with *multiscans*

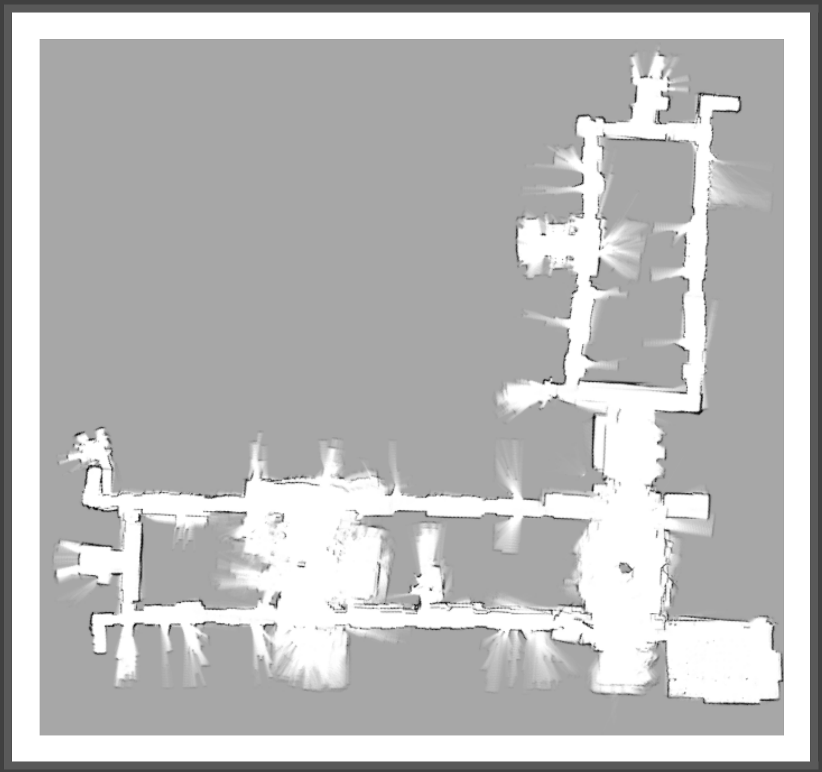
$$\mathbf{z}(k) = [z(k)\ z(k-1)\ \ldots\ z(k-m+1)]$$

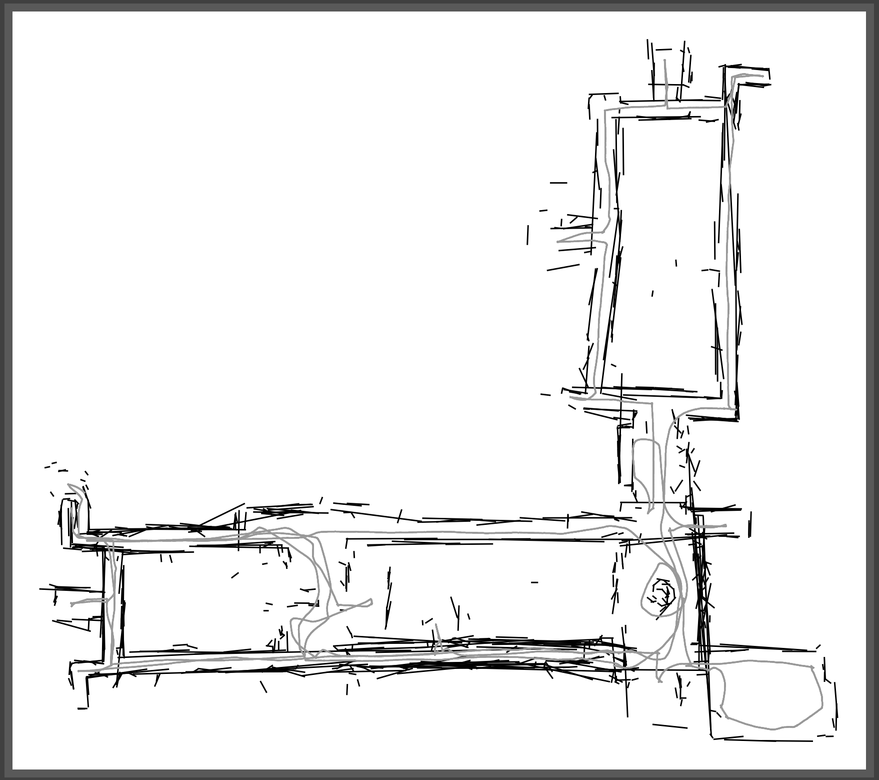


SLAM with *multiscans*

- Treat pose error as measurement error
- Use *expected* pose history for multiscan
- SLAM update every m steps

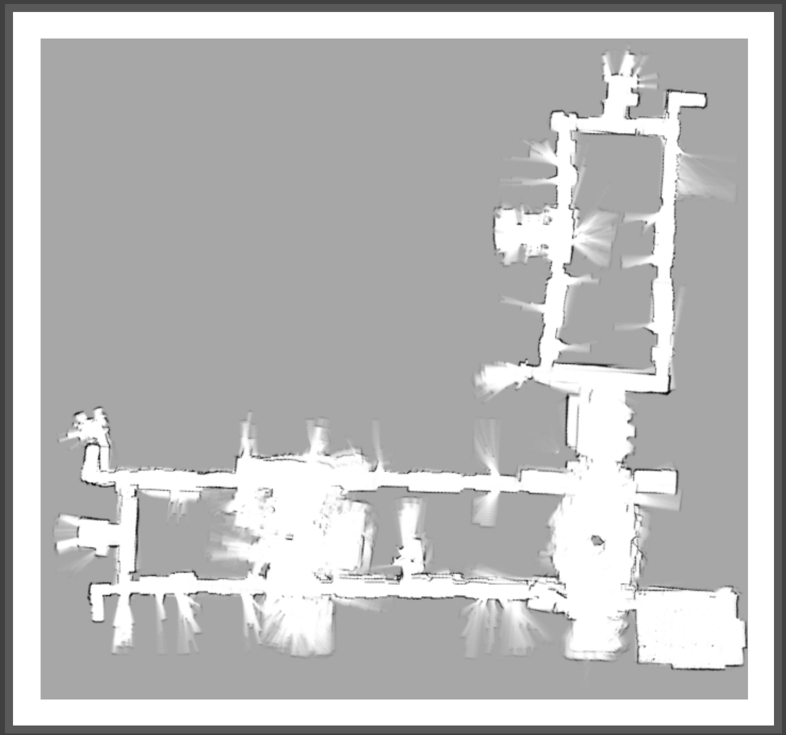
$$\boxed{\text{Particle filter: } O(N \log n)}$$

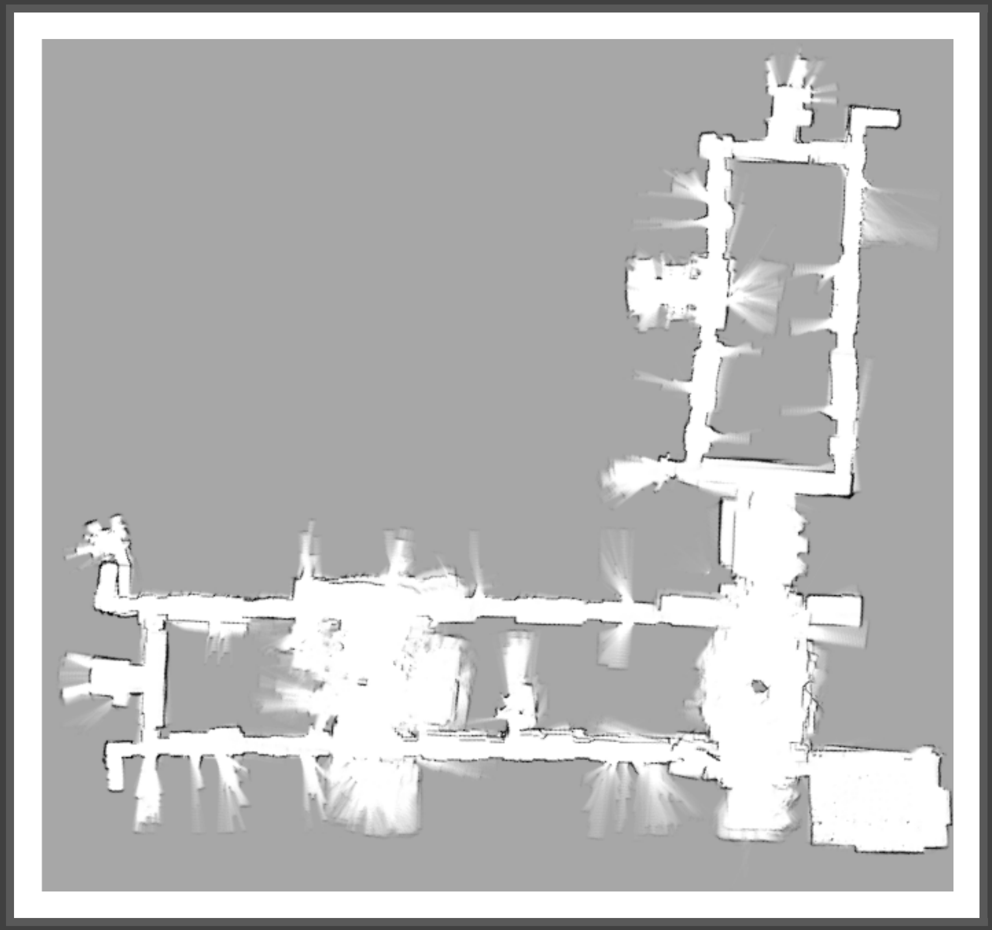


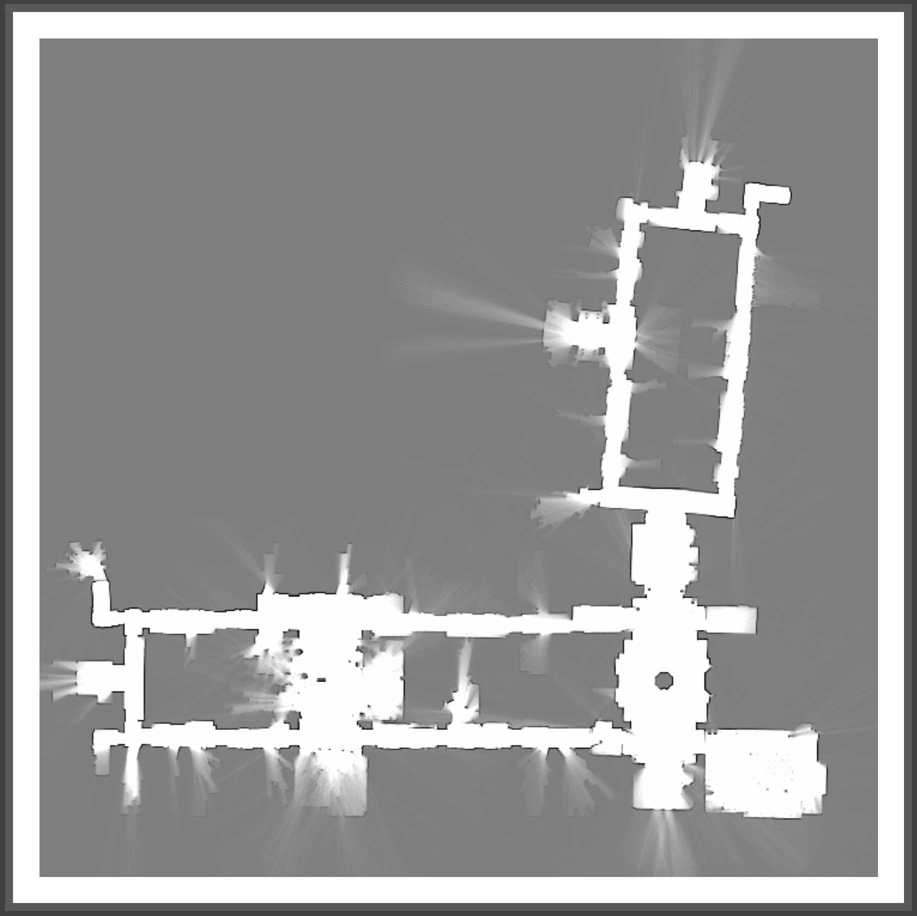


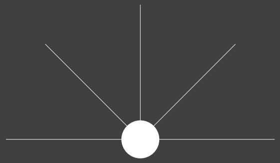
Data from RADISH courtesy B. Gerkey

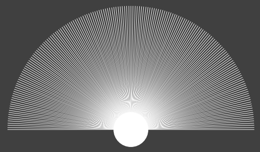
Dimensions	64m $\times$ 56m
Trajectory length	517m
Trajectory rotation	495 rad
Landmarks	750





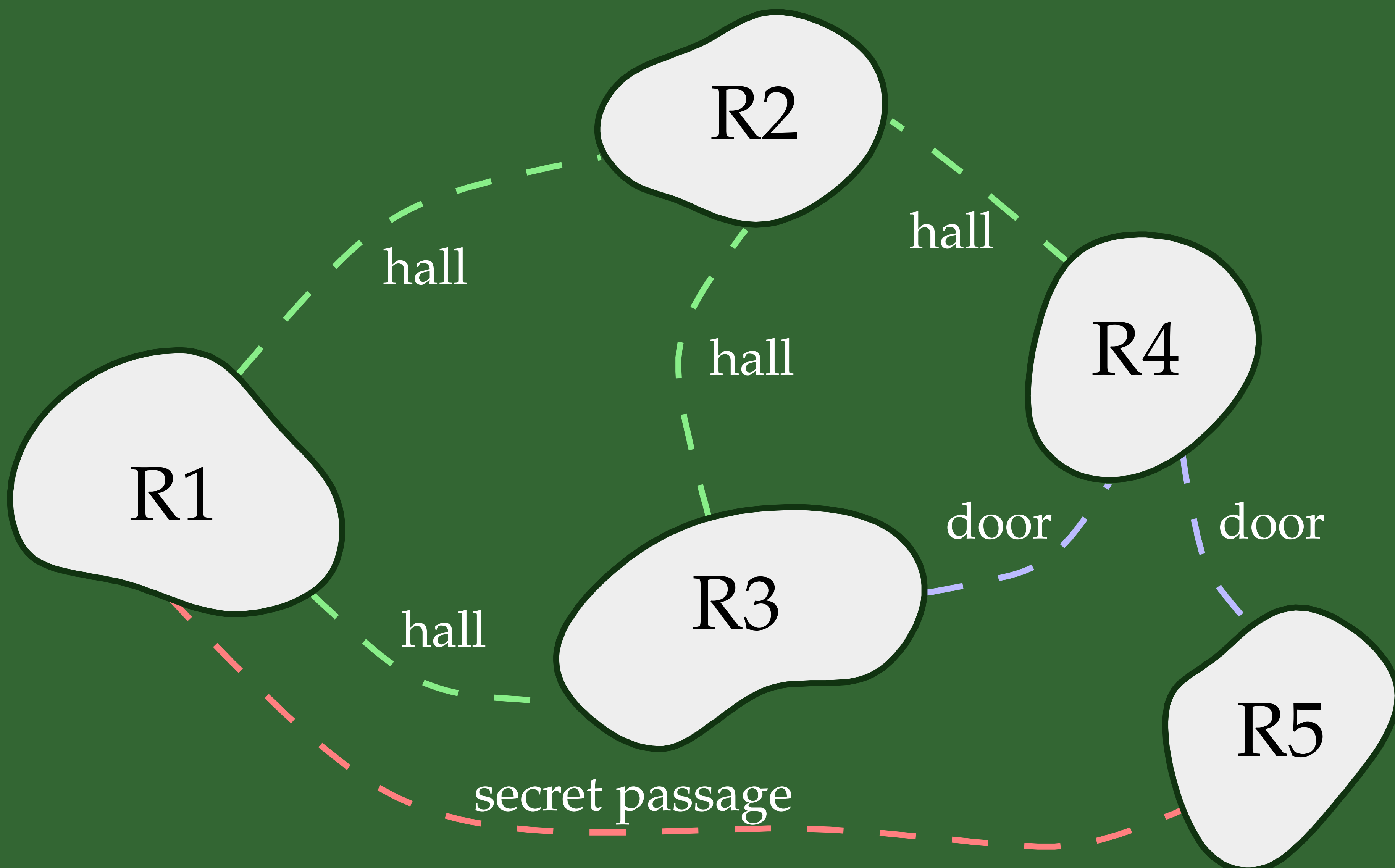






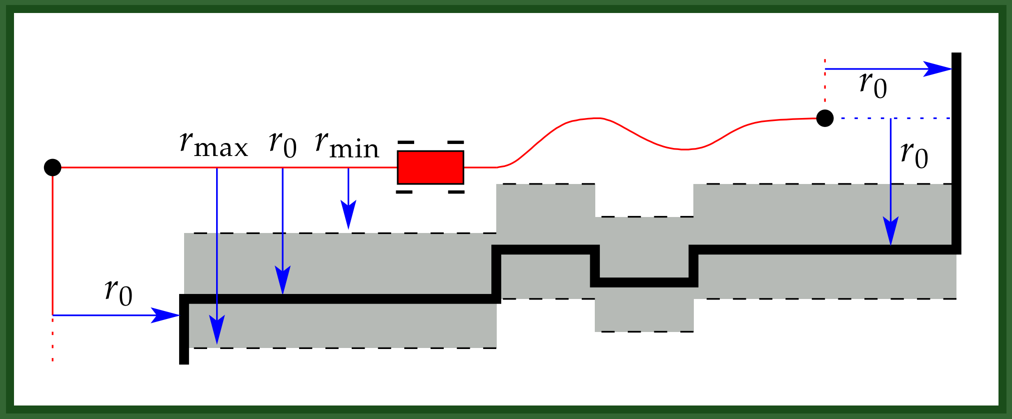
Topological approaches

some highlights

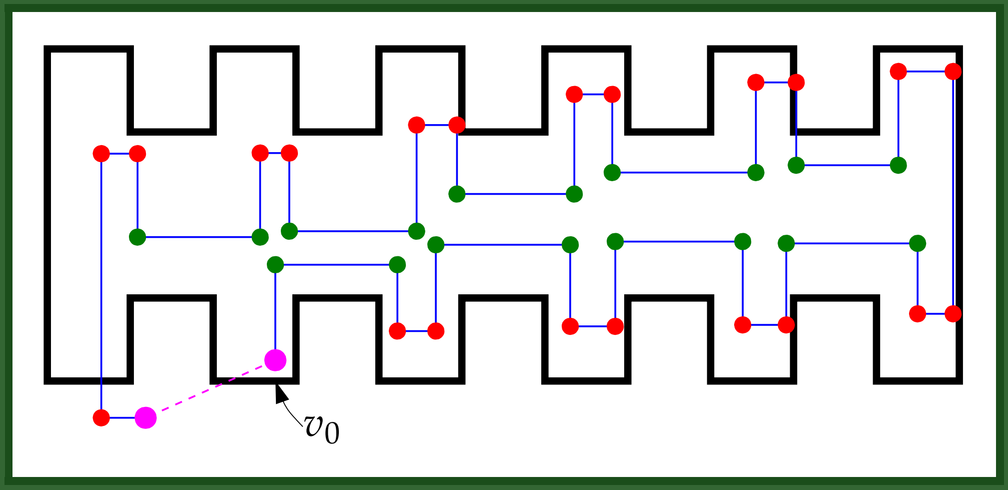

R2
hall
hall
hall
R4
R1
door
door
R3
hall
R5
secret passage

A simple strategy (WAFR 2004)

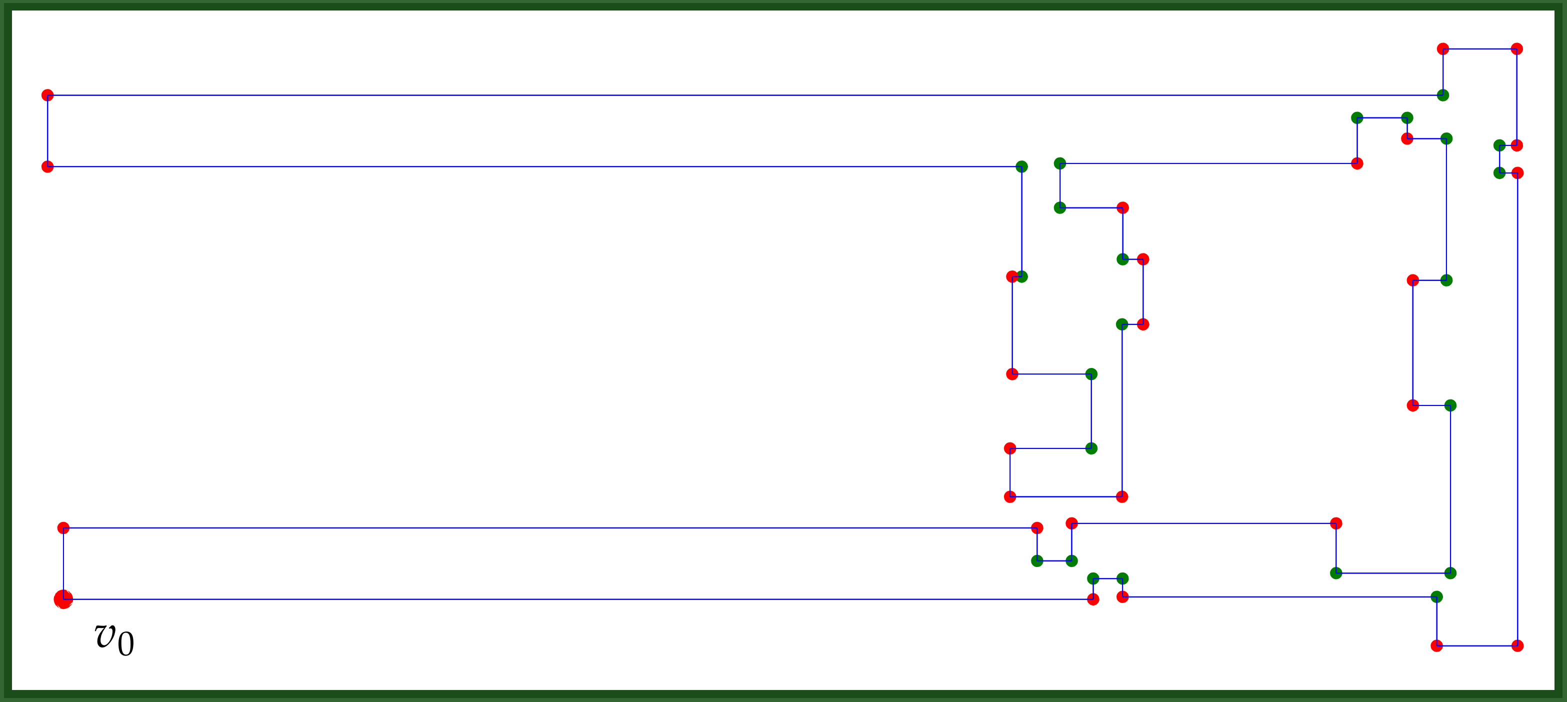
- Rectilinear environments; sensing: 
- Nodes: interior/exterior corners
- Edges: wall-following, hall-following



Closing loops (ICRA 2005)

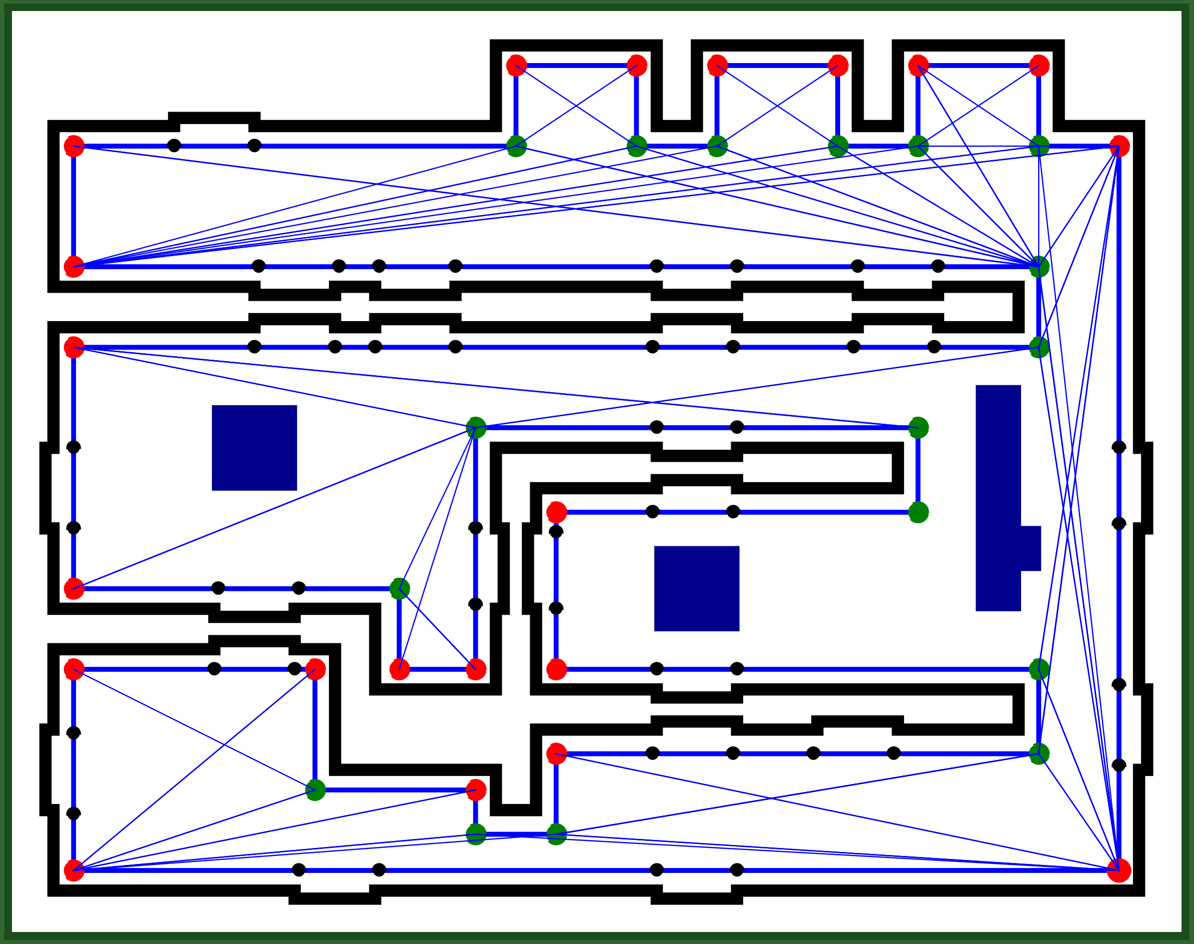


- Revisitation problem
- Most corners “look” identical
- Hardest in self-similar environments



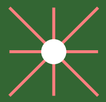
Amos Eaton Bldg., RPI

(30 m $\times$ 12 m)

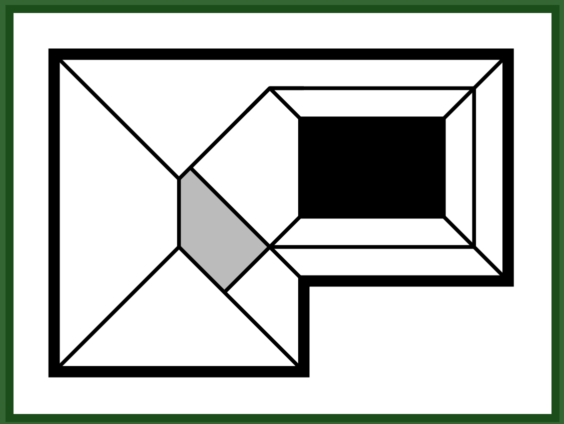


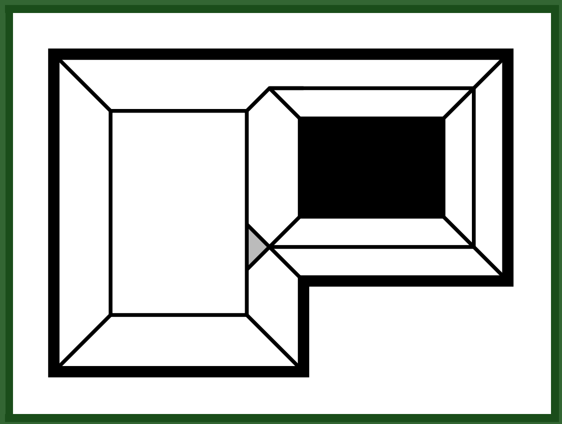
Simulation

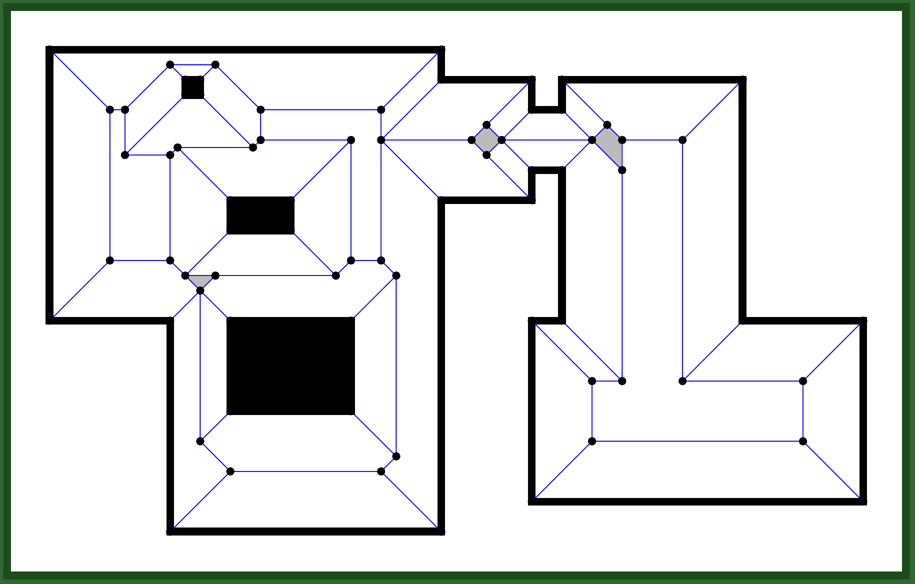
Tracing the SGVD$_\infty$ (TR 2005)

- Rectilinear environment; sensing: 
- Saturated Generalized Voronoi Diagram, L_∞:

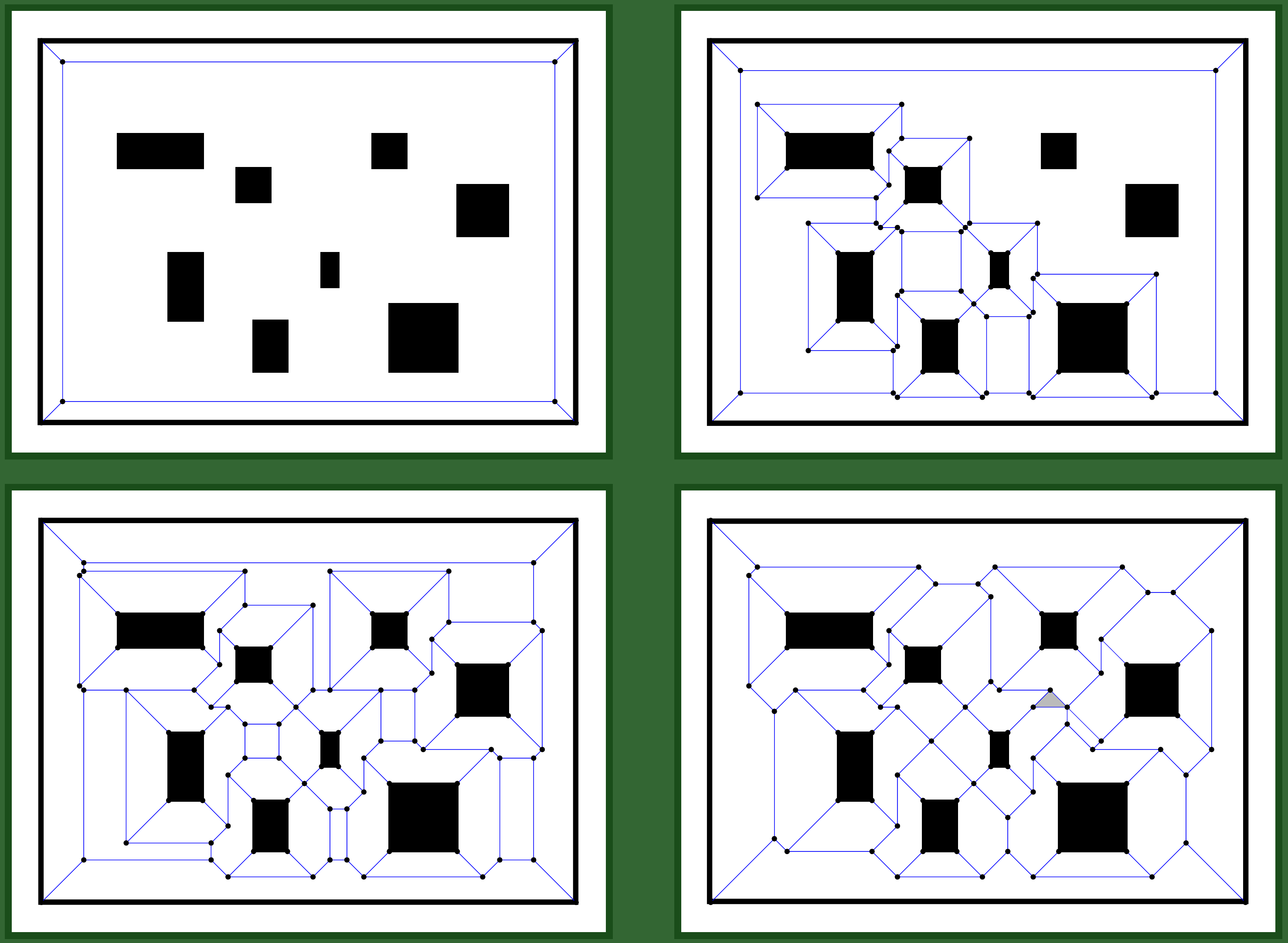
$$\boxed{d_\infty(\mathbf{p}, \mathbf{q}) = \max_i |p_i - q_i|}$$







Complete algorithm for tracing $SGVD_\infty$ from any starting point



Recap

- Mapping with limited sensing is possible
- SLAM with sparse sensing
- Topological mapping with limited sensing

Thank you!