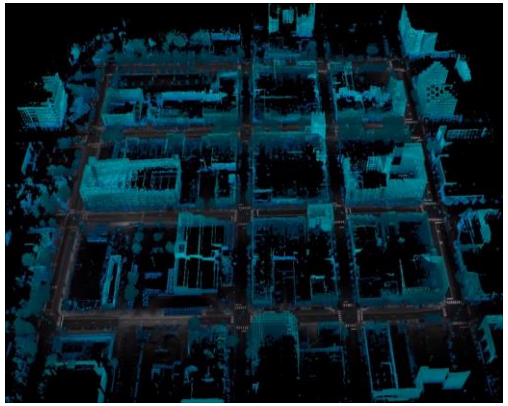
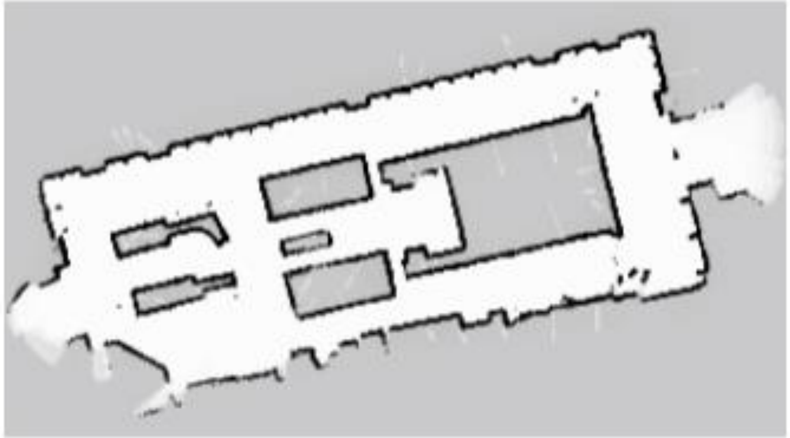
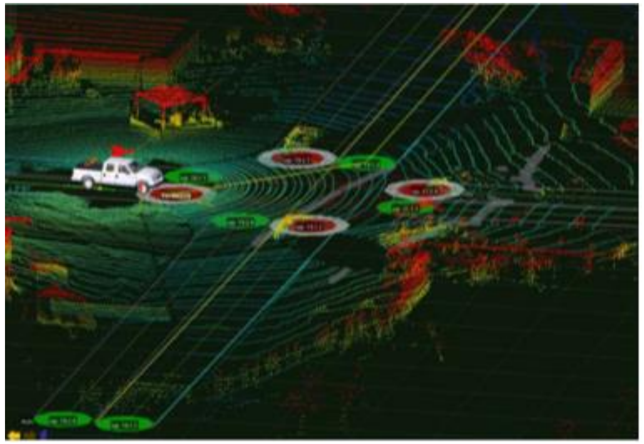
BYU Electrical & Computer Engineering





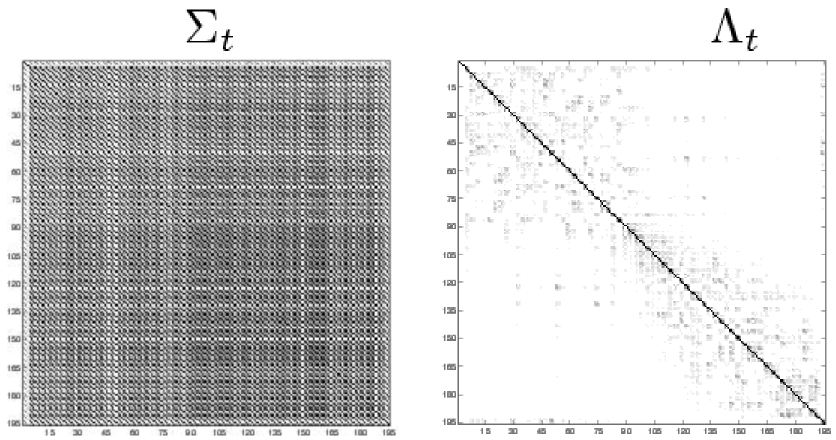
INTRO TO FACTOR GRAPH

ECEN 633: Robotic Localization and Mapping

Some slides courtesy of Ryan Eustice and Michael Kaess

Today's Topic

- Least Squares (See Notes)
- Pose-Graph SLAM
- Factor Graphs
- Factor Graphs and Non-linear Least Squares



Announcements

- Midterm Starts Friday
- Project Proposal also Due Friday

Midterm

- Runs Friday to Wednesday

This exam is open book, meaning you can use the following ONLY:

- Your own notes
- The class textbooks
- The class notes
- The lecture slides
- The minisets/solutions
- Python/MATLAB/Calculators for evaluating individual equations

If you do use python/MATLAB/Calculators, **writeout the equations** on the test as if you were doing it by hand. **I won't grade your code and you won't get credit for just having the answer on the paper.**

You may **NOT** use the internet (except to lookup python/MATLAB syntax).

You have **24 hours** to take this exam once you begin looking at this page.

Please have integrity and do not:

- Spend more time on the exam then the allotted time
- Share your exam with anyone else (including future students who take the course) before, during, or after taking it.
- Use any resources other than those described above.

Write your name and byu id at the top of every page.

Project Proposal

- Also Due Friday
- Groups of 3

Purpose: The purpose of your project proposal is to outline the project you plan to carry out so that I can help you determine if the scope of the project is reasonable for the course and the time.

Your proposal will **not** count towards your grade, but approval of it is required before you may move forward with your project.

Deliverable: You will write a 1-2 page extended abstract in the format of an IEEE RAS conference publication. This abstract must be submitted to Learning Suite before the deadline listed there.

After submitting the abstract, we will schedule a time during office hours to discuss the your proposed project and make sure it fits within the scope and timescale of the course.

Instructions:

You need to select a project that is closely related to the course; specifically, your project must either:
1. Build on and extend existing robotics research relating to mobile robotic perception, localization, mapping, or SLAM, or
2. Implement an existing paper or system that builds on what we have covered in the course and evaluates it on real-world data.

Your abstract (and final paper) should include the following sections:

Introduction - This section should summarize the problem you are proposing a solution to, emphasize why that problem is important, and succinctly sell why your method is awesome. This section should

Three Main SLAM Paradigms

Kalman filter

Particle filter

Graph-based

Least Squares Approach to SLAM

Courtesy: C. Stachniss

Least Squares in General

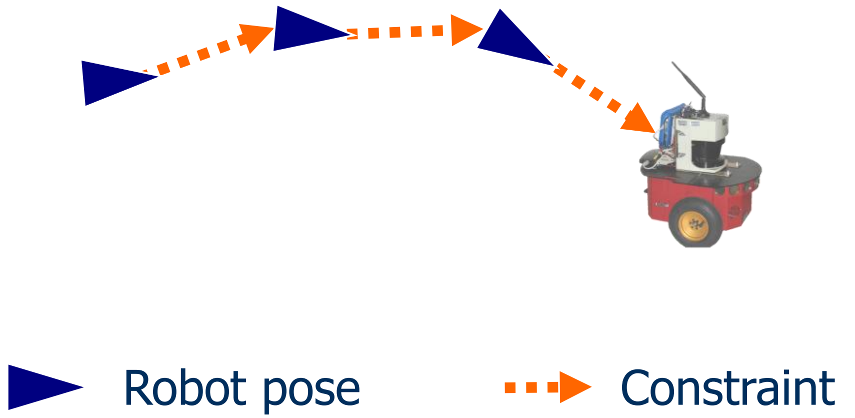
- Approach for computing a solution for an **overdetermined system**
- "More equations than unknowns"
- Minimizes the **sum of the squared errors** in the equations
- Standard approach to a large set of problems

Today and Most of the Rest of the Semester: Application to SLAM

Courtesy: C. Stachniss

Pose-Graph SLAM

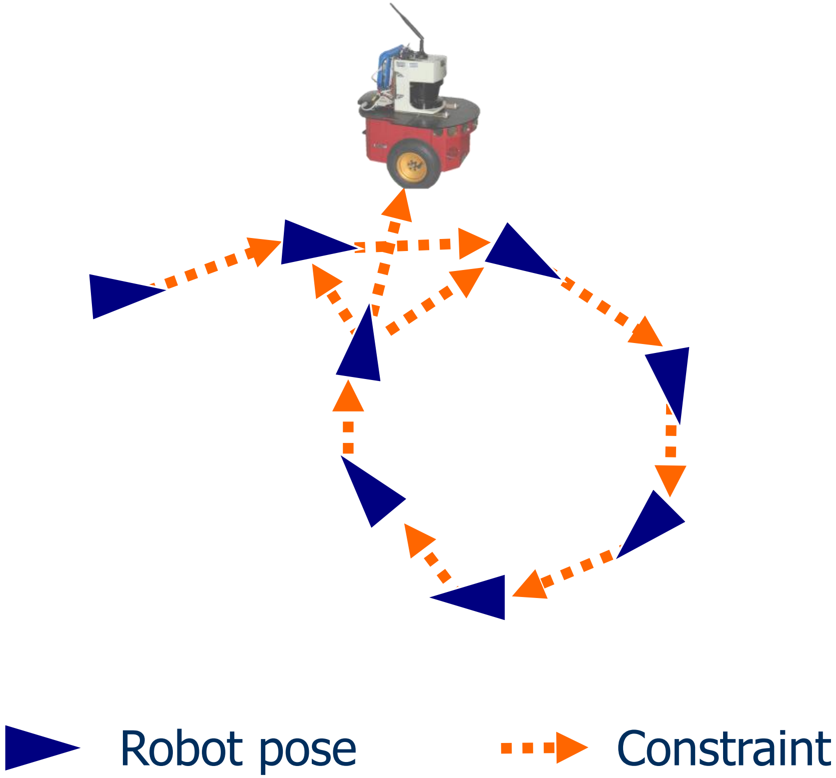
- Constraints connect the poses of the robot while it is moving
- Constraints are inherently uncertain



Courtesy: C. Stachniss

Pose-Graph SLAM

- Observing previously seen areas generates constraints between non-successive poses



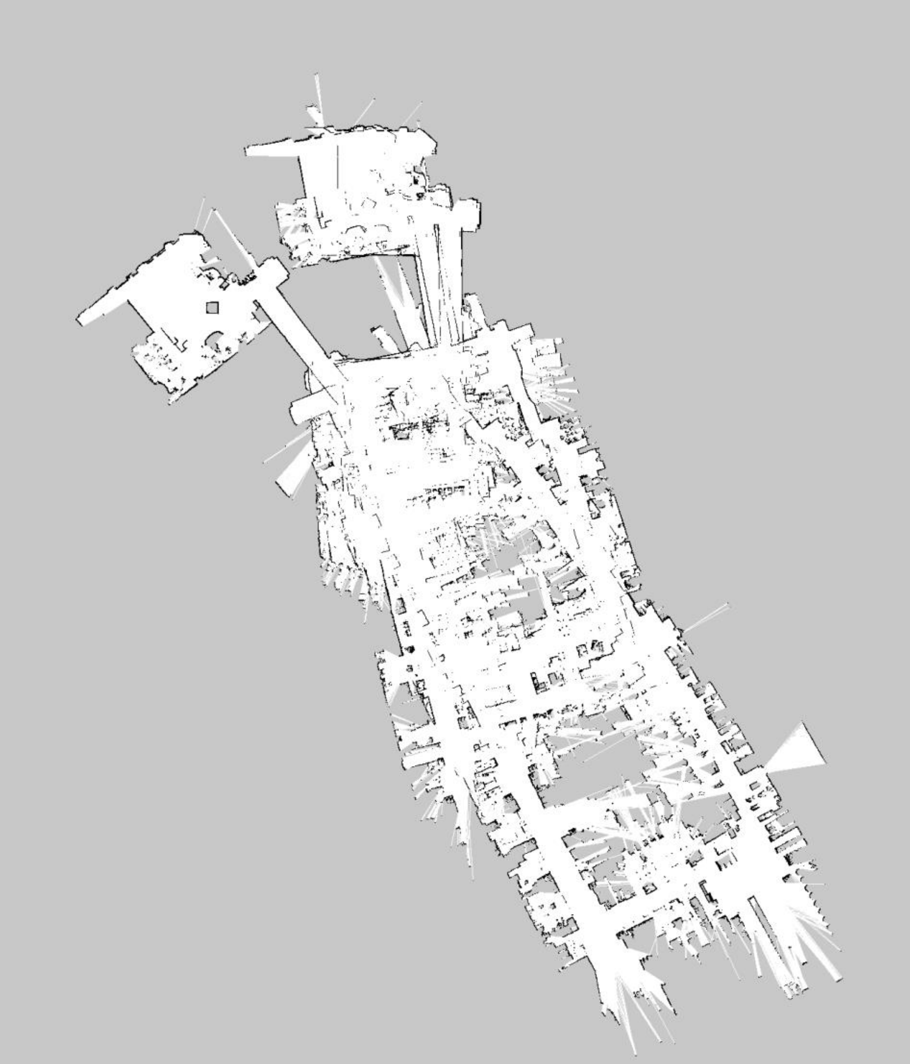
Courtesy: C. Stachniss

Idea of Pose-Graph SLAM

- Use a **graph** to represent the problem
- Every **variable node** in the graph corresponds to a pose of the robot during mapping
- Every **factor-node** between two variable nodes corresponds to a spatial constraint between them
- **Pose-Graph SLAM:** Build the graph and find a node configuration that minimizes the error introduced by the constraints (or factors)

Pose-Graph SLAM in a Nutshell

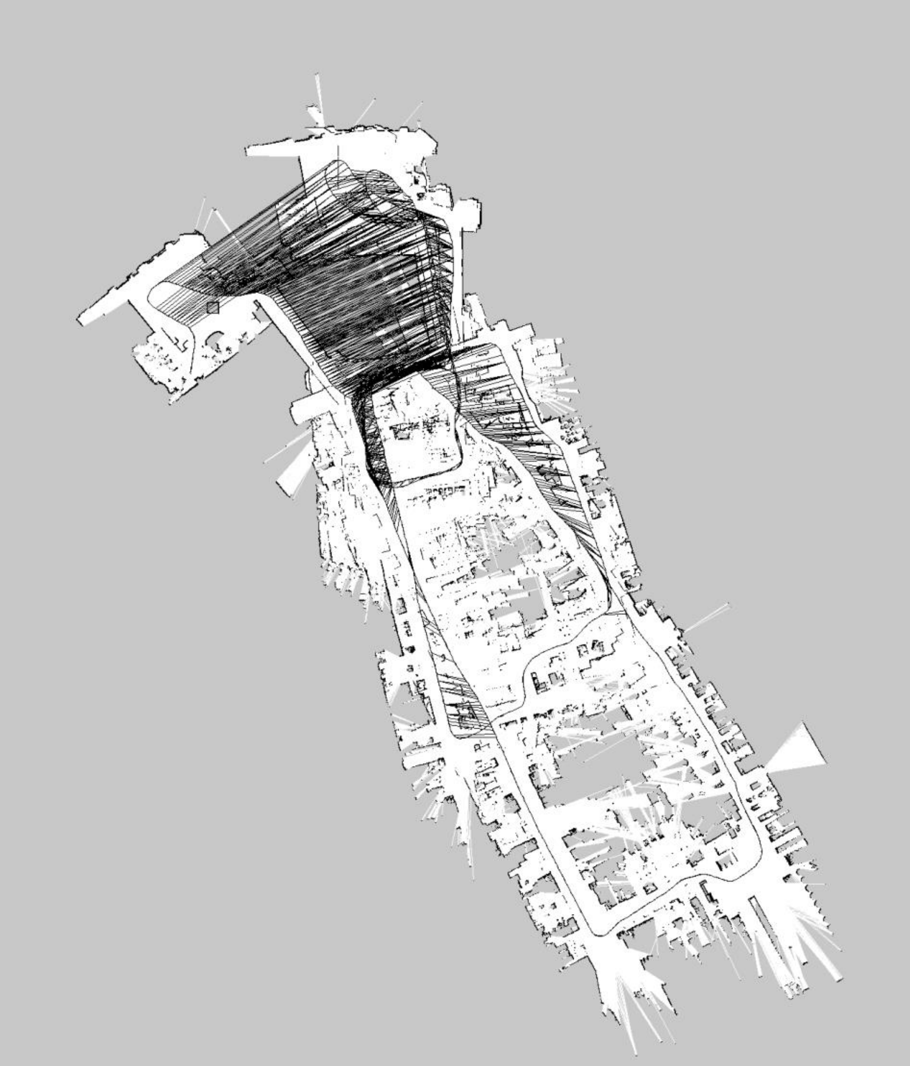
- Every variable node in the graph corresponds to a robot position and a laser measurement
- An edge (or factor-node) between two nodes represents a spatial constraint between the nodes



KUKA Halle 22, courtesy of P. Pfaff

Pose-Graph SLAM in a Nutshell

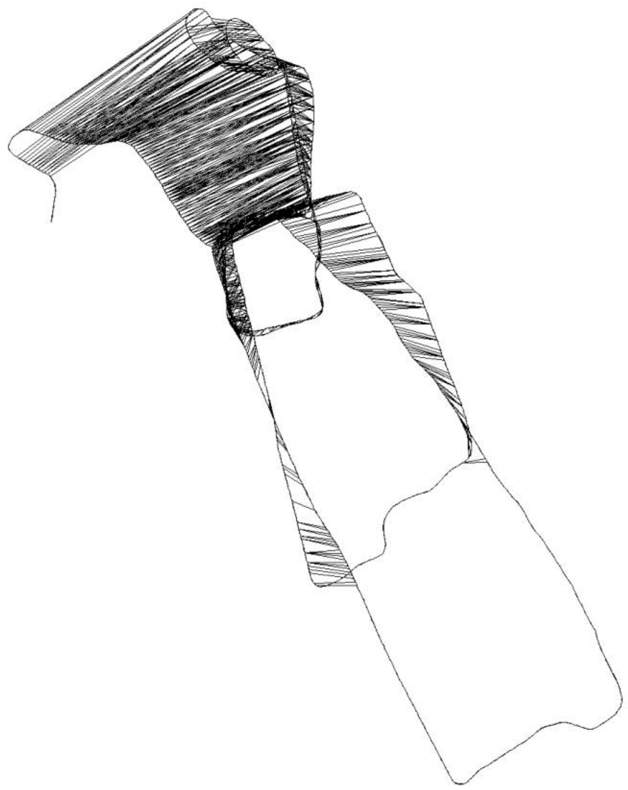
- Every variable node in the graph corresponds to a robot position and a laser measurement
- An edge (or factor-node) between two nodes represents a spatial constraint between the nodes



KUKA Halle 22, courtesy of P. Pfaff

Graph-Based SLAM in a Nutshell

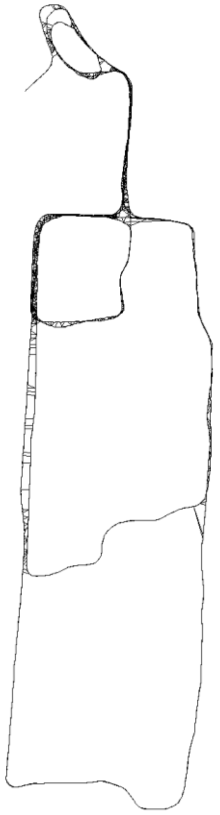
- Once we have the graph, we determine the most likely map by correcting the nodes



Graph-Based SLAM in a Nutshell

- Once we have the graph, we determine the most likely map by correcting the nodes

… like this

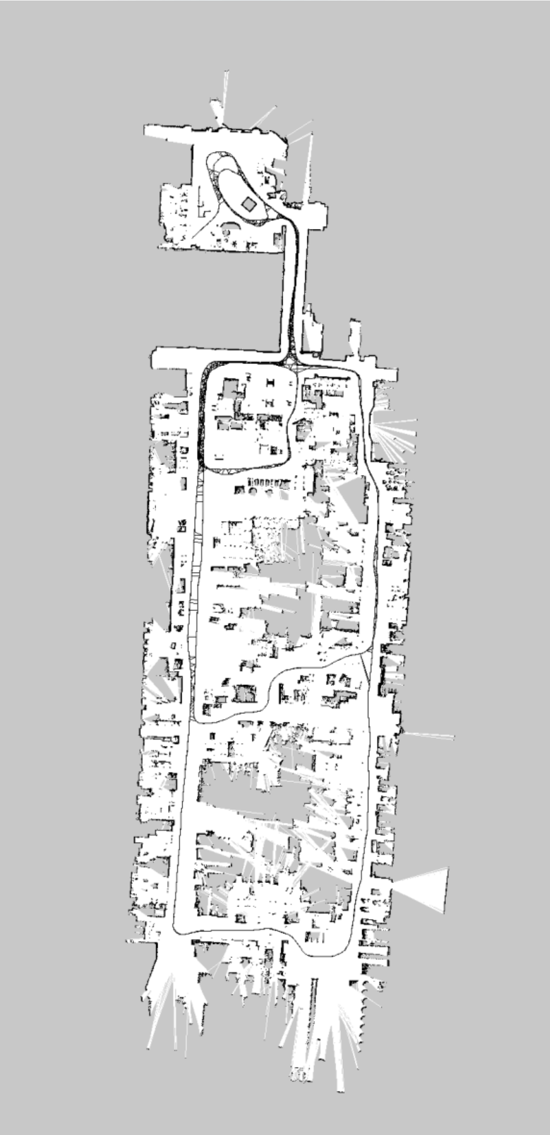


Graph-Based SLAM in a Nutshell

- Once we have the graph, we determine the most likely map by correcting the nodes

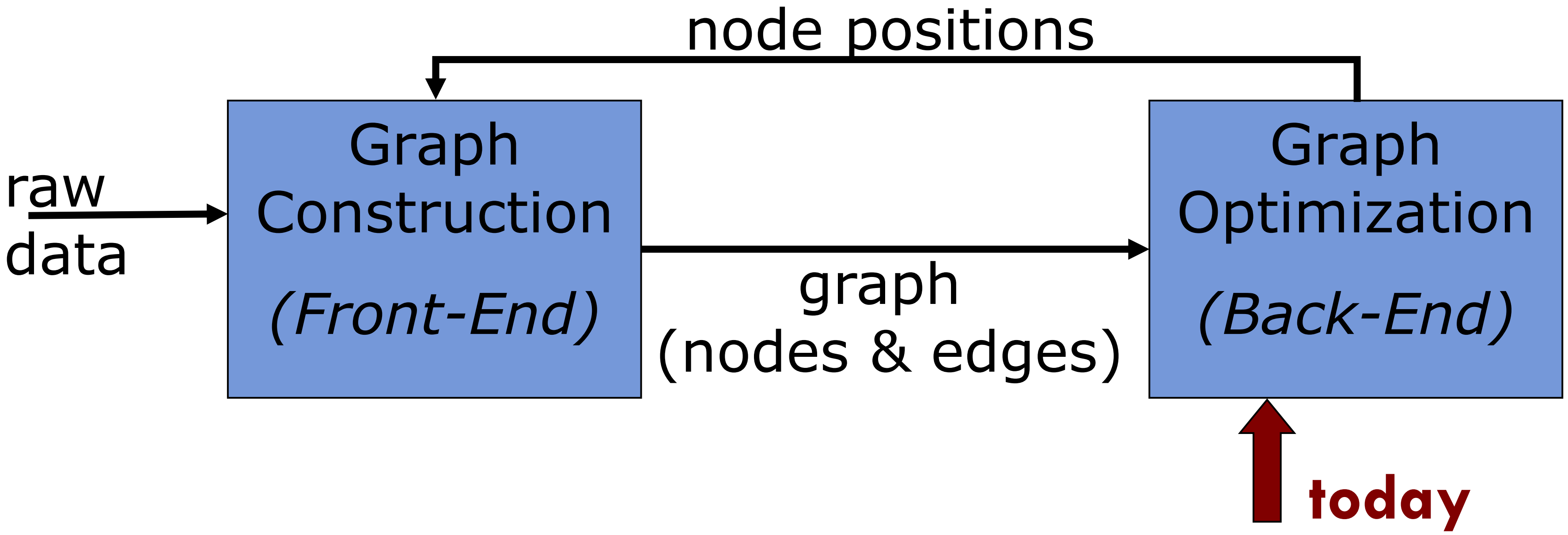
  … like this

- Then, we can render a map based on the known poses



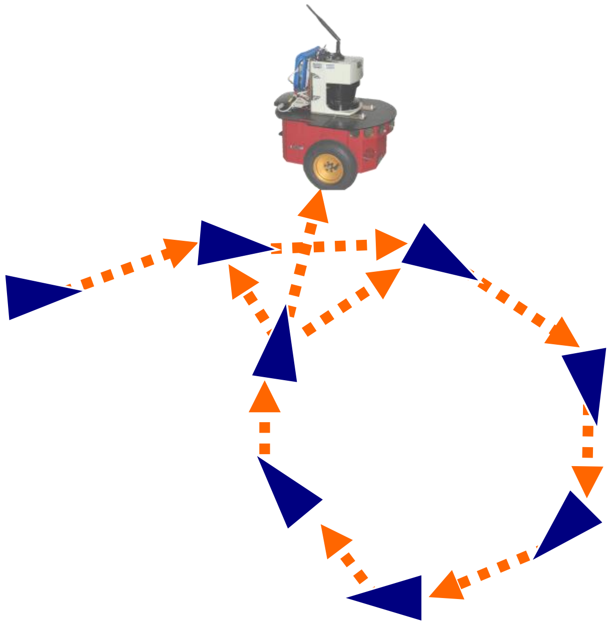
The Overall SLAM System

- Interplay of front-end and back-end
- Map helps to determine constraints by reducing the search space
- Topic today: optimization



The Pose-Graph SLAM Problem

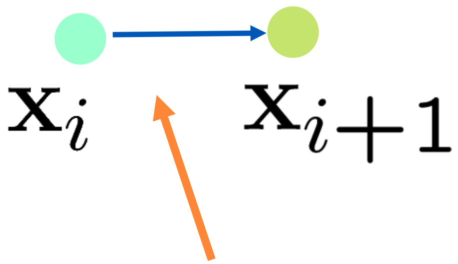
- It consists of n nodes $\mathbf{x} = \mathbf{x}_{1:n}$
- Each $\mathbf{x}_i$ is a 2D or 3D transformation (the pose of the robot at time t_i)
- A constraint/edge exists between the nodes $\mathbf{x}_i$ and $\mathbf{x}_j$ if…



Courtesy: C. Stachniss

Pose-Graph SLAM: Create an Edge If… (1)

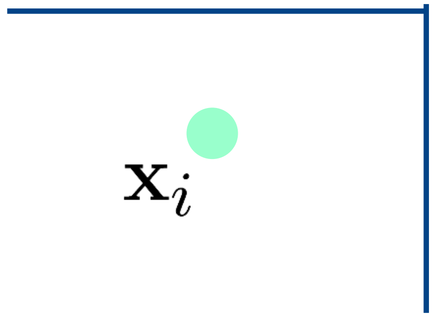
- …the robot moves from $\mathbf{x}_i$ to $\mathbf{x}_{i+1}$
- Edge (or factor) corresponds to odometry

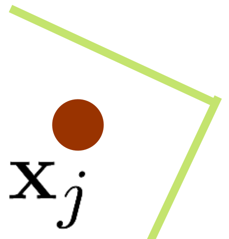


The edge represents the **odometry** measurement

Pose-Graph SLAM: Create an Edge If… (2)

- …the robot observes the same part of the environment from $\mathbf{x}_i$ and from $\mathbf{x}_j$

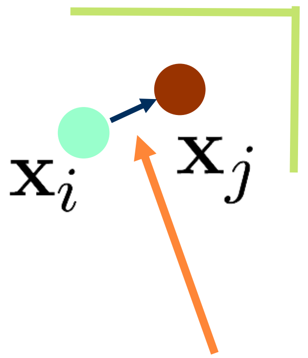




Measurement from $\mathbf{x}_i$ Measurement from $\mathbf{x}_j$

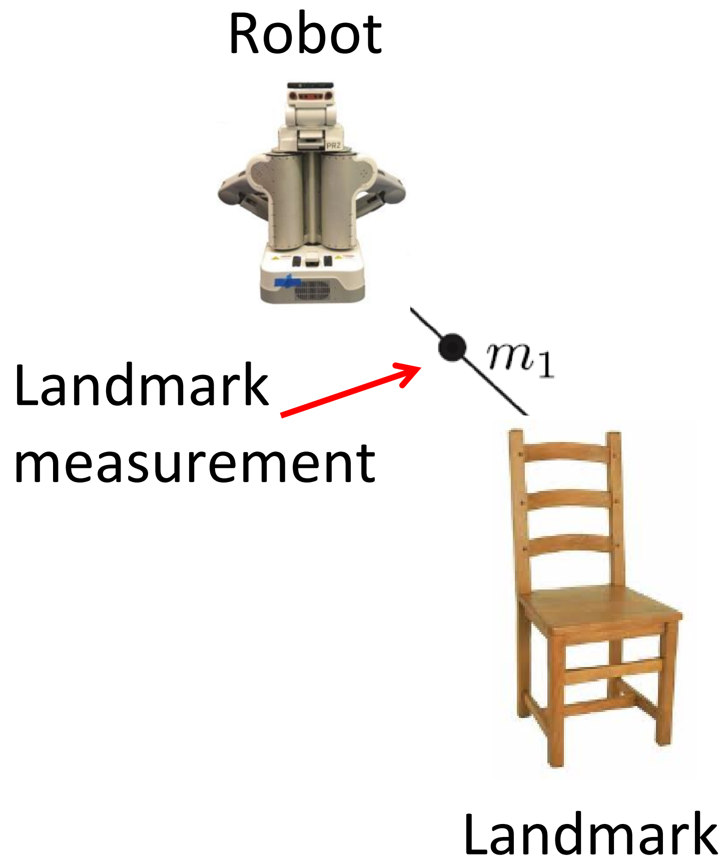
Pose-Graph SLAM: Create an Edge If… (2)

- …the robot observes the same part of the environment from $\mathbf{x}_i$ and from $\mathbf{x}_j$
- Construct a **virtual measurement** about the position of $\mathbf{x}_j$ seen from $\mathbf{x}_i$



Edge represents the position of $\mathbf{x}_j$ seen from $\mathbf{x}_i$ based on the **observation**

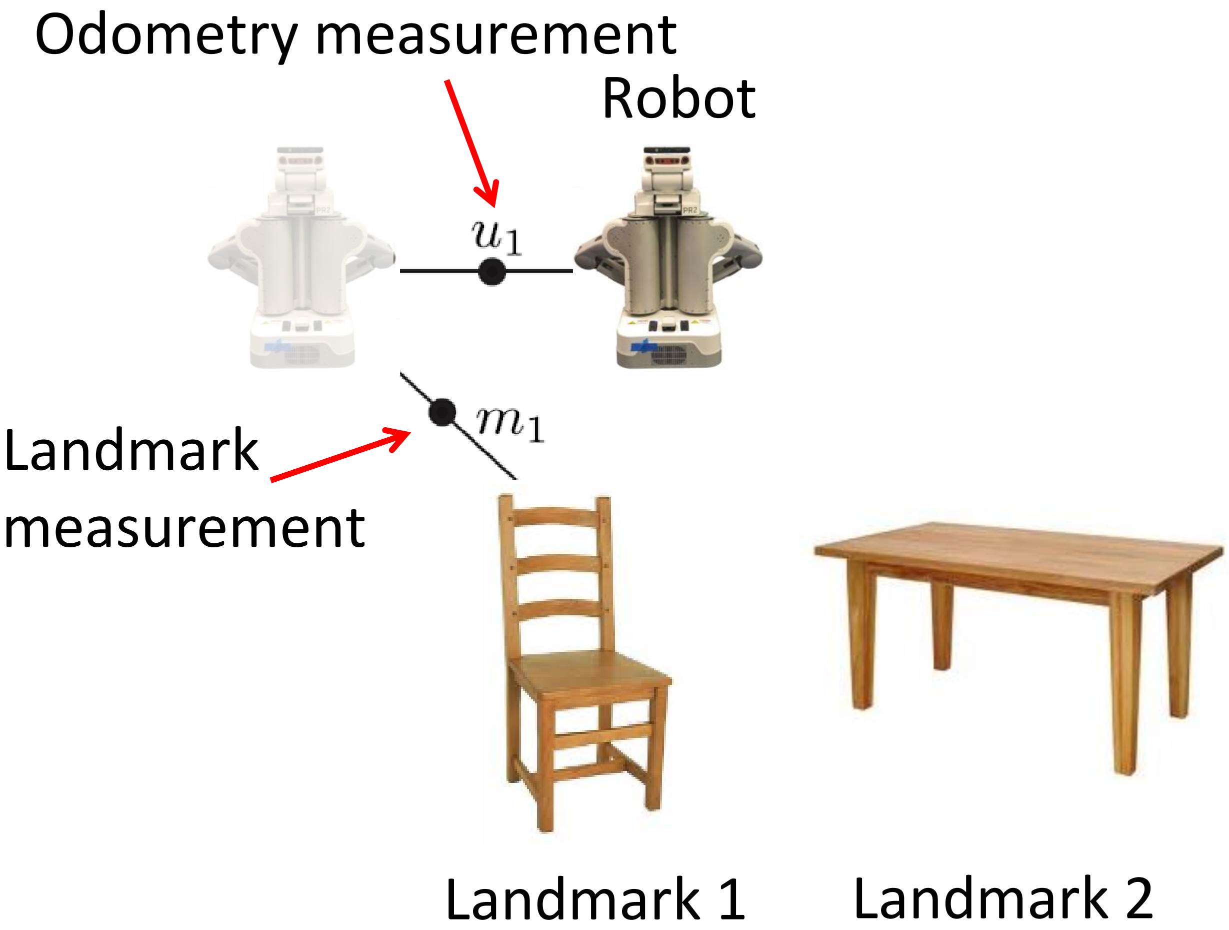
The Landmark SLAM (or Full-SLAM) Problem (t=0)



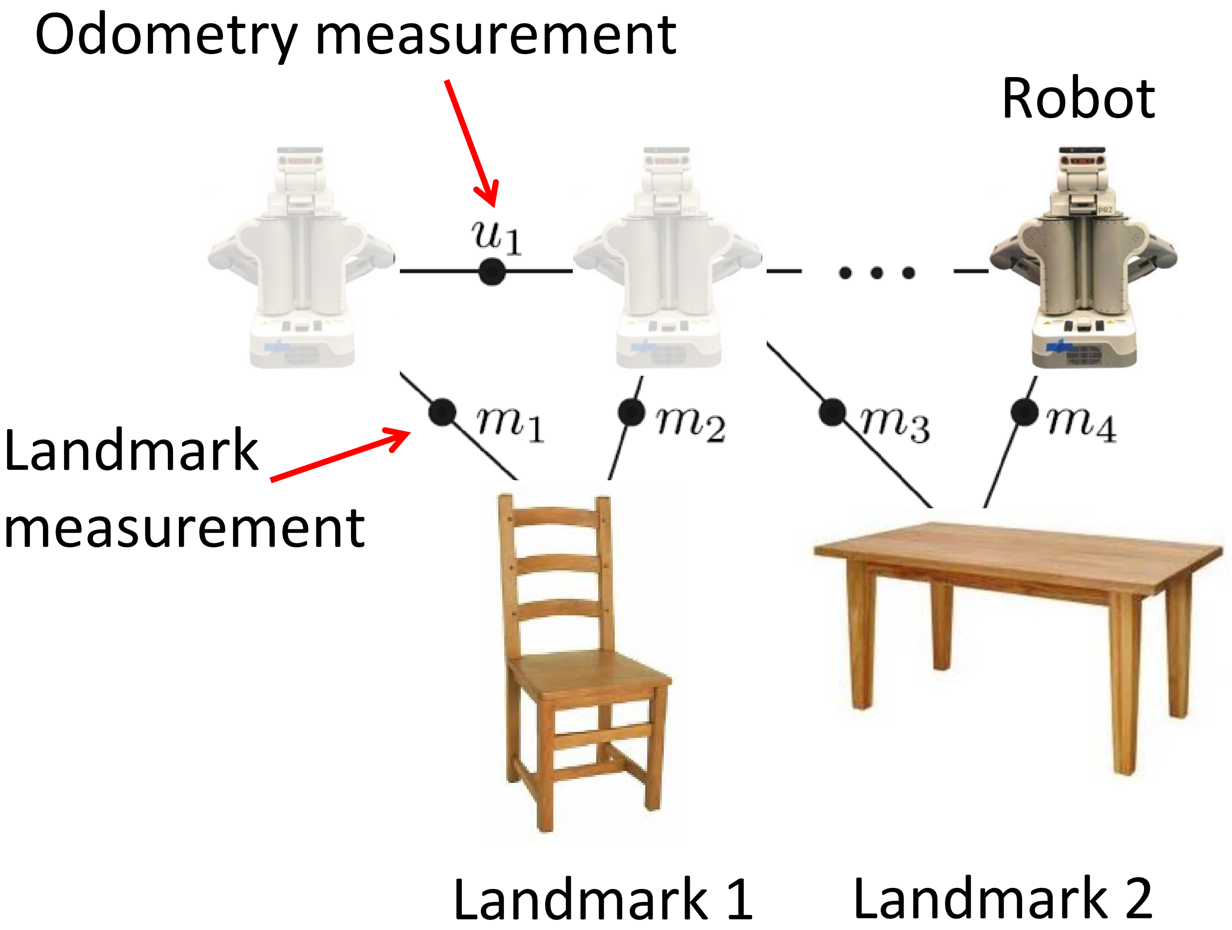
Onboard sensors:

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

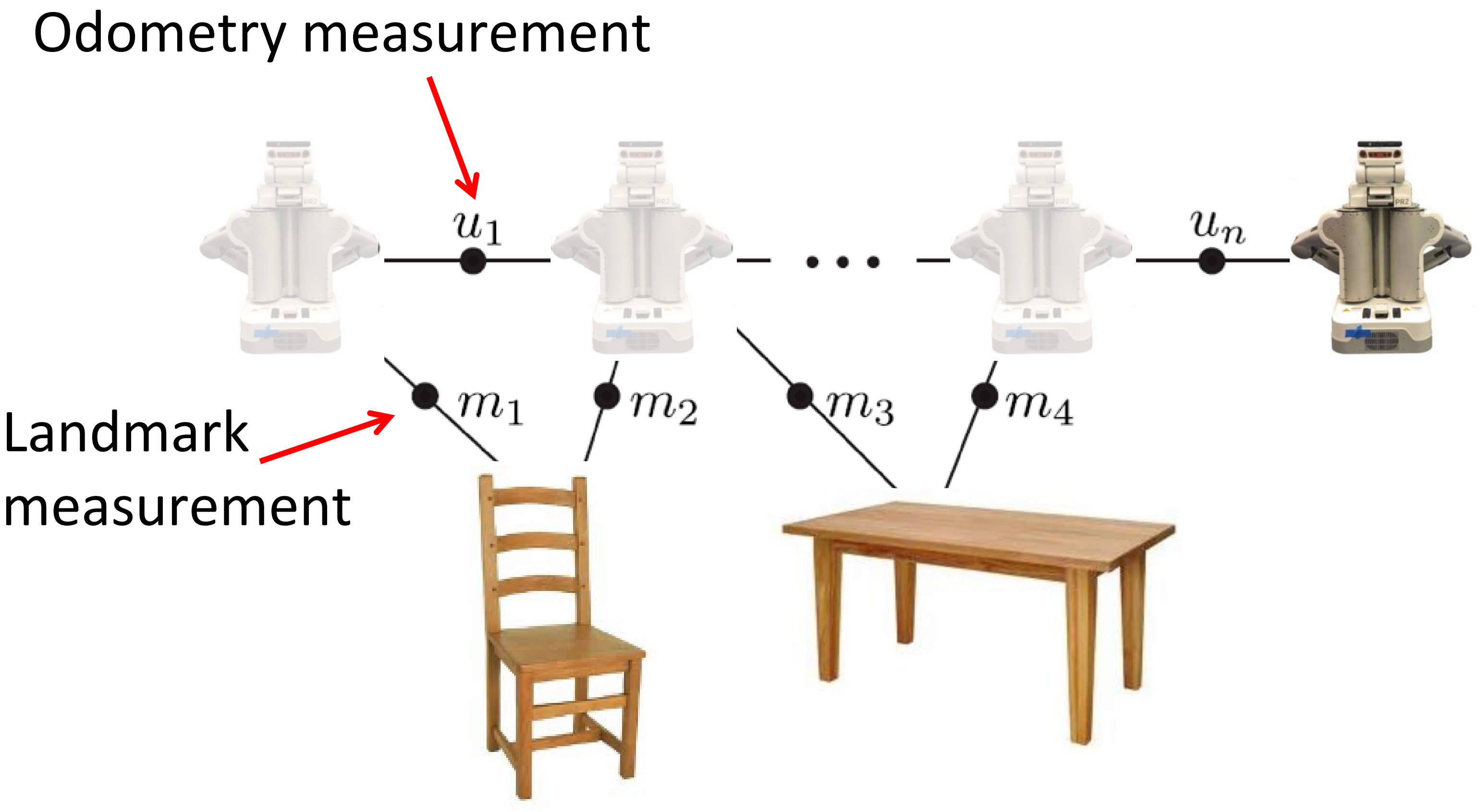
The Landmark SLAM (or Full-SLAM) Problem(t=1)



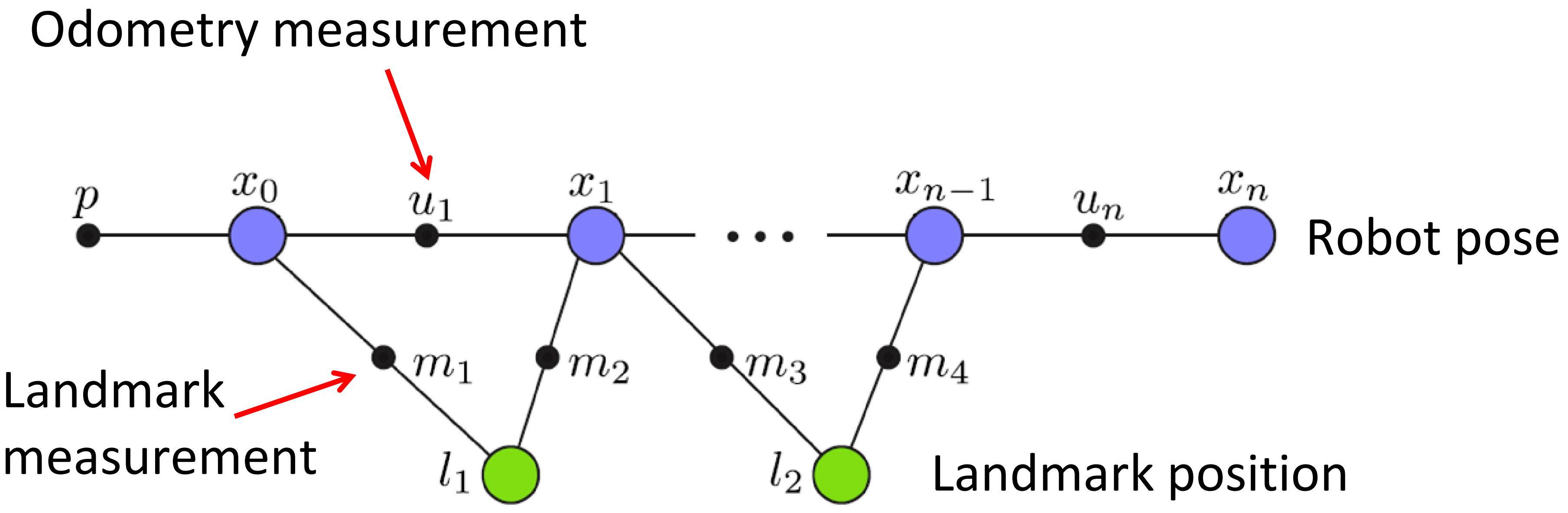
The Landmark SLAM (or Full-SLAM) Problem(t=n-1)



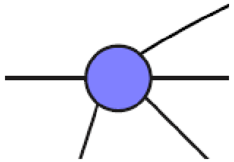
The Landmark SLAM (or Full-SLAM) Problem(t=n)

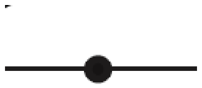


Factor Graph Representation of SLAM

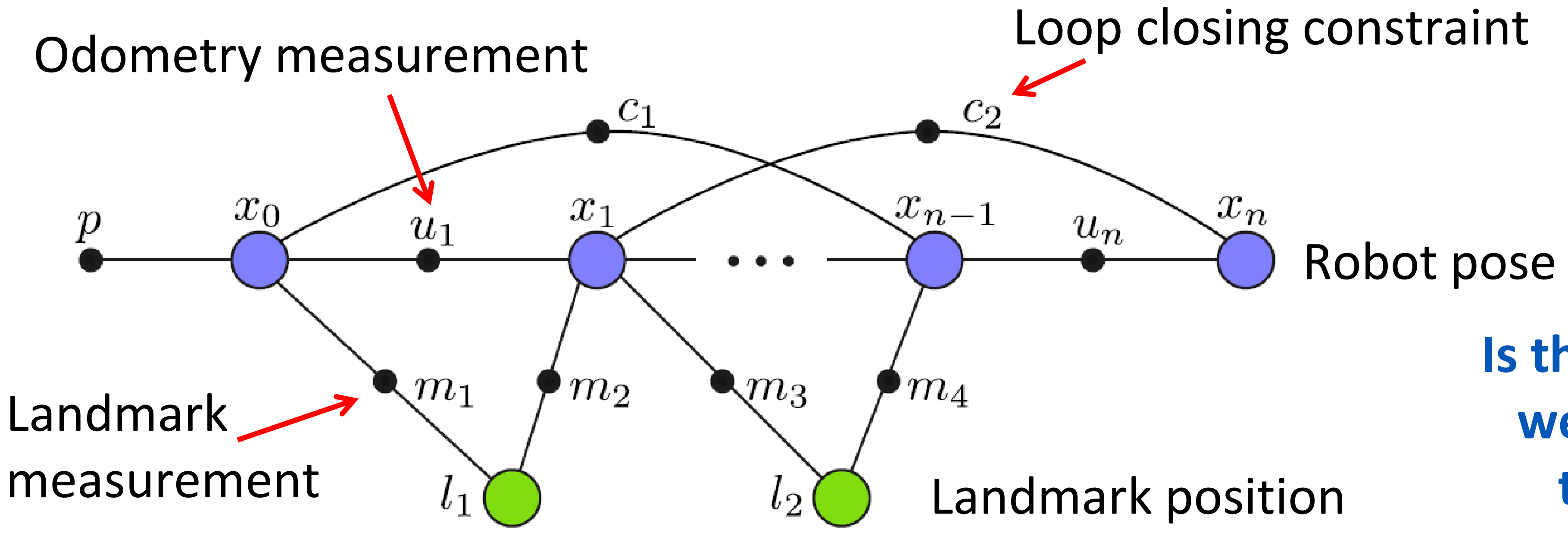


Bipartite graph with ***variable nodes*** and ***factor nodes***





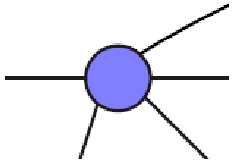
Factor Graph Representation of SLAM

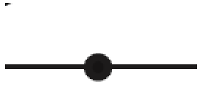


Is there another graph that we've talked about that this reminds you of?

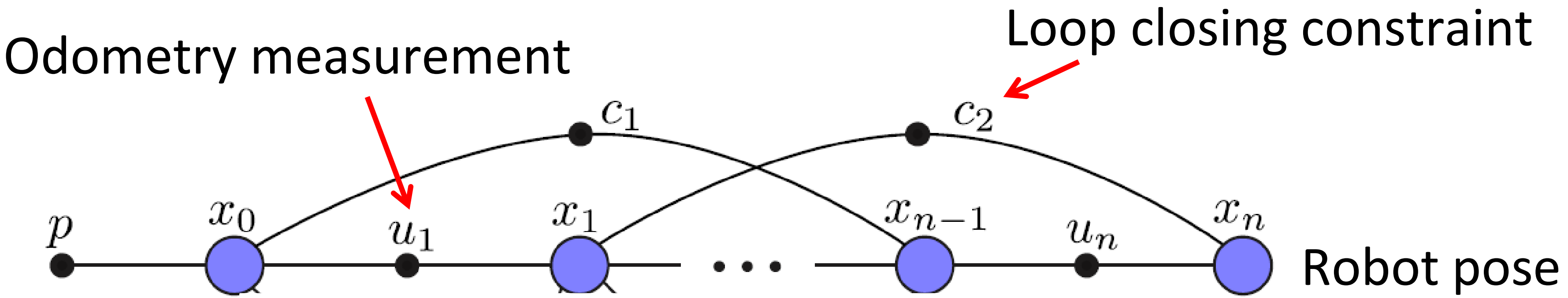
What does the factor graph encode?

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





Factor Graph Representation of SLAM



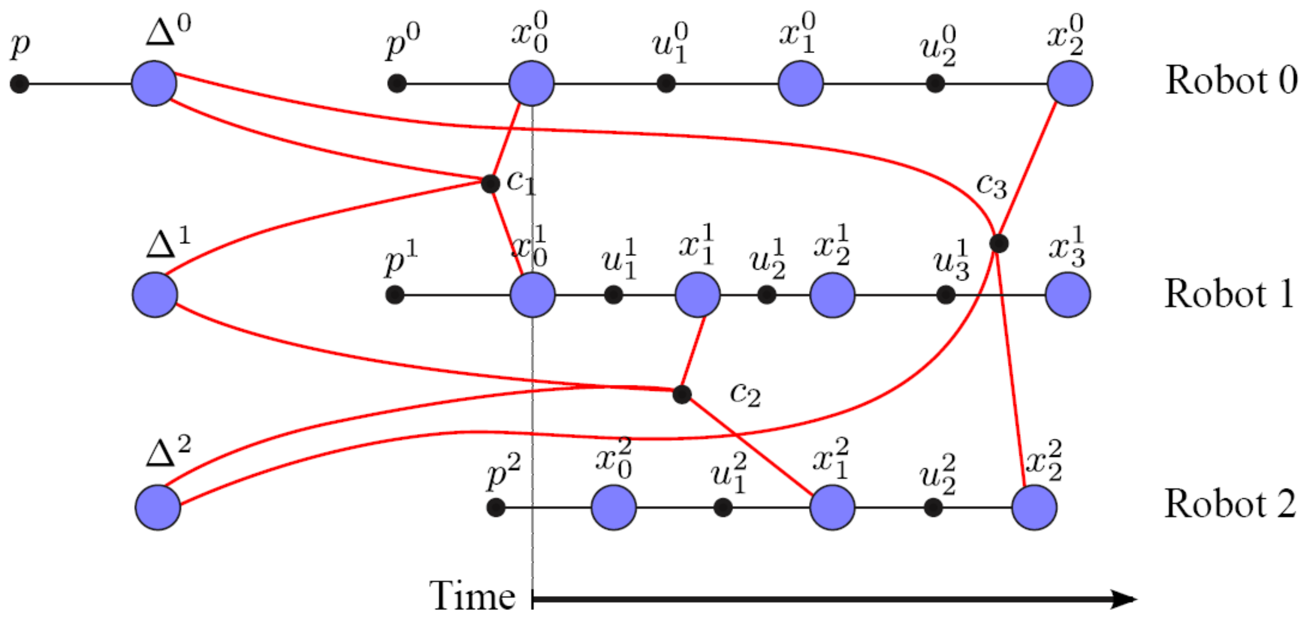
"Pose graph" (no explicit modeling of landmarks)

Is there another graph th
we've talked about tha
this reminds you of?

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

Factor Graph: Advanced Example

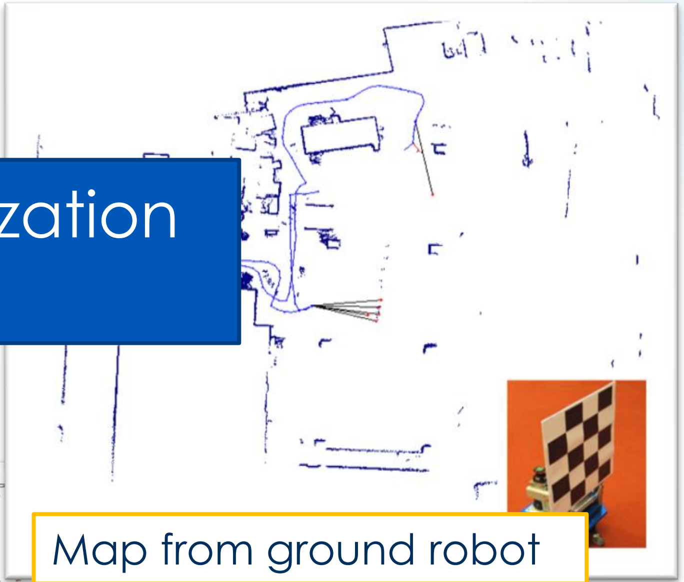
- Anchor nodes, Kim et al ICRA 2010



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

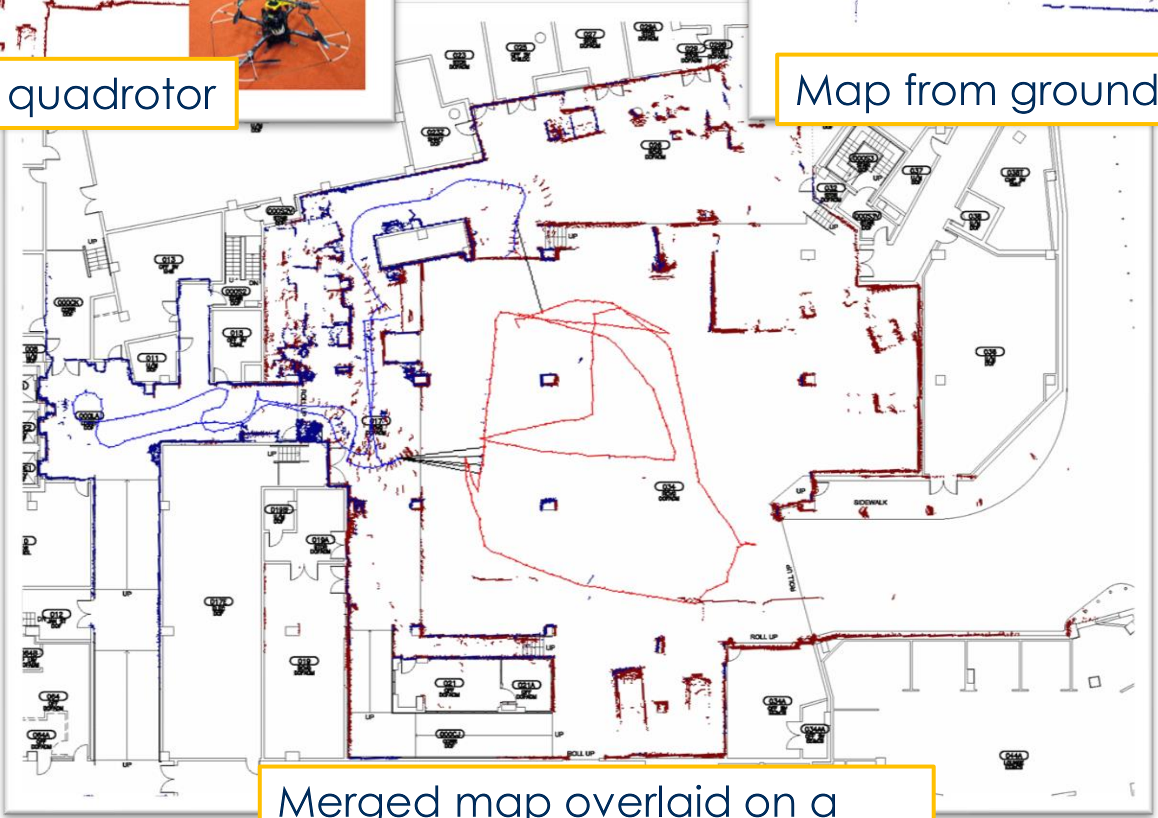
Map from quadrotor

0.3s overall optimization time!



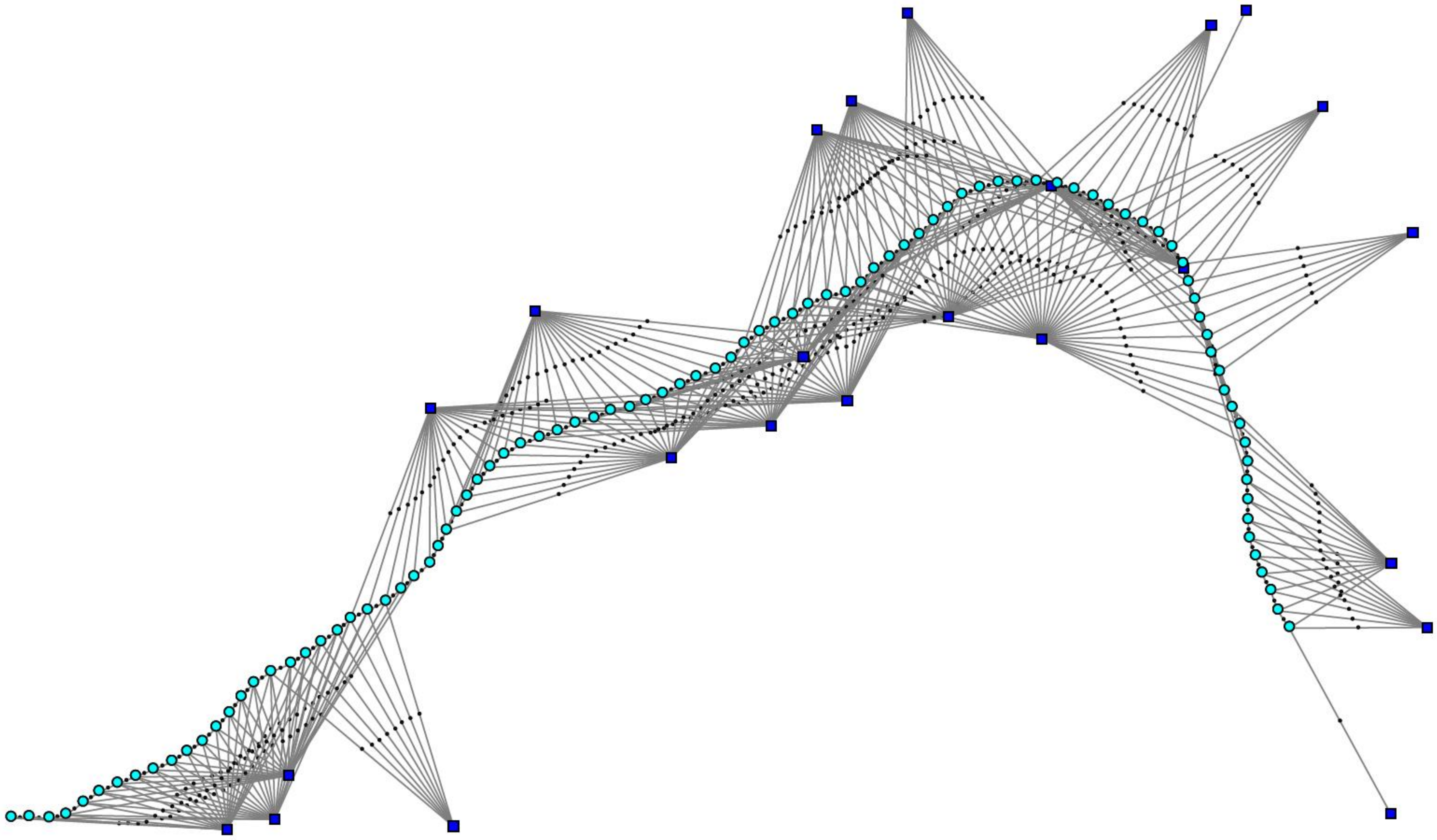
Map from ground robot

Anchor Nodes
Kim et al ICRA 2010



Merged map overlaid on a floor plan

Larger Factor Graph SLAM Example



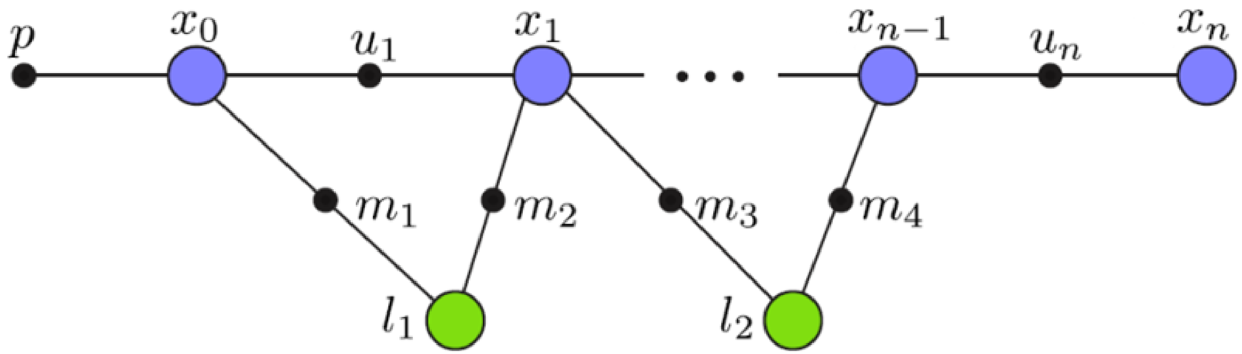
Nodes - Variables and Measurements

Is there another graph that we've talked about that this reminds you of?

- 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 (or Factors):

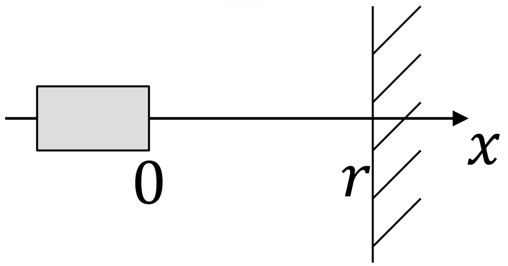
$$Z = \{p, u_1 \cdots u_n, m_1 \cdots m_4\}$$

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

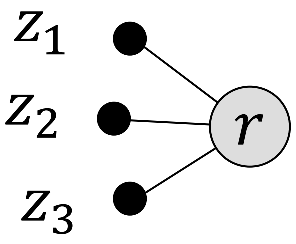
What does the factor graph encode?

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?

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

- Simulate
  - Given the variable (r), we can draw samples from ν to simulate the measurement process
- Test
  - Given the variable (r) and a measurement (z), evaluate the measurements 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)

Optimally Inferring the Missing Variables

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

Bayes Rule

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

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

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)

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

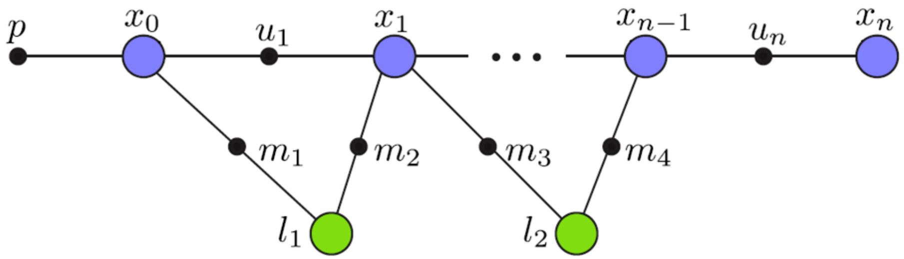
- Maximum Likelihood Estimator (MLE)

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

Factorization of Probability Density

- Conditional independence:

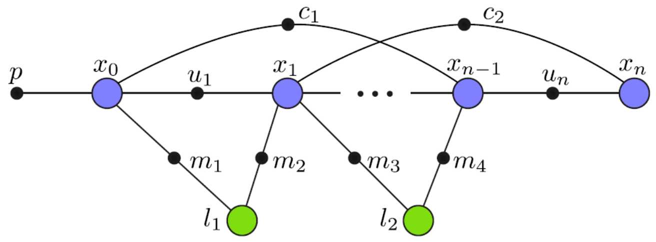
$$p(z_1 z_2|\Theta) = p\,(z_1|\Theta)\,p\,(z_2|\Theta)$$



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

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

Nonlinear Least-Squares

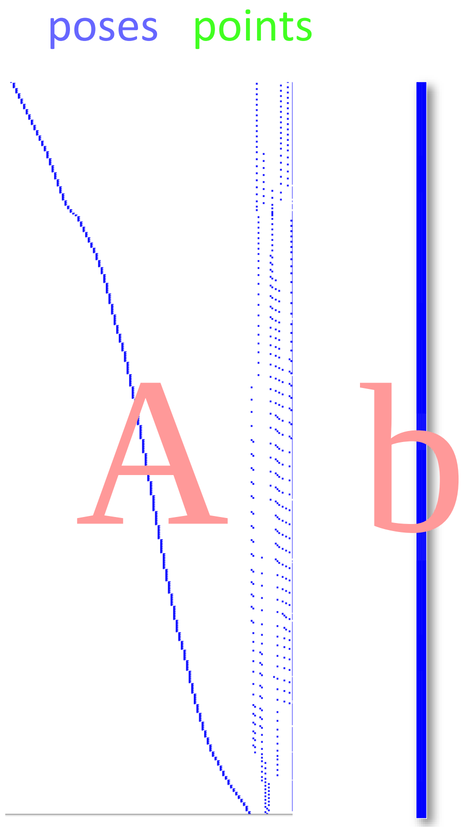


$$\operatorname{argmin}_{\mathrm{x}} \sum_i \|h_i(\mathrm{x})\|_{\Xi}{}^2$$

Repeatedly solve linearized system (GN)

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

$$A = \begin{bmatrix} F_{11} & & G_{11} & & \\ F_{12} & & & G_{12} & \\ F_{13} & & & & G_{13} \\ & F_{21} & G_{21} & & \\ & F_{22} & & G_{22} & \\ & F_{23} & & & G_{23} \end{bmatrix}, \quad x = \begin{bmatrix} \xi_1 \\ \xi_2 \\ \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}, \quad b = \begin{bmatrix} b_{11} \\ b_{12} \\ b_{13} \\ b_{14} \\ b_{15} \\ b_{16} \end{bmatrix}$$

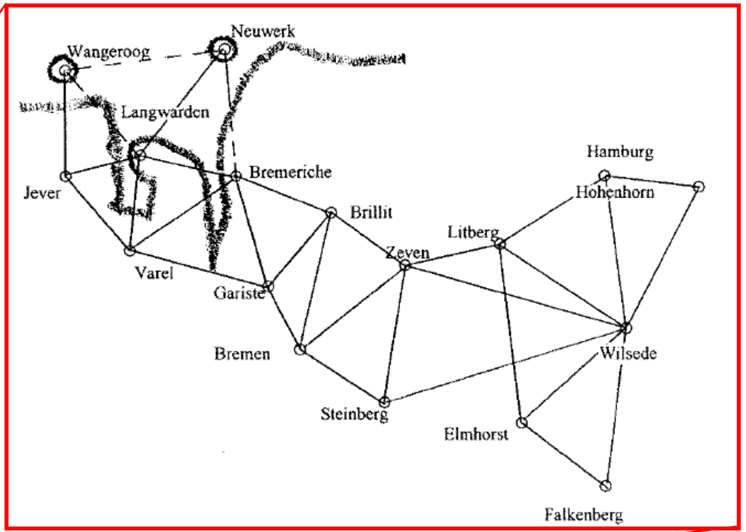


$$\left.\frac{\partial h_i}{\partial \mathrm{x}}\right|_{\hat{\mathrm{x}}} \qquad h_i\,(\hat{\mathrm{x}})$$

Mapping: A Brief History

Carl Friedrich Gauss

1828: Triangulation of Kingdom of Hanover



Courtesy: M. Kaess

SLAM as a Least-Squares Problem

- On the board:
  - SLAM as MLE or MAP Estimation
    - MLE and MAP Estimation
    - Transforming MLE/MAP Estimation into Non-Linear Least Squares