Carnegie Mellon
THE ROBOTICS INSTITUTE

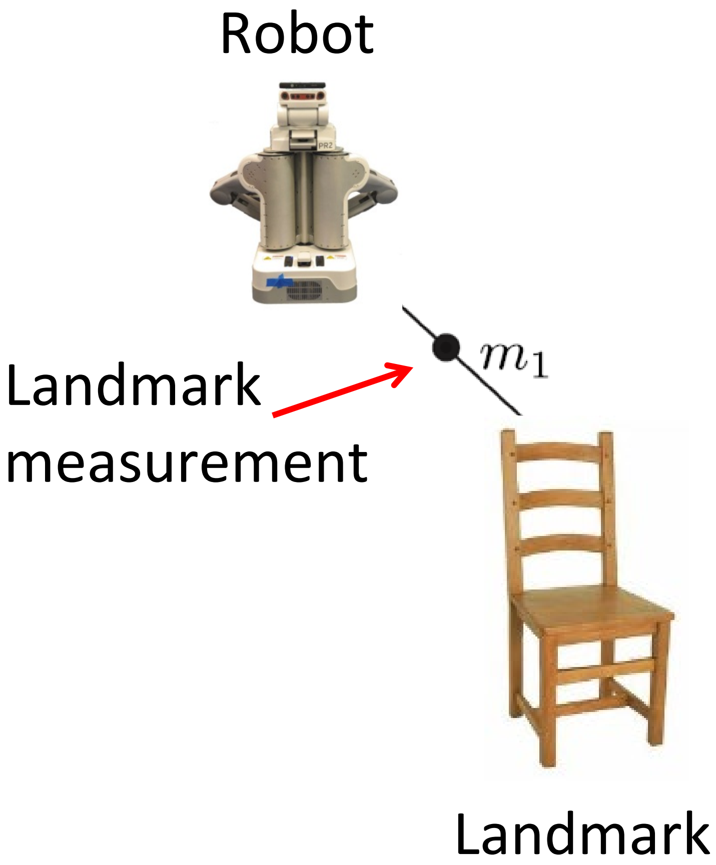
SLAM: Least Squares

Robot Localization and Mapping
16-833

Michael Kaess

October 2, 2024

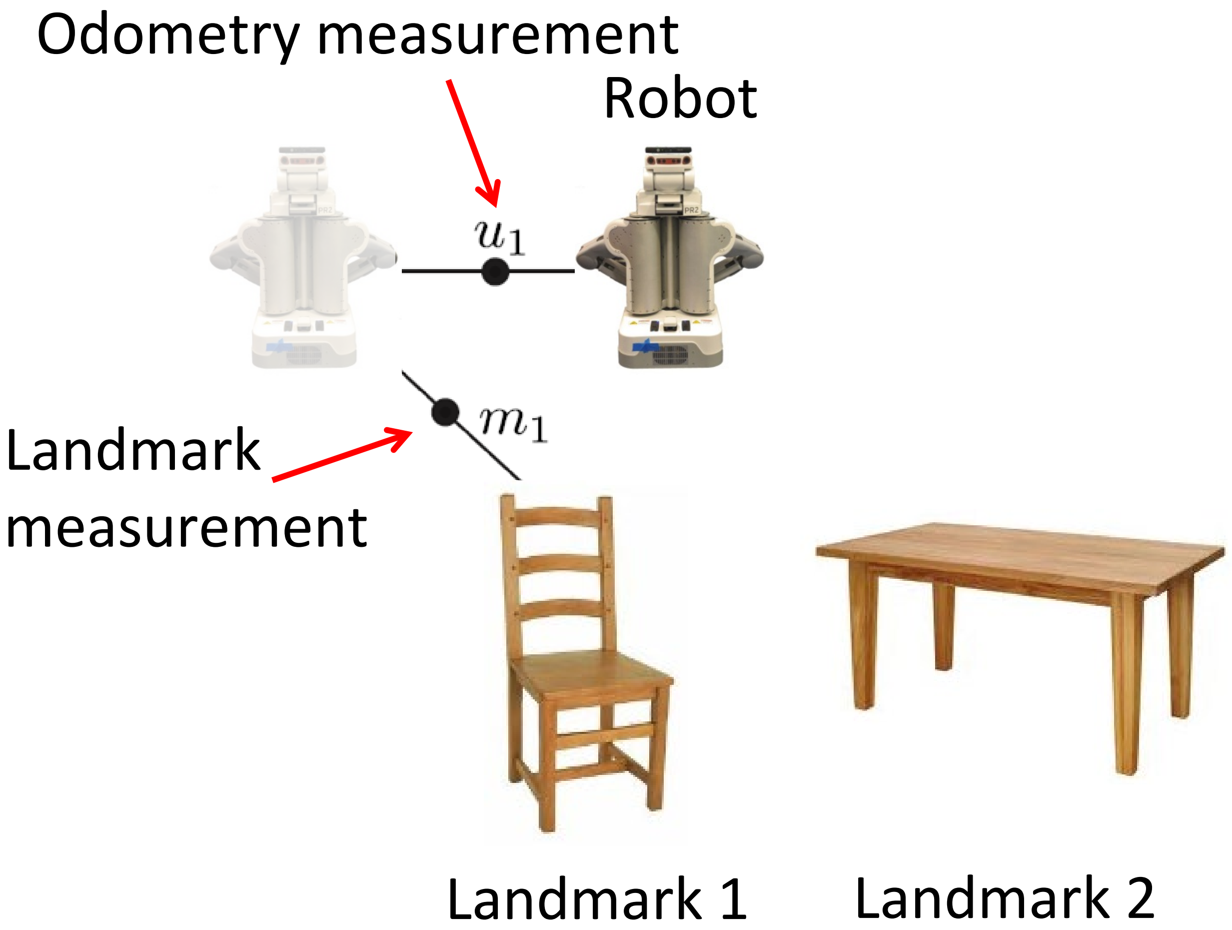
The SLAM Problem (t=0)



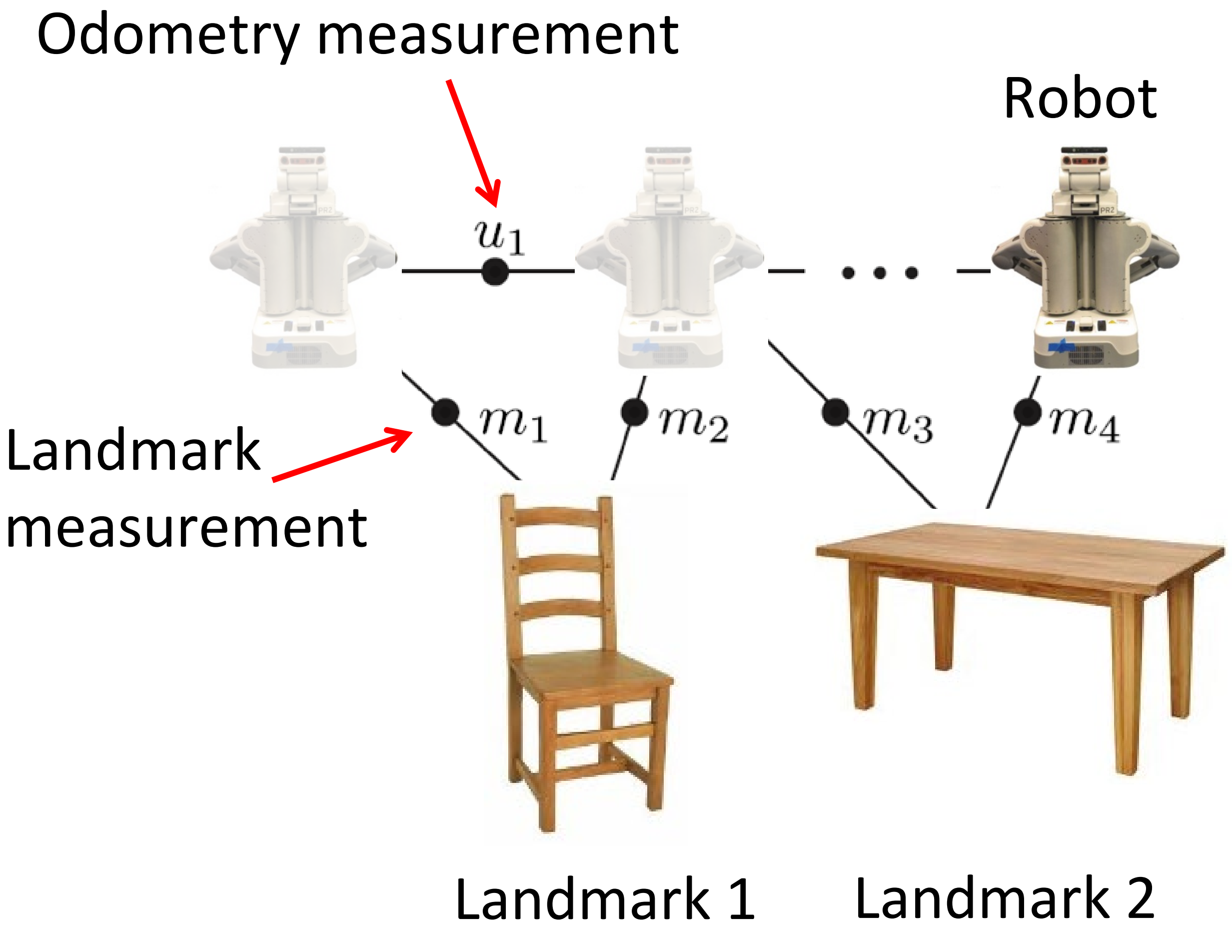
Onboard sensors:

- Wheel odometry
- Inertial measurement unit (gyro, accelerometer)
- Sonar
- Laser range finder
- Camera
- RGB-D sensors

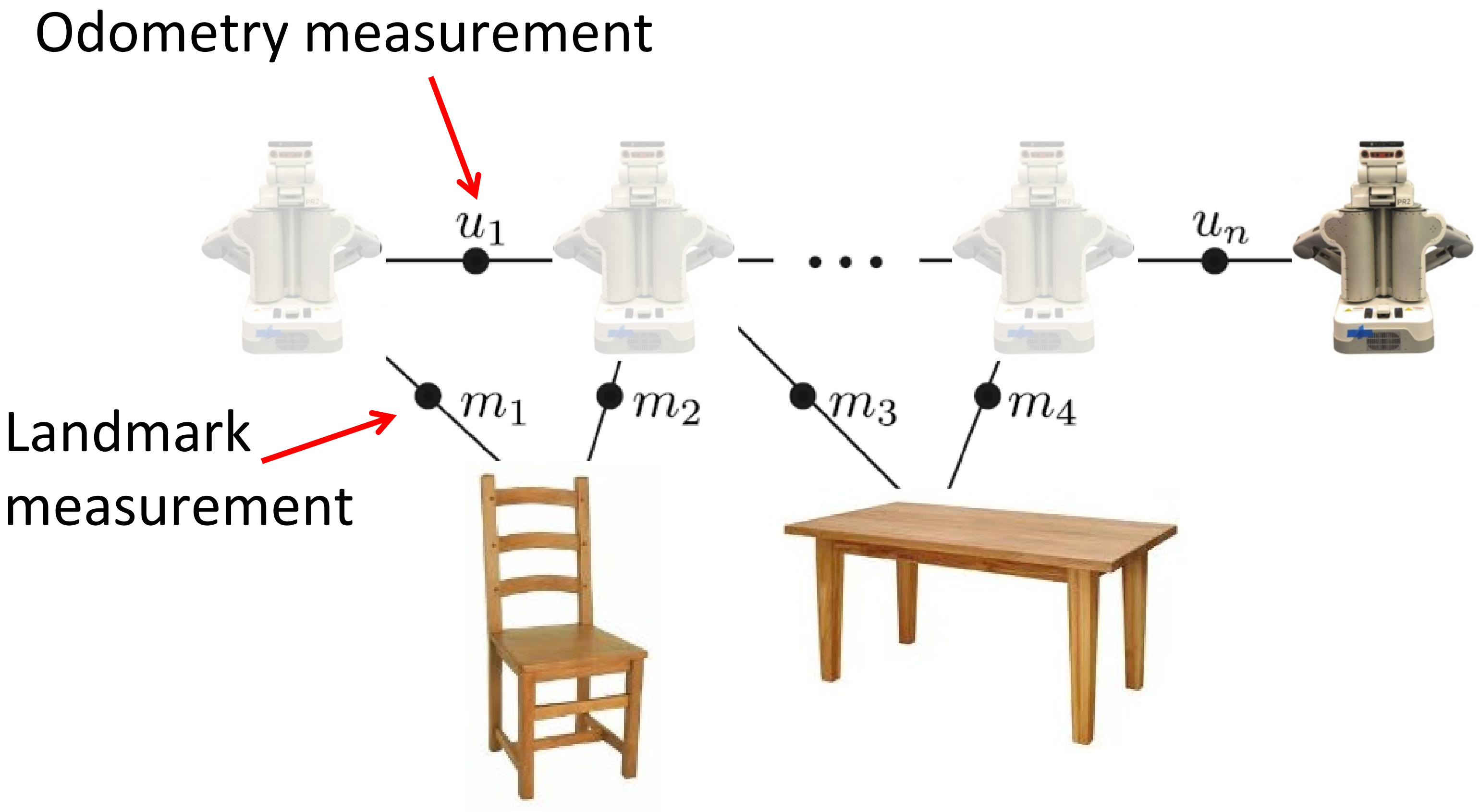
The SLAM Problem (t=1)



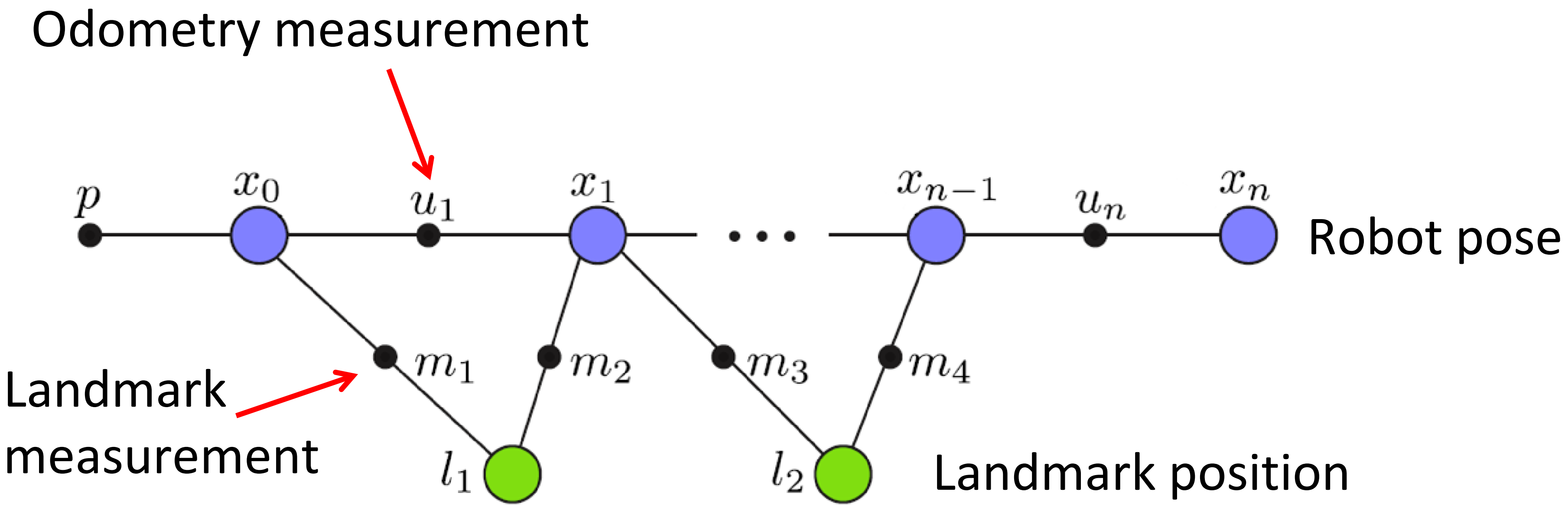
The SLAM Problem (t=n-1)



The SLAM Problem (t=n)



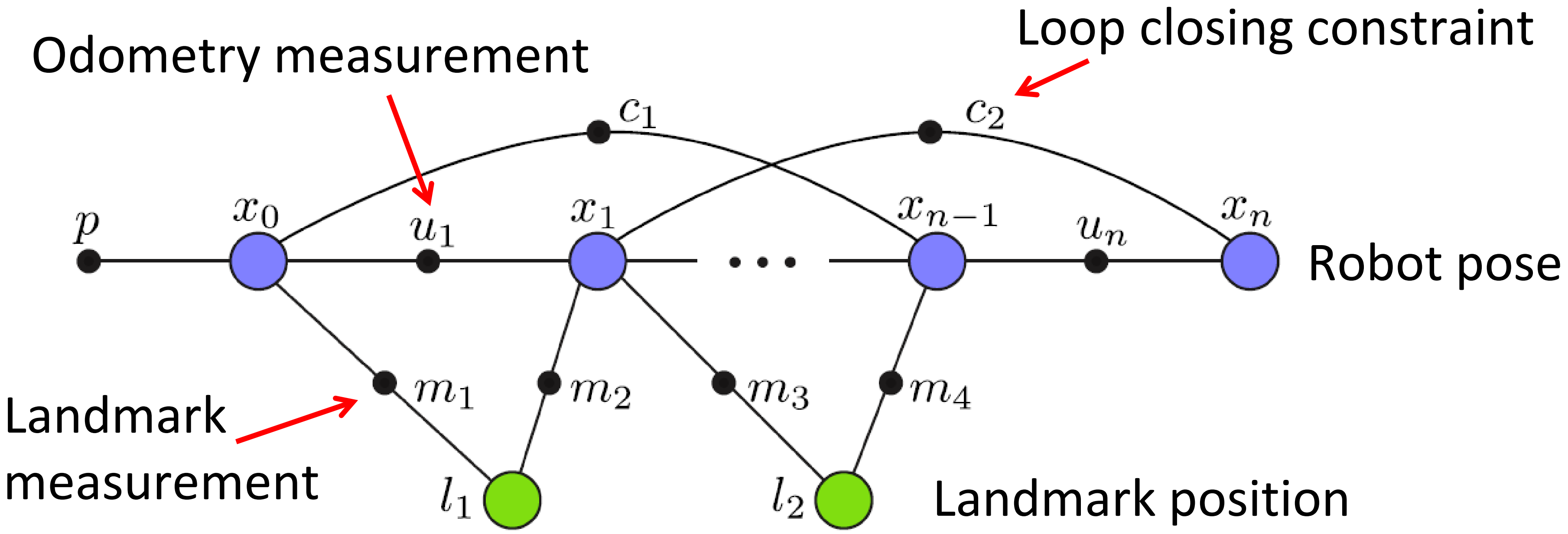
Factor Graph Representation of SLAM



Bipartite graph with ***variable nodes*** and ***factor nodes***

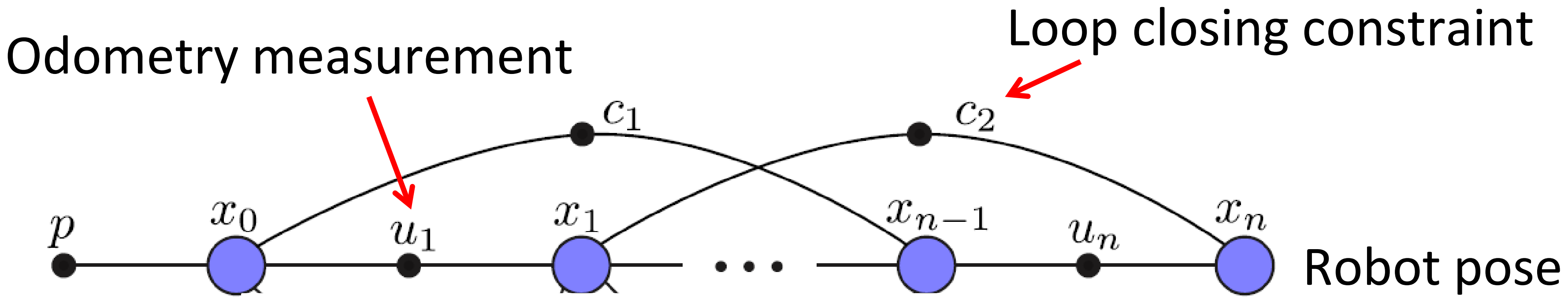
Carnegie Mellon
THE ROBOTICS INSTITUTE

Factor Graph Representation of SLAM



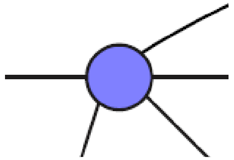
Bipartite graph with ***variable nodes*** and ***factor nodes***

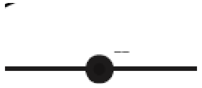
Factor Graph Representation of SLAM



“Pose graph” (no explicit modeling of landmarks)

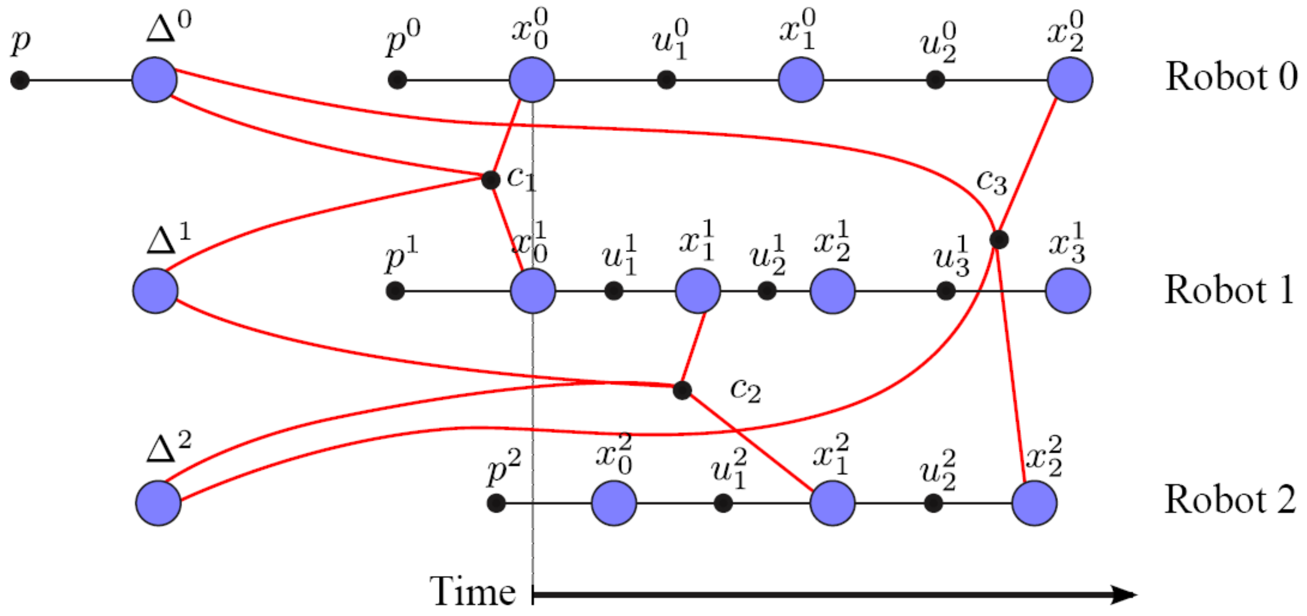
Bipartite graph with ***variable nodes*** and ***factor nodes***





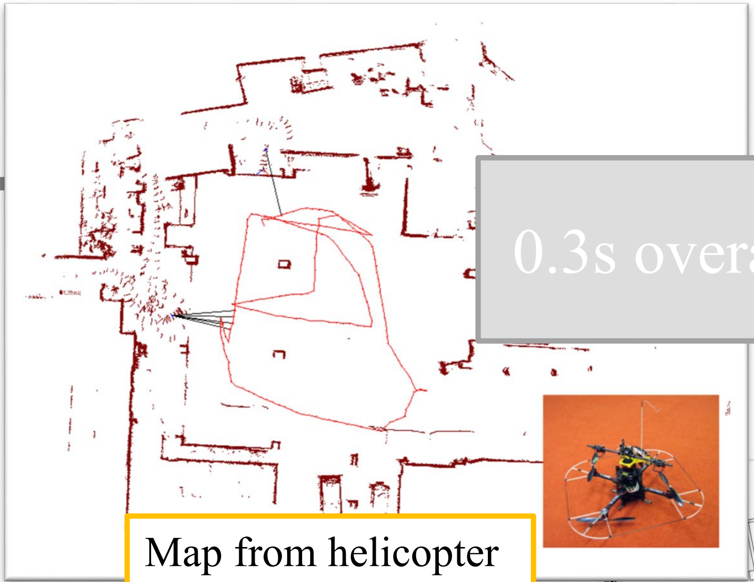
Factor Graph: Advanced Example

- Anchor nodes, Kim et al. ICRA 2010

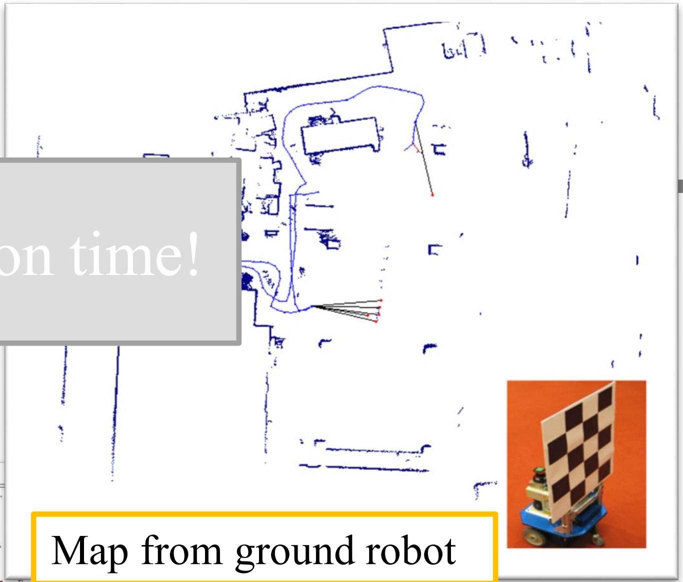


- Can also include calibration parameters
  - Camera intrinsics, sensor/vehicle alignment, wheel diameter…

0.3s overall optimization time!

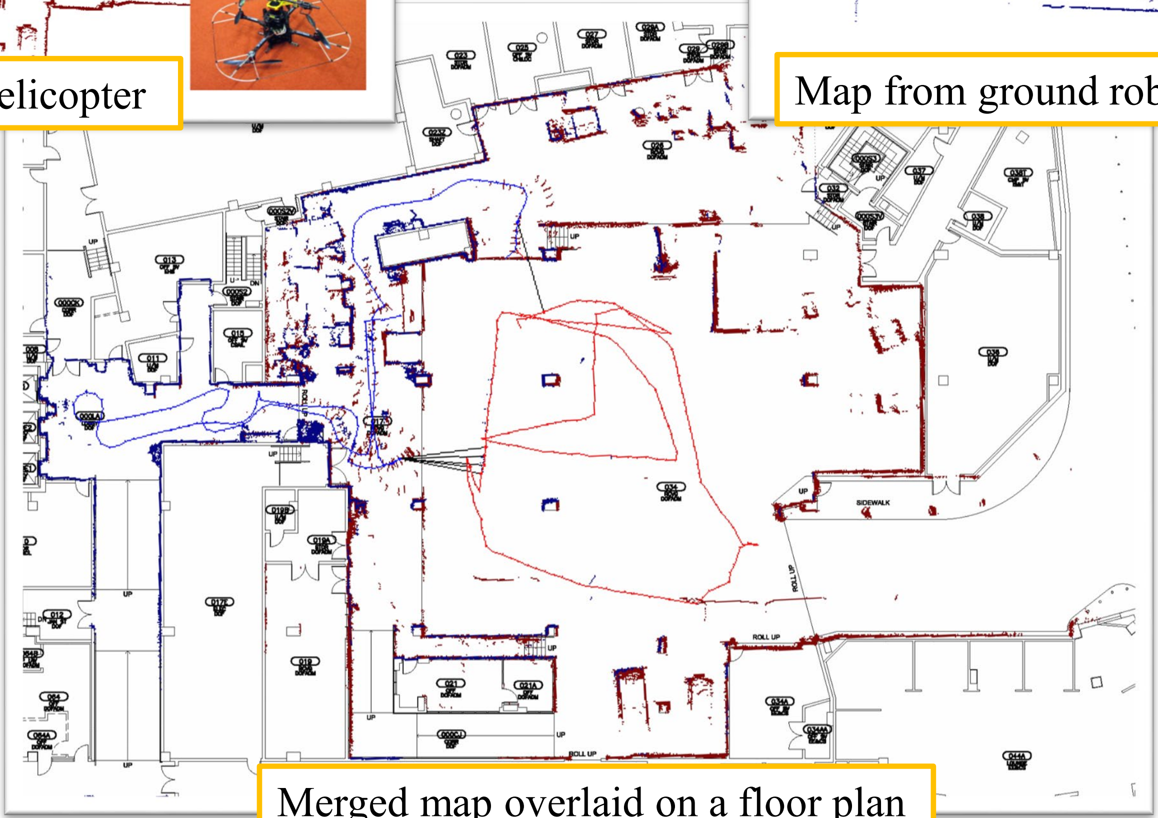


Map from helicopter



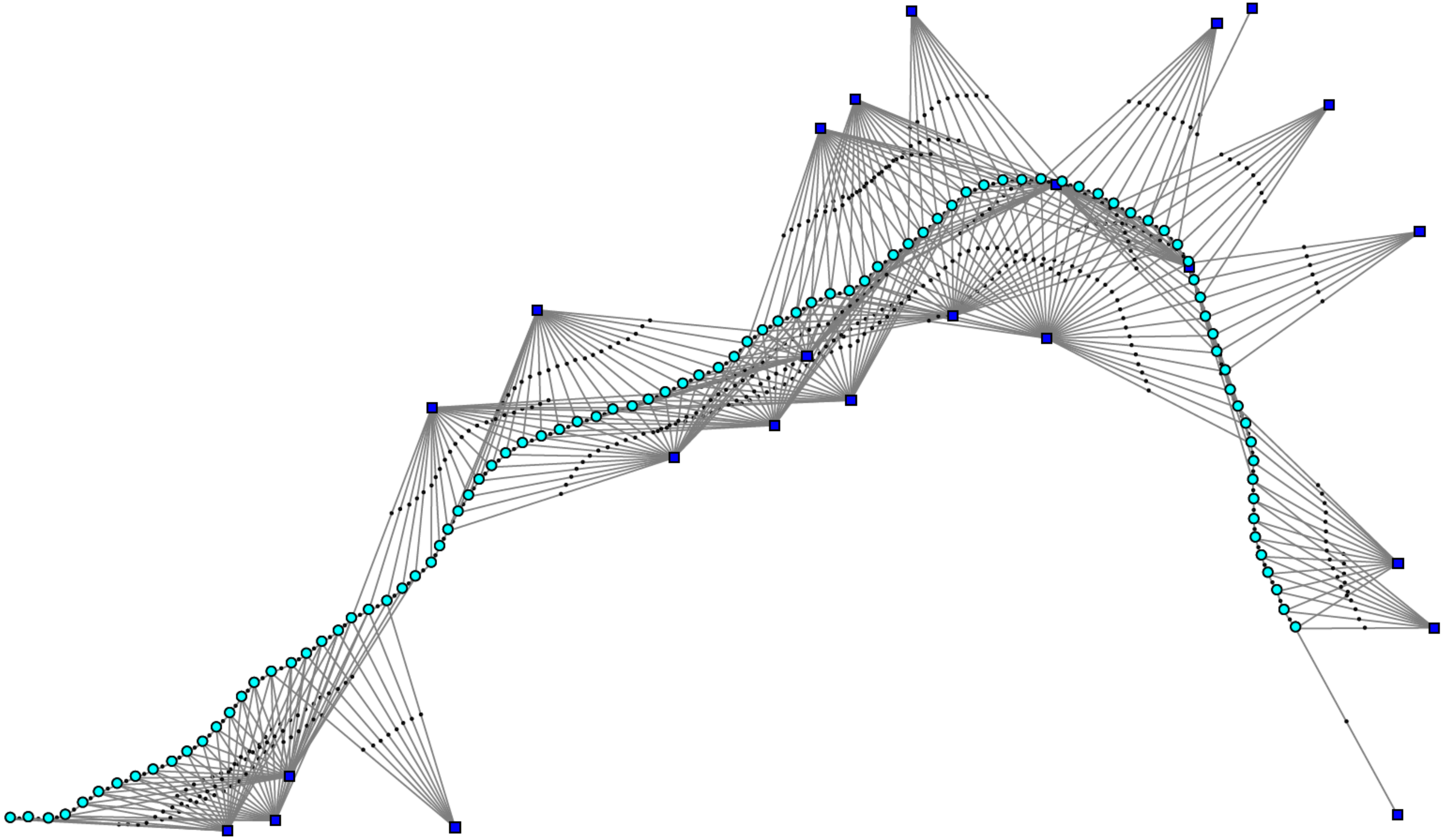
Map from ground robot

Anchor Nodes
Kim et al. ICRA 2010



Merged map overlaid on a floor plan

Larger Factor Graph SLAM Example

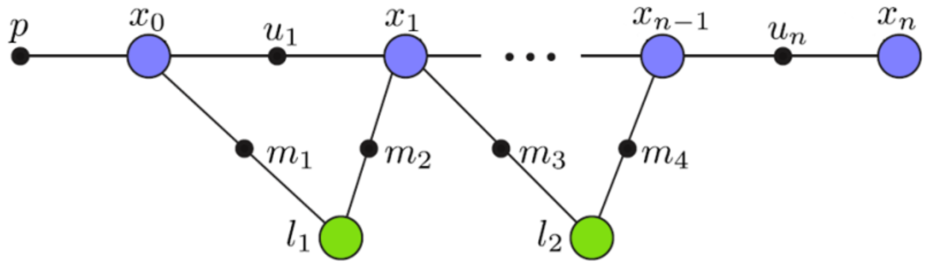


Variables and Measurements

- Variables:

$$\Theta = \{x_0, x_1 \cdots x_n, l_1, l_2\}$$

Might include other quantities such as lines, planes and calibration parameters



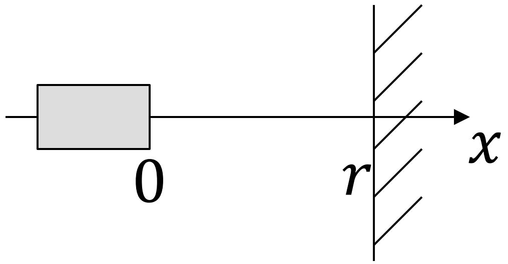
- Measurements:

$$Z = \{p, u_1 \cdots u_n, m_1 \cdots m_4\}$$

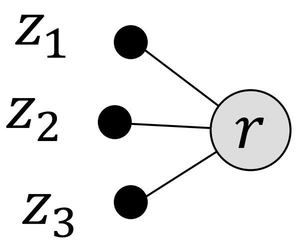
p is a prior to fix the gauge freedom (all other measurements are relative!)

Generative Sensor Model – 1D Example

- 1D world, laser range finder at x=0, wall at x=r



- Measurements: z_1, z_2, z_3
- Factor graph:



- Assumption: z_i are iid (independent and identically distributed) Gaussian random noise with mean r and covariance σ^2: $z \sim N(r, \sigma^2)$

Generative Sensor Model

What can we do with the generative sensor model?

$$\mathrm{z} = \mathrm{r} + \nu, \quad \nu \sim N(0, \sigma^2), \quad p(z|r)$$

- Simulate
  - Given the variable, we can draw samples from ν to simulate the measurement process
- Test
  - Given the variable and a measurement, evaluate its probability (density) under this model $p(z|r) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(z-r)^2}$
- Inference
  - Given the measurement, we can perform inference about the variable (typically from multiple measurements)

Finding the Best Solution

Our goal is to find the Θ that maximizes $p(\Theta|Z)$

Bayes Rule

Our goal is to find the Θ that maximizes $p(\Theta|Z)$

$$\underbrace{p(\Theta|Z)}_{\text{Posterior}} = \frac{\overbrace{p(Z|\Theta)}^{\text{Likelihood}}\ \overbrace{p(\Theta)}^{\text{Prior}}}{\underbrace{p(Z)}_{\text{Evidence}}}$$

Note:

- While the measurements Z are given, the generative sensor models provide us with likelihood functions $\mathrm{L}(\Theta; z_i) \propto p(z_i|\Theta)$
- Evidence is independent of Θ

Maximum Likelihood and Maximum A Posteriori

- Maximum A Posteriori (MAP)

$$\Theta_{MAP} = \operatorname{argmax}_{\Theta}\ p(Z|\Theta)\, p(\Theta)$$

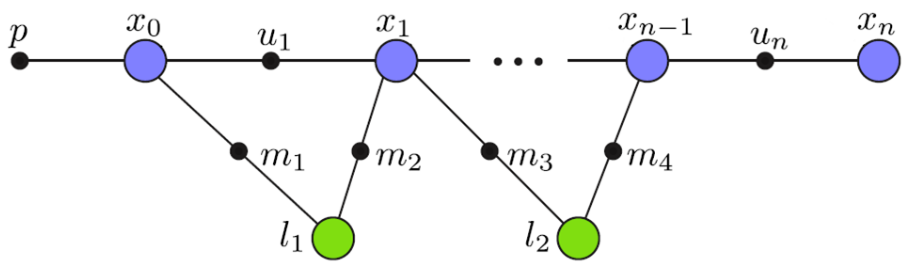
- Maximum Likelihood Estimator (MLE)

$$\Theta_{MLE} = \operatorname{argmax}_{\Theta}\ L(\Theta; Z)$$

Factorization of Probability Density

- Conditional independence:

$$p(z_1 z_2|\Theta) = p(z_1|\Theta)\, p(z_2|\Theta)$$



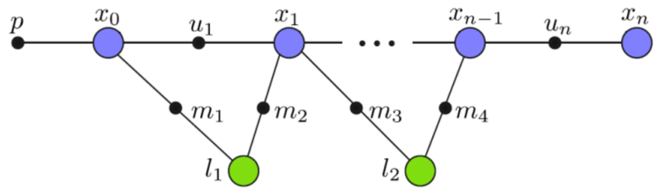
$$\Leftrightarrow \quad \operatorname{argmax}_{\Theta} \prod_{z \in Z} p(z|\Theta)$$

$$\operatorname{argmax}_{\Theta}\; p(p|\Theta)\, p(u_1|\Theta) \cdots p(u_n|\Theta)\, p(m_1|\Theta) \cdots p(m_4|\Theta)$$

SLAM as a Least-Squares Problem

- On the board:
  - Log monotonic
  - Linear case
  - Normal equations
  - Solving with pseudo inverse

SLAM as a Least-Squares Problem



$$\mathrm{argmax}_\Theta \prod_{z\in Z} p\left(z|\Theta\right)$$

Gaussian noise

$$\mathrm{argmin}_\Theta \sum_i \|h_i(\Theta) - z_i\|_\Sigma^2$$

$h()$ linear

$$\mathrm{argmin}_x \|Ax - b\|^2$$

Normal equations:

$$A^T A x = A^T b$$

Summary

- Instead of filtering, we can formulate the SLAM problem as a batch least-squares problem
- Normal density --log(monotonic)--> Mahalanobis distance
- Normal equations

- Factor Graphs for Robot Perception: 2.1-2.4

- Next: Exploiting Sparsity
  Factor Graphs for Robot Perception: 3