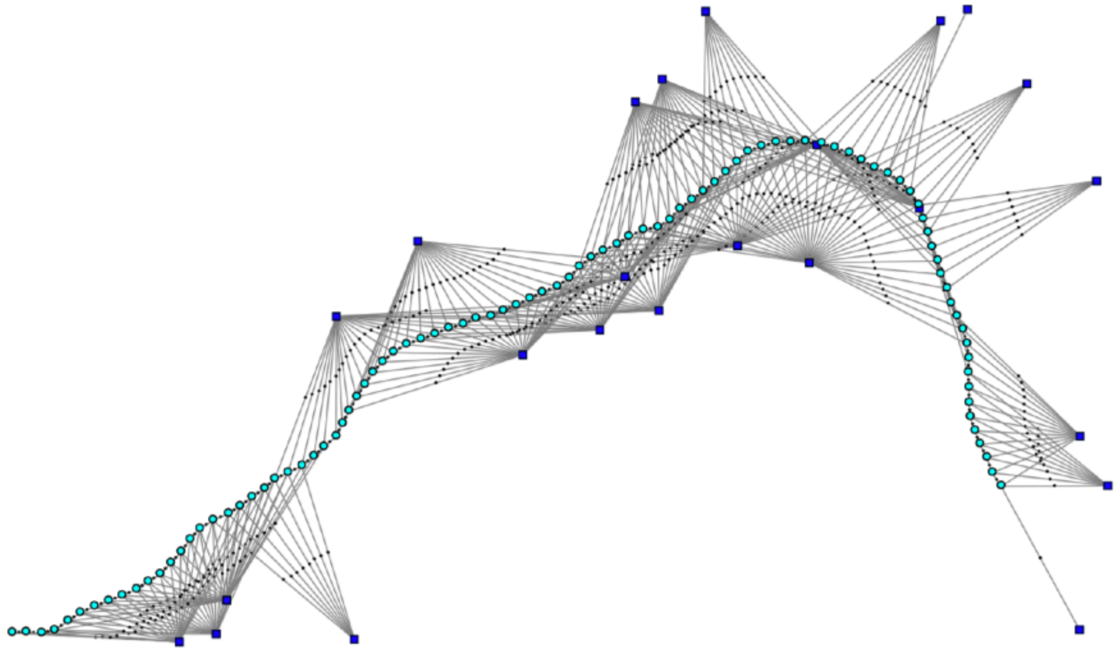


[courtesy of Frank Dellaert]

16.485: VNAV - Visual Navigation for Autonomous Vehicles

Luca Carlone

Lecture 24: SLAM II:
Factor Graphs
and Marginalization

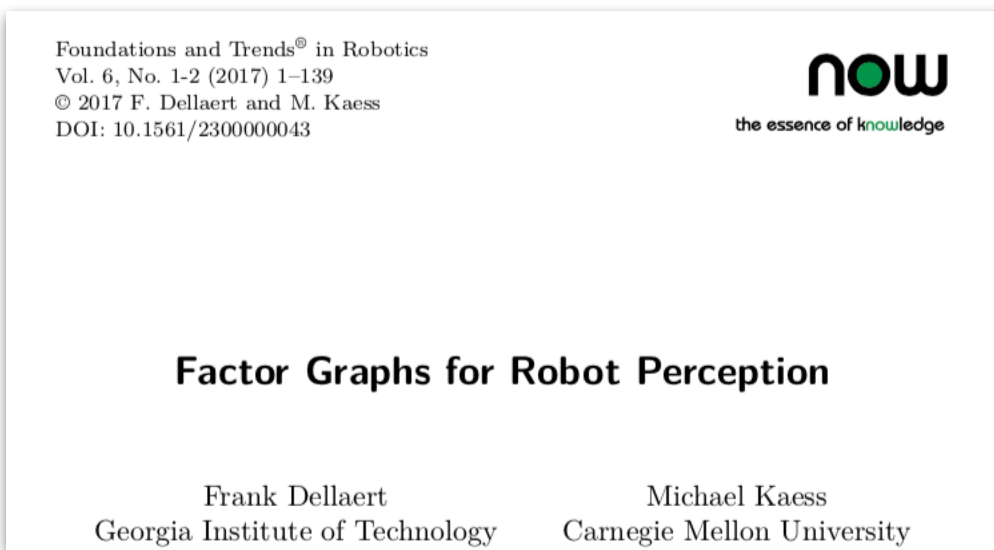
based on slides by Kasra Khosoussi

Week	Dates	Lecture topic	Lab
1	Sep 8, 10	Introduction	Lab 1: Linux, C++, Git
2	Sep 13, 15, 17	3D Geometry	Lab 2: ROS
3	Sep 20, 22, 24	Geometric Control	Lab 3: 3D trajectory following
4	Sep 27, 29	Trajectory Optimization	Lab 4: 3D trajectory optimization
5	Oct 1, 4, 6	2D Computer Vision	Lab 5: feature detection
6	Oct 8, 13, 15	2-view Geometry and Minimal Solvers	Lab 6: object localization
7	Oct 18, 20, 22	Non-minimal Solvers and Visual Odometry	Lab 7: GTSAM
8	Oct 25, 27, 29	Place Recognition	Lab 8: ML for robotics
9	Nov 1, 3, 5	SLAM and Visual-Inertial Navigation	Lab 9: SLAM
10	Nov 8, 10, 12	Advanced Topics: Open Problems in Robot Perception	Final project
11	Nov 15, 17, 19	Advanced Topics: Robustness	Final project
12	Nov 22, 24, 29, Dec 1	Advanced Topics: Metric-Semantic Understanding and Learning	Final project
13	Nov 25-26	Thanksgiving Break	
14	Dec 3, 6, 8	Guest Lectures and Students Presentations	Final project

MIT AEROASTRO

Today

- **Recap**: pose graph optimization + landmark-based SLAM
- Factor Graphs
- Marginalization


Foundations and Trends® in Robotics
Vol. 6, No. 1-2 (2017) 1–139

DOI: 10.1561/2300000043

now
the essence of knowledge

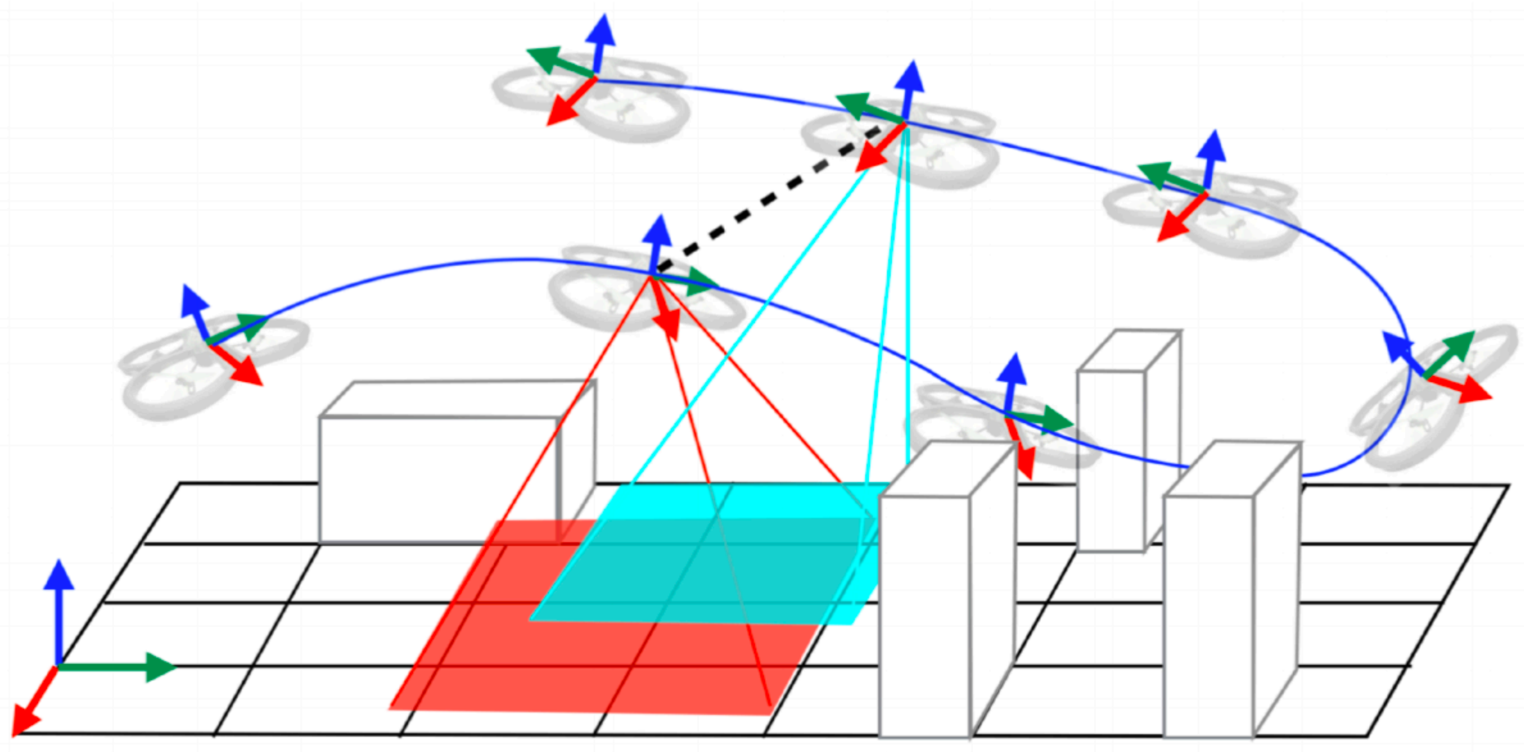
Factor Graphs for Robot Perception

Frank Dellaert
Georgia Institute of Technology

Michael Kaess
Carnegie Mellon University

MIT AEROASTRO

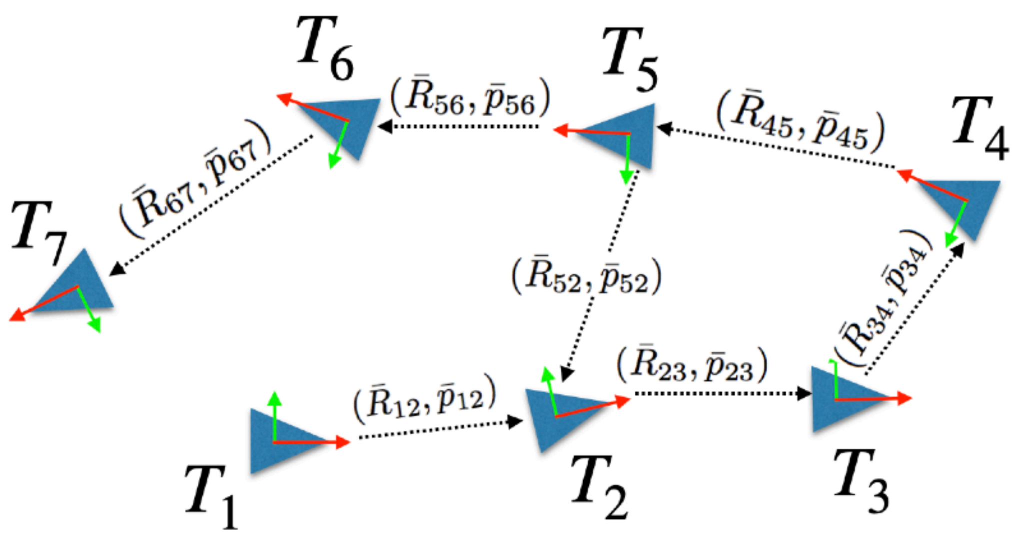
Pose Graph Optimization



- **Measurements**: odometry + loop closures (relative poses)
- **Variables**: robot poses

Pose Graph Optimization

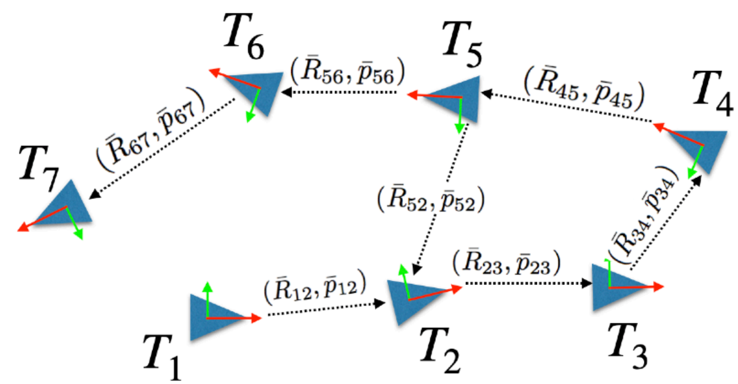
pose
graph



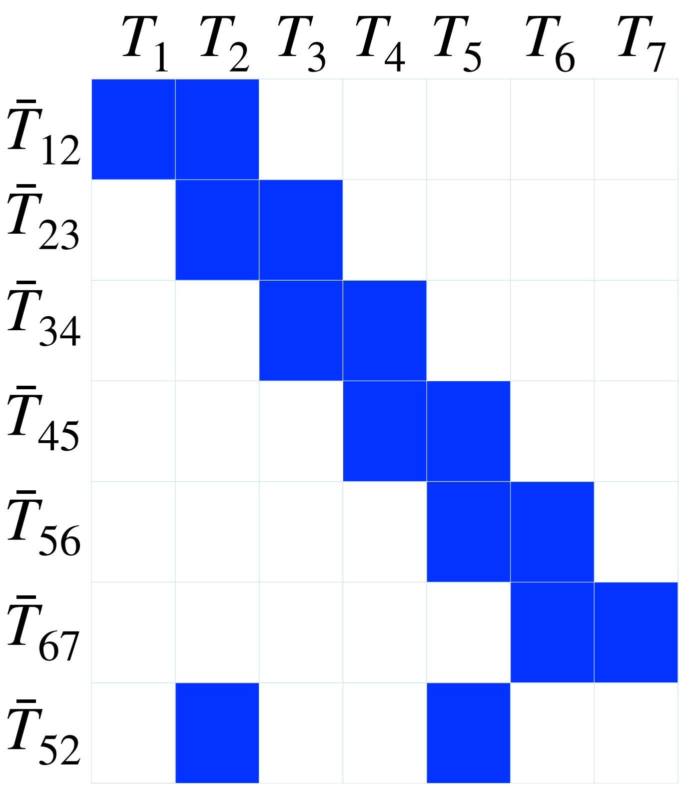
pose graph
optimization

$$\min_{\boldsymbol{T}_t, t=1,\ldots,n} \sum_{(i,j)\in\mathcal{E}} \|(\boldsymbol{T}_i^{-1}\,\boldsymbol{T}_j) \boxminus \bar{\boldsymbol{T}}_j^i\|^2_{\boldsymbol{\Sigma}_{ij}}$$

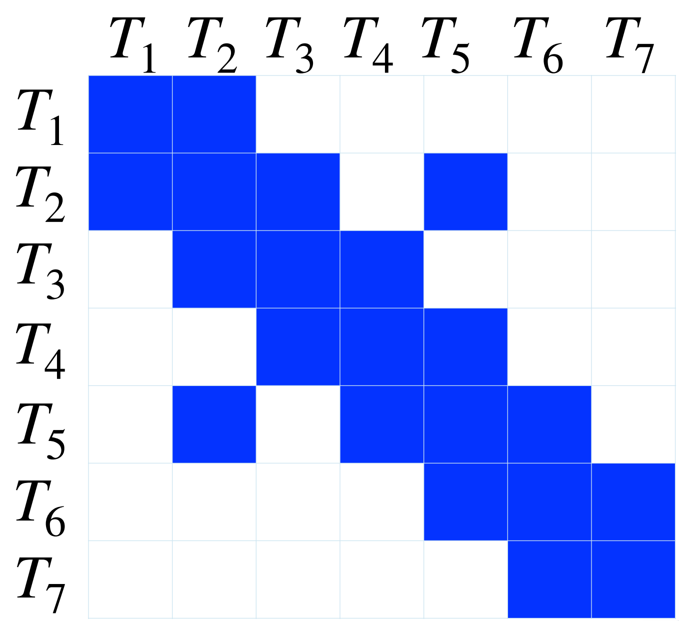
Pose Graph Optimization: Sparsity



Jacobian **J**



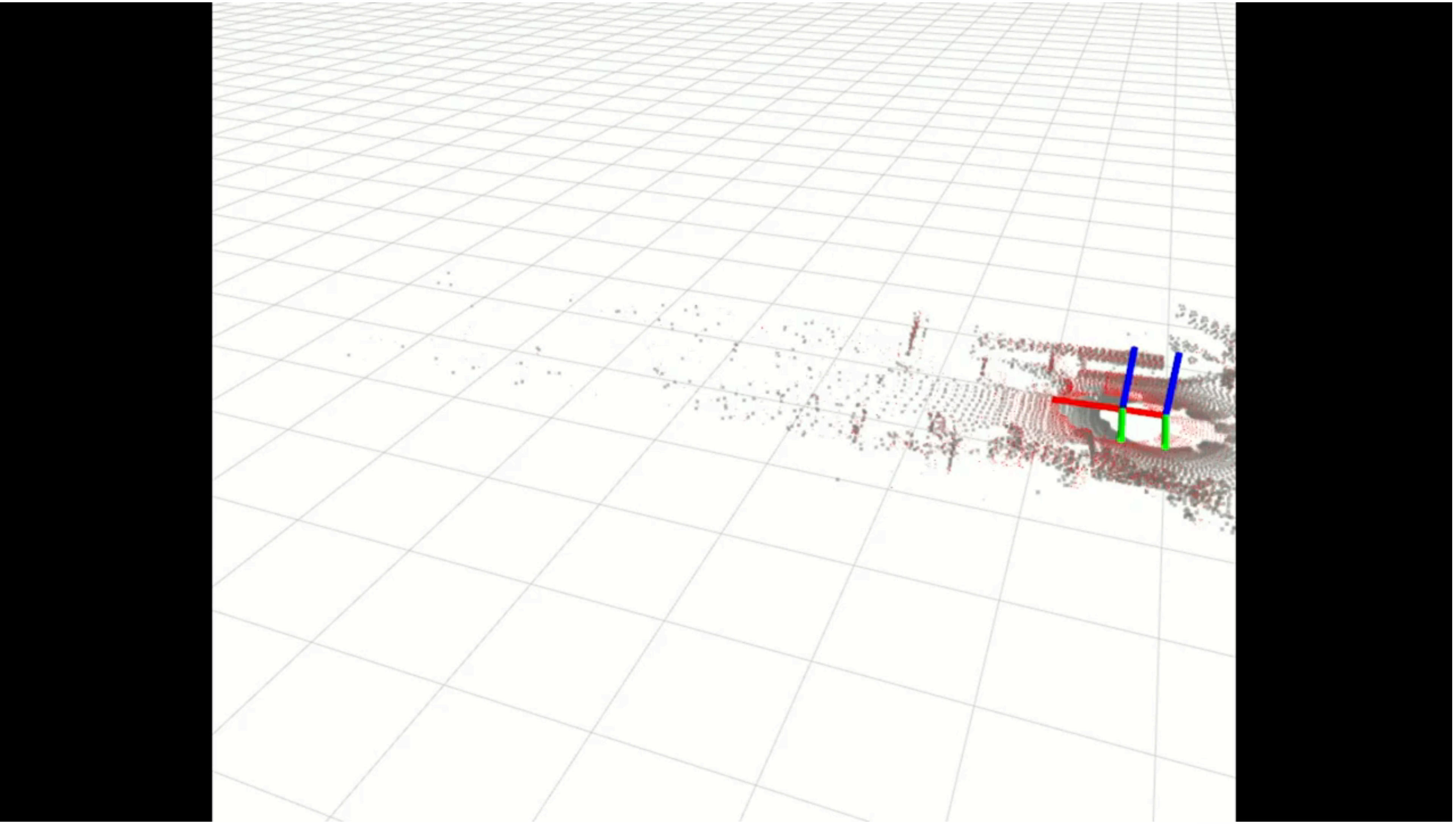
Hessian $\mathbf{J}^T \mathbf{J}$



a.k.a. Information Matrix of the estimate

MIT AEROASTRO

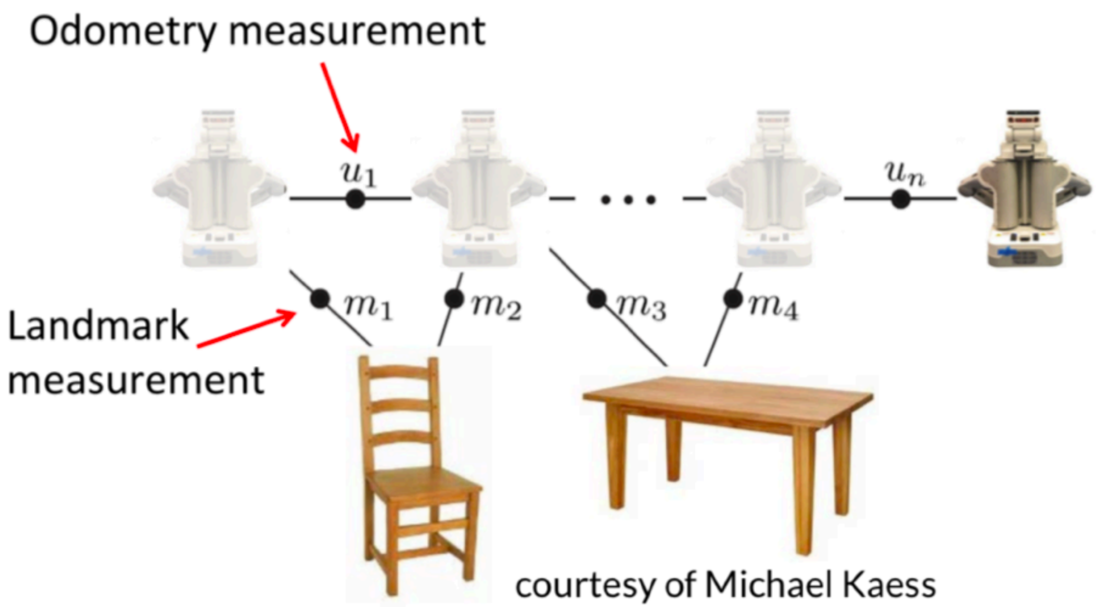
Pose Graph Optimization: Example

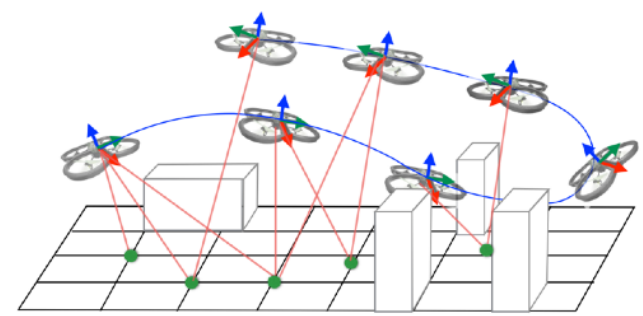


https://www.youtube.com/watch?v=KYvOqUB_odg

Landmark-based SLAM

- Sequence of robot (camera) poses $\mathbf{T}_1, \mathbf{T}_2, \ldots, \mathbf{T}_t \in \mathrm{SE}(d)$
- Robot measures the relative pose between $\mathbf{T}_i$ and $\mathbf{T}_{i+1}$ (odometry)
- Robot measures the environment (e.g., point landmarks $\mathbf{p}_i \in \mathbb{R}^d$)



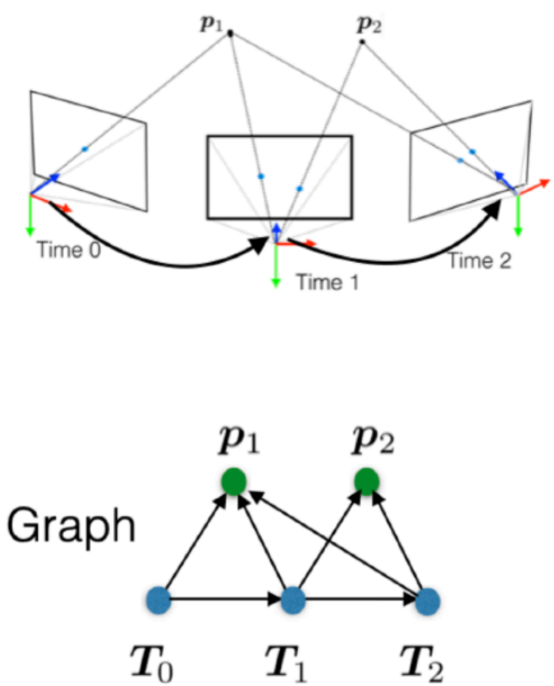


- **Measurements**: odometry + measurements of (projection, range, position, or others) of external landmarks
- **Variables**: robot poses and landmark positions

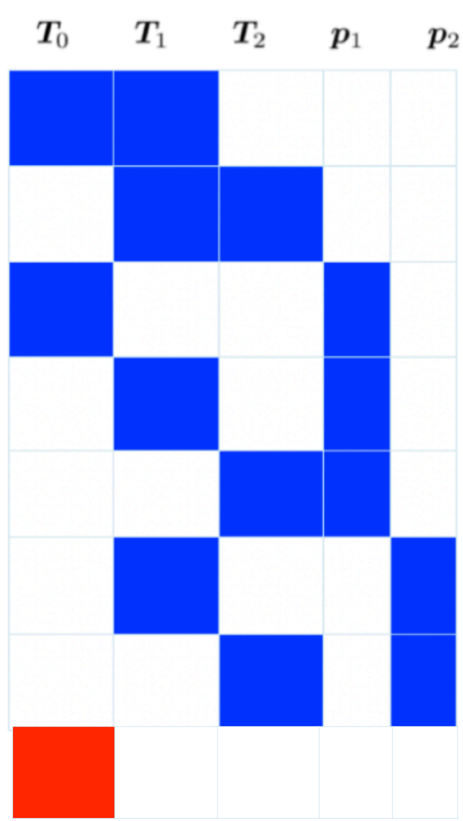
MIT AEROASTRO

Landmark-based SLAM: Sparsity

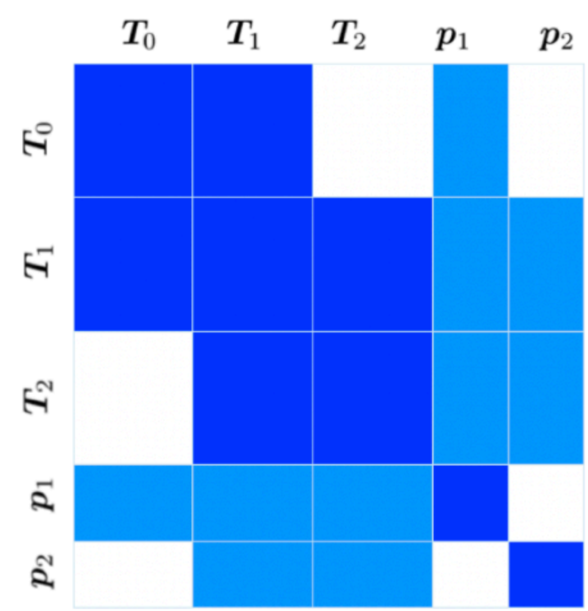
$$\min_{\substack{\boldsymbol{T}_t, t=1,\dots,n \\ \boldsymbol{l}_k, k=1,\dots,K}} \sum_{t=1,\dots,n-1} \|(\boldsymbol{T}_t^{-1}\, \boldsymbol{T}_{t+1}) \boxminus \bar{\boldsymbol{T}}_{t+1}^{t}\|_{\boldsymbol{\Sigma}_o}^2 + \sum_{k=1,\dots,K} \sum_{t \in \mathcal{S}_k} \|\bar{\boldsymbol{y}}_{k,t} - h_i(\boldsymbol{T}_t, \boldsymbol{l}_k)\|_{\boldsymbol{\Sigma}_l}^2$$



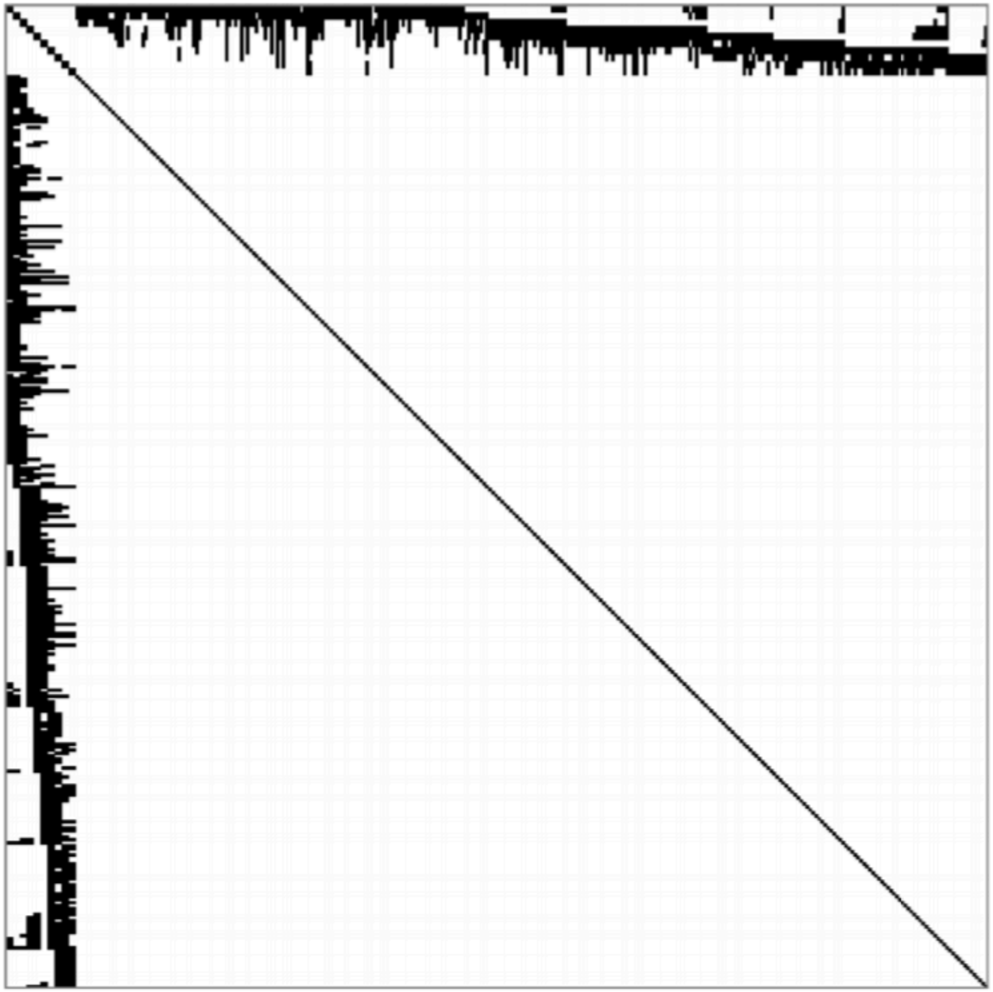
Jacobian **J**



Hessian **JᵀJ**

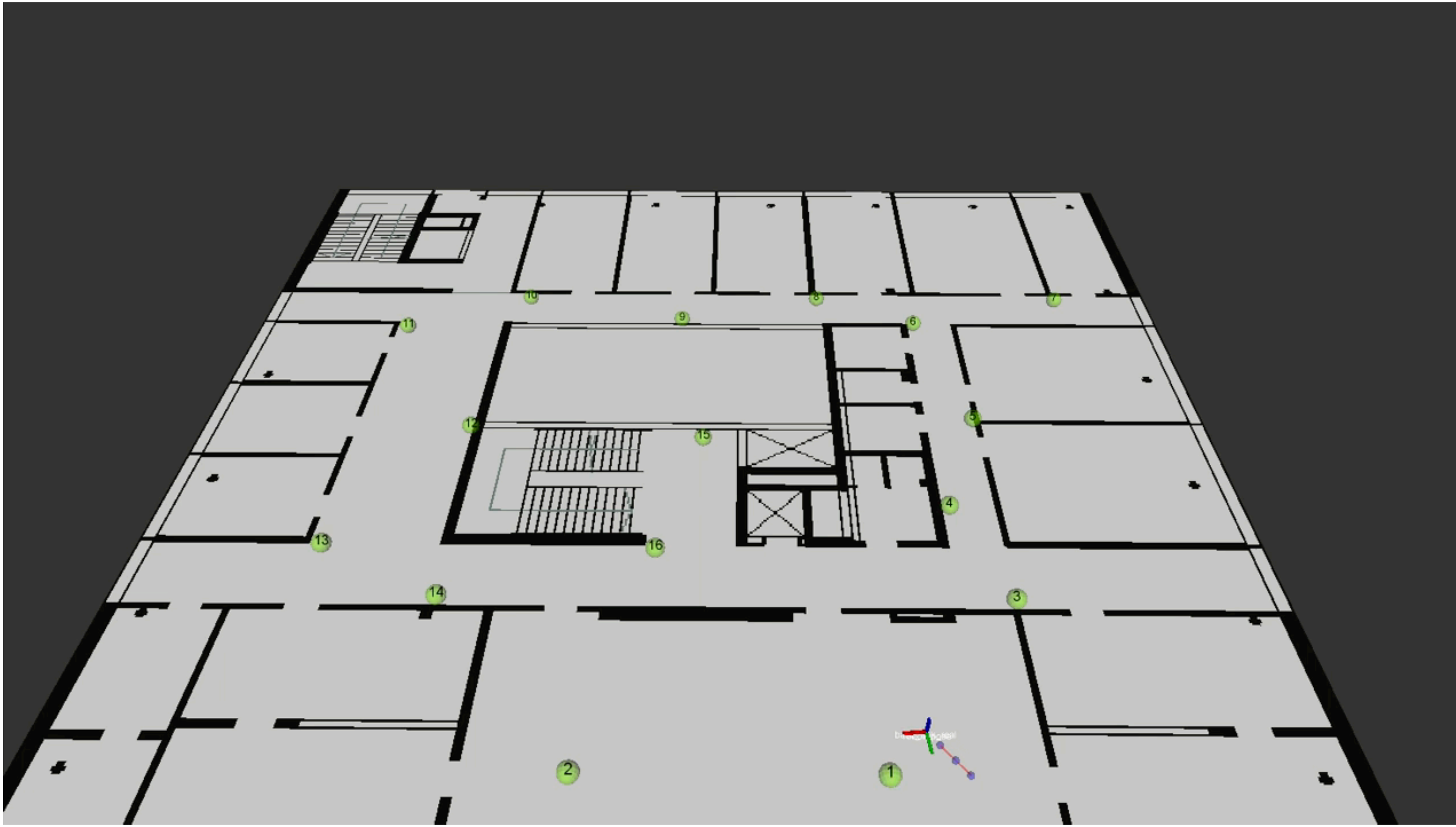


Example of Hessian (sparsity) in BA



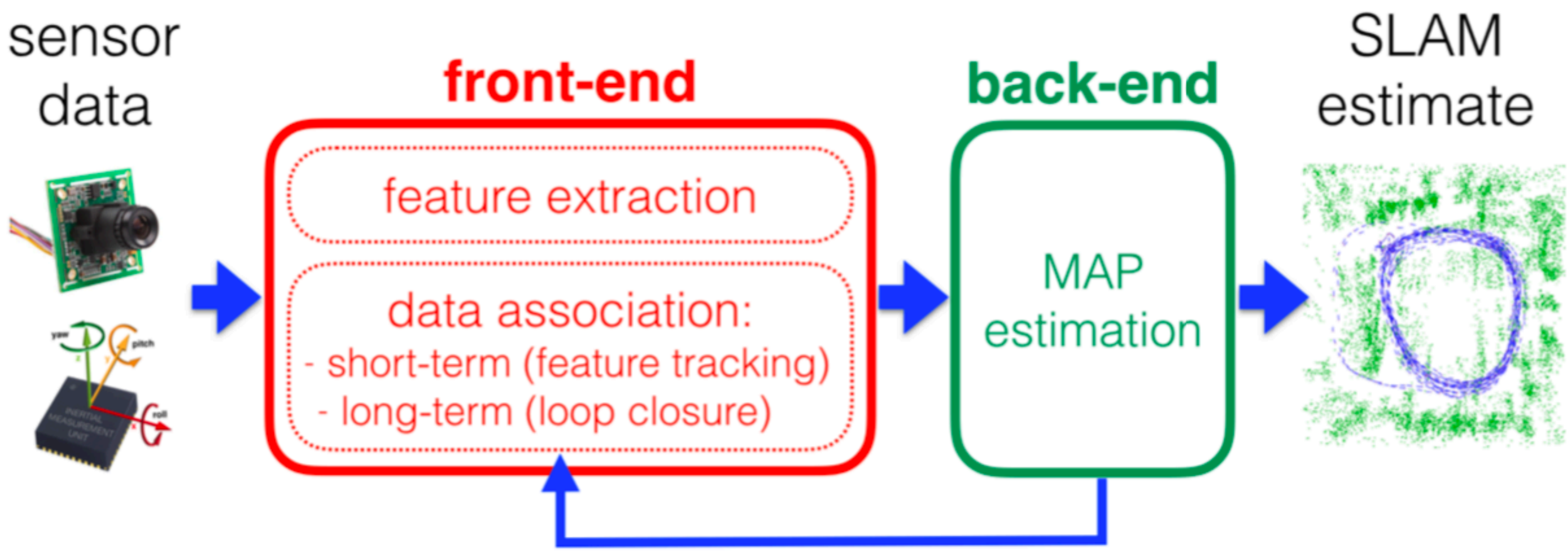
Credit: *Lourakis and Argyros*

Landmark-based SLAM: Example



https://www.youtube.com/watch?v=OdJ042prg_M

Some terminology



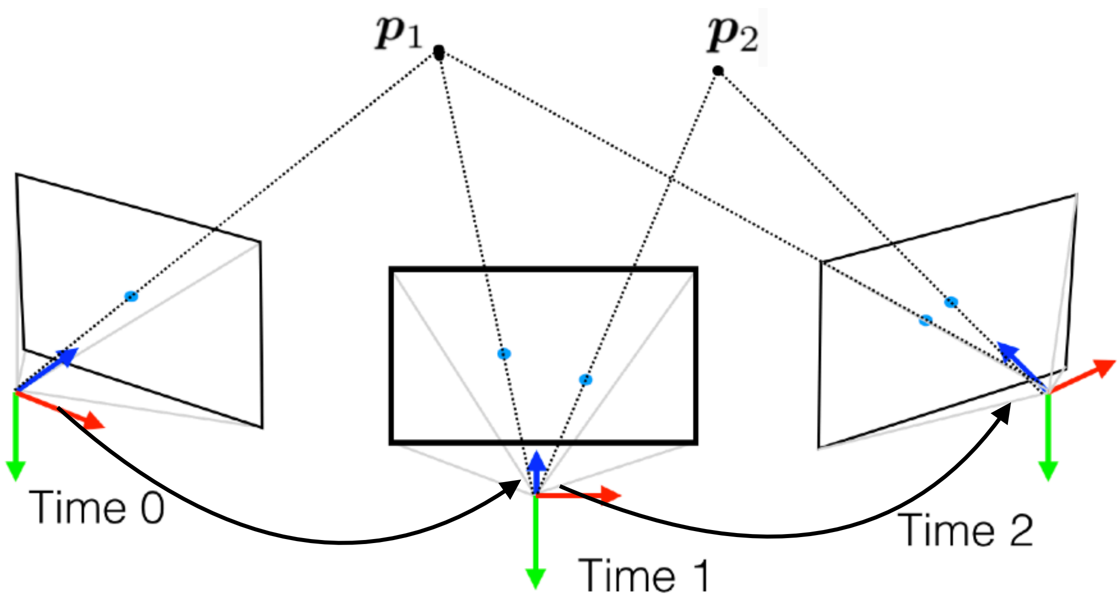
MAP is maximum *a posteriori* estimation
(MLE if no prior is available ["uninformative" prior])

courtesy of Cadena et al.

Other SLAM Problems

- Consider a visual-SLAM problem where we also want to estimate the camera calibration:

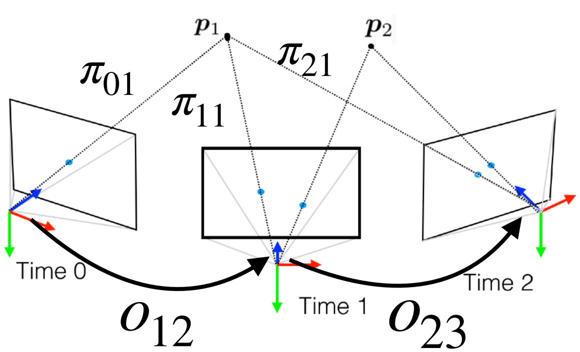
$K?$

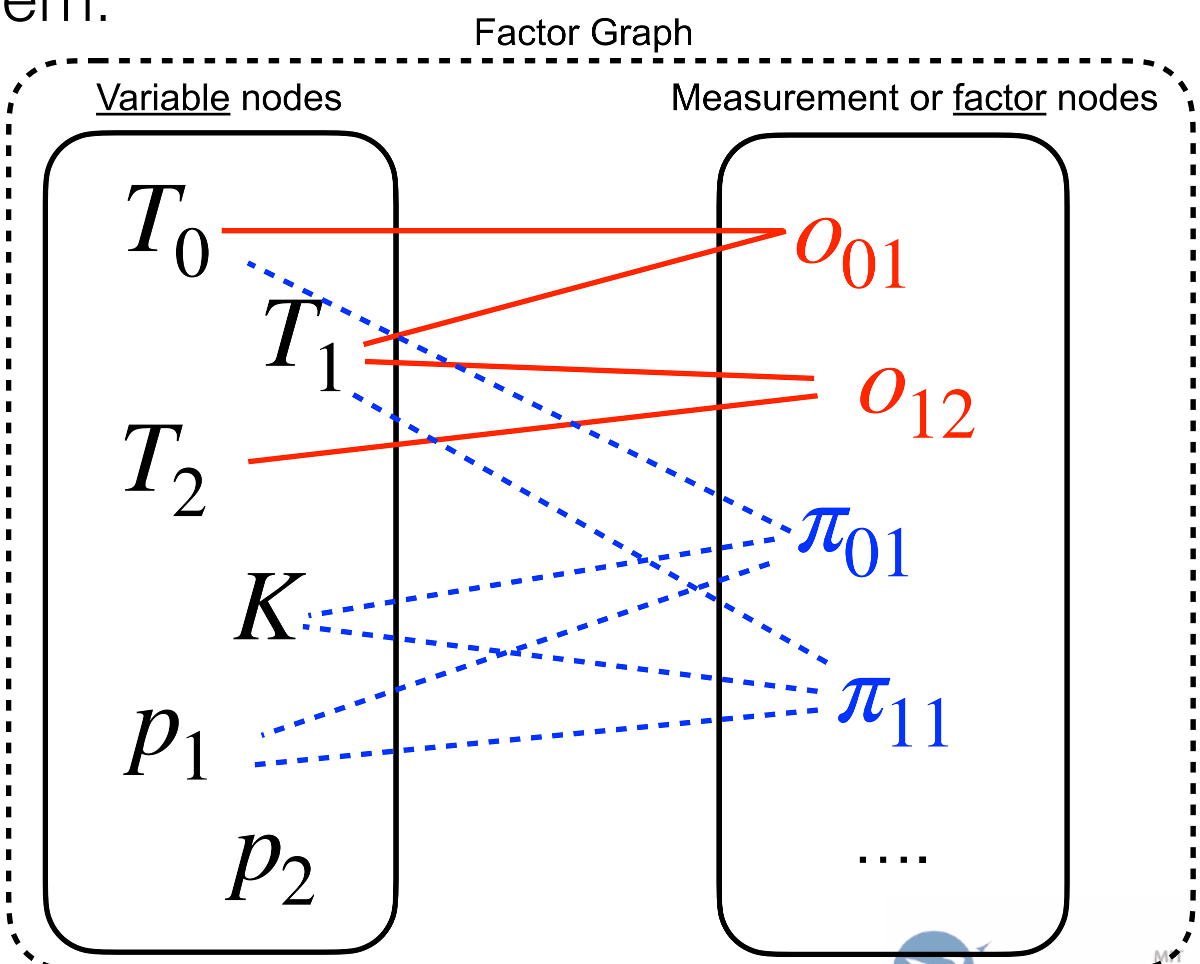


Problem: the projective measurements depend on (i) a pose, (ii) a 3D point, and (iii) the unknown calibration. We can no longer use a standard graph representation where measurements are (pairwise) edges

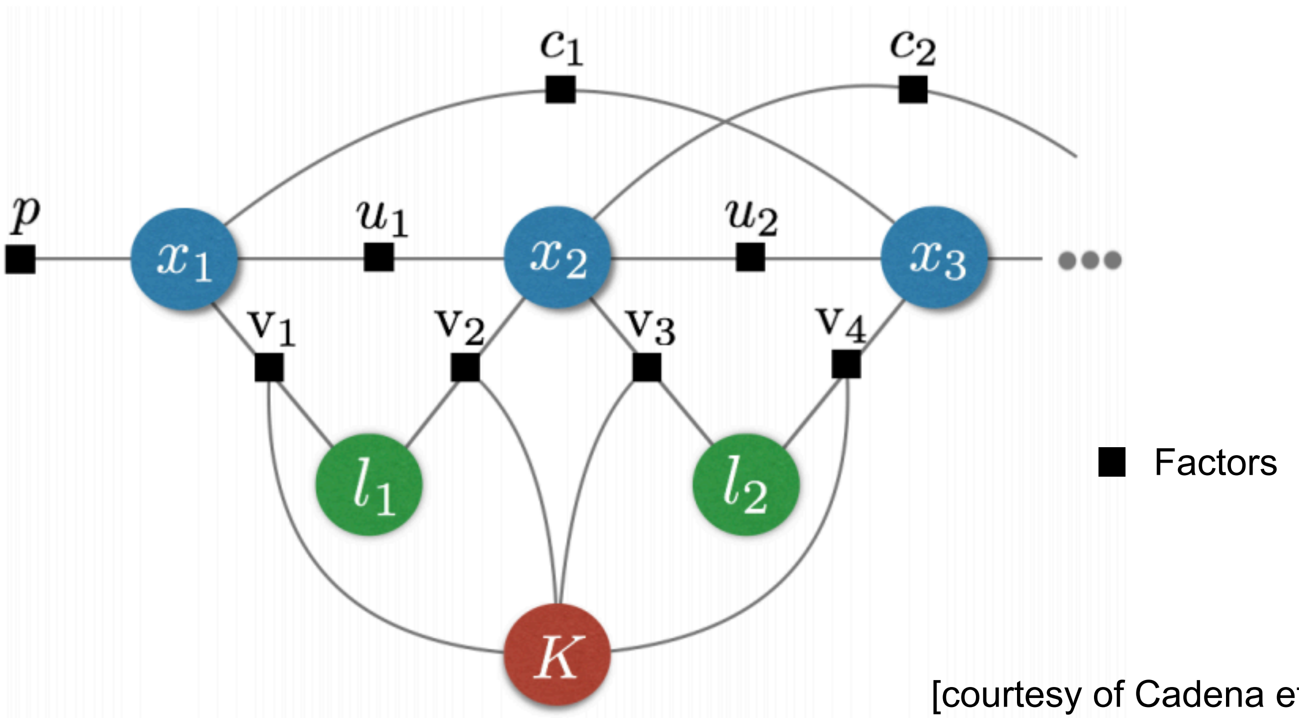
A General Model: Factor Graphs

- Bipartite graph describing measurements and variables in our SLAM problem:





Factor Graph: Example

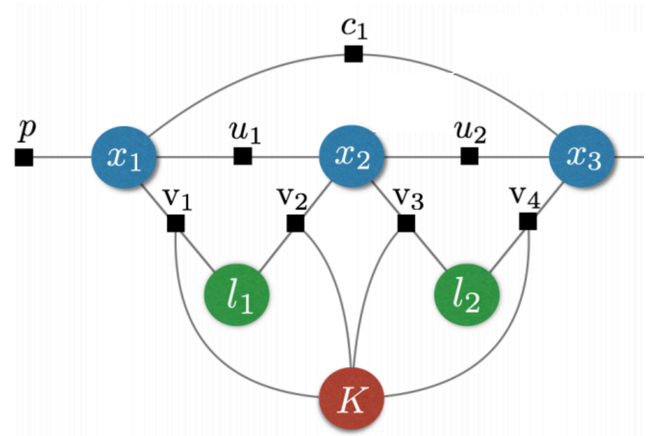


[courtesy of Cadena et al.]

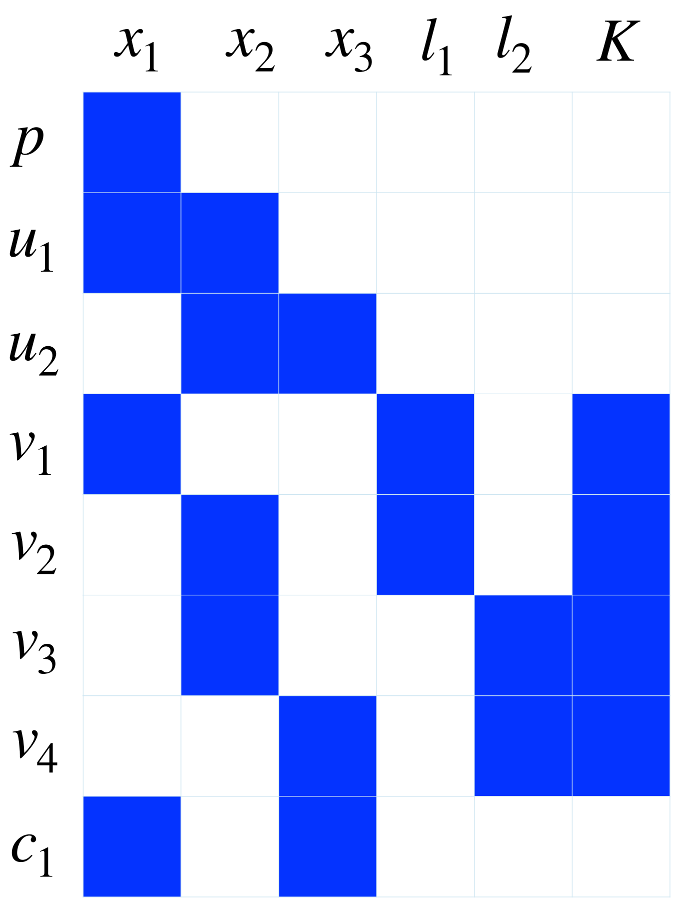
Fig. 3: **SLAM as a factor graph**: Blue circles denote robot poses at consecutive time steps ($x_1, x_2, \ldots$), green circles denote landmark positions ($l_1, l_2, \ldots$), red circle denotes the variable associated with the intrinsic calibration parameters (K). Factors are shown as black squares: the label "u" marks factors corresponding to odometry constraints, "v" marks factors corresponding to camera observations, "c" denotes loop closures, and "p" denotes prior factors.

Factor Graph: Sparsity

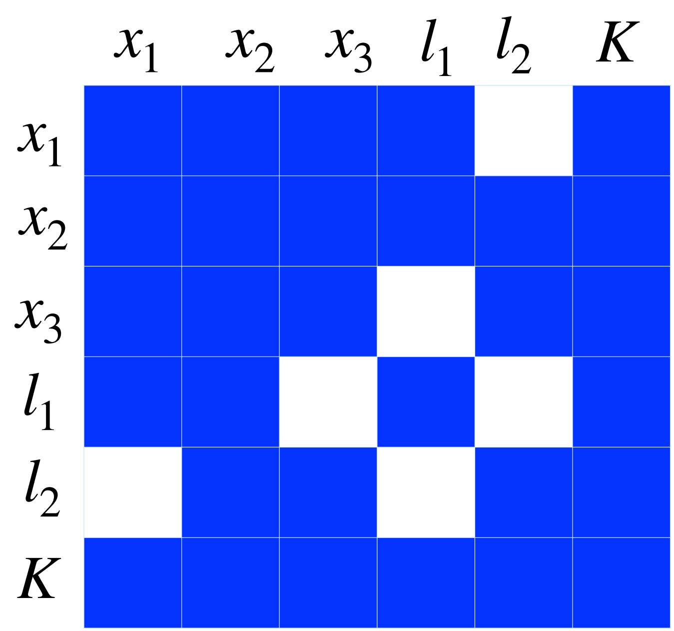
- Sparsity is dictated by topology of the factor graph:



Jacobian **J**



Hessian $J^T\Sigma^{-1}J$



a.k.a. Information Matrix of the estimate

- Normal equations: $(\mathbf{J}^\top\Sigma^{-1}\mathbf{J})\,\mathbf{d} = -\mathbf{J}^\top\Sigma^{-1}\mathbf{r}$

What if we only care about subset of variables?

- Normal equations: $(\mathbf{J}^\top \Sigma^{-1} \mathbf{J})\, \mathbf{d} = -\mathbf{J}^\top \Sigma^{-1} \mathbf{r}$

- What if we only want to compute a subset of variables?
  - $\mathbf{J} = [\mathbf{J}_\mathsf{p} \quad \mathbf{J}_\mathsf{l}]$, i.e., partial derivatives w.r.t. poses and w.r.t. landmarks
  - Information matrix (LHS) blocks

Block structure in the Information Matrix

$$\mathbf{J}^\top \Sigma^{-1} \mathbf{J} = \left[\begin{array}{c|c} \mathbf{J}_\mathsf{p}^\top \Sigma^{-1} \mathbf{J}_\mathsf{p} & \mathbf{J}_\mathsf{p}^\top \Sigma^{-1} \mathbf{J}_\mathsf{l} \\ \hline \mathbf{J}_\mathsf{l}^\top \Sigma^{-1} \mathbf{J}_\mathsf{p} & \mathbf{J}_\mathsf{l}^\top \Sigma^{-1} \mathbf{J}_\mathsf{l} \end{array}\right] =: \left[\begin{array}{c|c} \mathbf{H}_\mathsf{pp} & \mathbf{H}_\mathsf{pl} \\ \hline \mathbf{H}_\mathsf{pl}^\top & \mathbf{H}_\mathsf{ll} \end{array}\right]$$

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{w} \\ \mathbf{z} \end{bmatrix}$$

MIT AEROASTRO

The Schur Complement (Linear Algebra Perspective)

Consider the following linear system with a symmetric coefficient matrix (doesn't have to be symmetric)

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{w} \\ \mathbf{z} \end{bmatrix}$$

If $\mathbf{C}$ is invertible, pre-multiplying LHS/RHS by

$$\begin{bmatrix} \mathbf{I} & -\mathbf{B}\mathbf{C}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$$

(i.e., subtracting $\mathbf{B}\mathbf{C}^{-1} \times$ second equation from the first one) results in

$$\begin{bmatrix} \mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^\top & \mathbf{0} \\ \mathbf{B}^\top & \mathbf{C} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} = \begin{bmatrix} \mathbf{w} - \mathbf{B}\mathbf{D}^{-1}\mathbf{z} \\ \mathbf{z} \end{bmatrix}$$

- Can solve the smaller system $(\mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^\top)\mathbf{x} = \mathbf{w} - \mathbf{B}\mathbf{C}^{-1}\mathbf{z}$ for $\mathbf{x}$
- We have thus *eliminated* $\mathbf{y}$ from the linear system
- If needed, $\mathbf{y}$ can be recovered by back-substituting $\mathbf{x}$
- $\mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^\top$ is called the Schur complement of block $\mathbf{C}$

The Schur Complement Trick in BA / landmark-based SLAM

- Exploit the unique sparsity pattern of the information matrix to solve normal equations efficiently
- Normal equations $(\mathbf{J}^\top \Sigma^{-1} \mathbf{J})\, \mathbf{d} = -\mathbf{J}^\top \Sigma^{-1} \mathbf{r}$ in block form

$$\left[\begin{array}{c|c} \mathbf{H}_{\mathsf{pp}} & \mathbf{H}_{\mathsf{pl}} \\ \hline \mathbf{H}_{\mathsf{pl}}^\top & \mathbf{H}_{\mathsf{ll}} \end{array}\right] \begin{bmatrix} \mathbf{d}_{\mathsf{p}} \\ \mathbf{d}_{\mathsf{l}} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_{\mathsf{p}} \\ \mathbf{b}_{\mathsf{l}} \end{bmatrix}$$

- Schur complement of the map $(\mathbf{H}_{\mathsf{ll}})$ block

$$(\mathbf{H}_{\mathsf{pp}} - \mathbf{H}_{\mathsf{pl}} \mathbf{H}_{\mathsf{ll}}^{-1} \mathbf{H}_{\mathsf{pl}}^\top)\, \mathbf{d}_{\mathsf{p}} = \mathbf{b}_{\mathsf{p}} - \mathbf{H}_{\mathsf{pl}} \mathbf{H}_{\mathsf{ll}}^{-1} \mathbf{b}_{\mathsf{l}}$$

- Schur complement may add non-zero off-diagonal blocks to $\mathbf{H}_{\mathsf{pp}}$
- $\mathbf{H}_{\mathsf{ll}}$ is block-diagonal $\rightarrow$ easy to compute the Schur complement
- # of landmarks $\gg$ # of poses $\rightarrow$ much smaller system
- We can first solve the reduced system for $\mathbf{d}_{\mathsf{p}}$ ising sparse Cholesky/QR
- And then recover $\mathbf{d}_{\mathsf{l}}$ by back-substitution

$$\mathbf{H}_{\mathsf{ll}} \mathbf{d}_{\mathsf{l}} = \mathbf{b}_{\mathsf{l}} - \mathbf{H}_{\mathsf{pl}}^\top \mathbf{d}_{\mathsf{p}}$$

- Once again, $\mathbf{H}_{\mathsf{ll}}$ is block-diagonal $\rightarrow$ easy to solve

The Schur Complement (Probabilistic Perspective)

Review: Canonical Parametrization of Gaussians

$\mathcal{N}(\mu,\Sigma)$ can also be parametrized in terms of

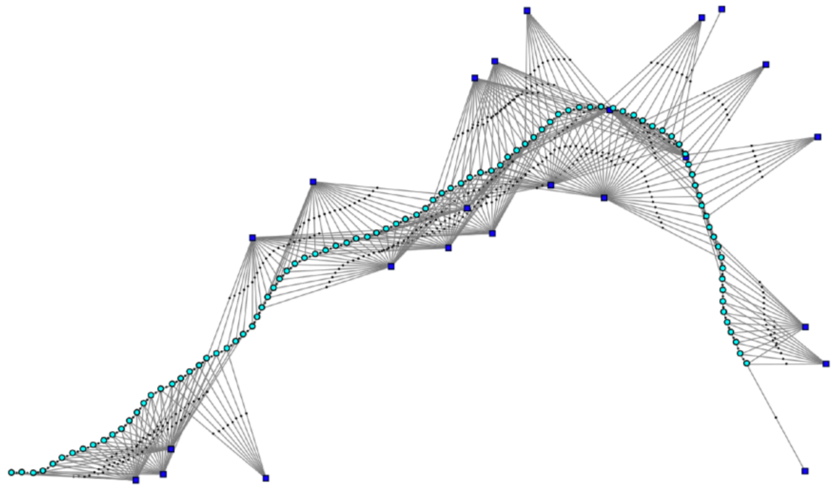
1. Information (precision) matrix $\Lambda \triangleq \Sigma^{-1}$
2. Information vector $\eta \triangleq \Sigma^{-1}\mu$

We write $\mathcal{N}^{-1}(\eta,\Lambda) \equiv \mathcal{N}(\mu,\Sigma)$

- Suppose $p(\mathbf{x},\mathbf{y}) = \mathcal{N}^{-1}\left(\begin{bmatrix}\mathbf{w}\\ \mathbf{z}\end{bmatrix}, \begin{bmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{B}^\top & \mathbf{C}\end{bmatrix}\right)$
- One can marginalize out $\mathbf{y}$ to obtain $p(\mathbf{x}) = \int_{-\infty}^{+\infty} p(\mathbf{x},\mathbf{y})\, d\mathbf{y}$
- Marginal distribution for $p(\mathbf{x}) = \mathcal{N}^{-1}\left(\mathbf{w} - \mathbf{B}\mathbf{C}^{-1}\mathbf{z}, \underline{\mathbf{A} - \mathbf{B}\mathbf{C}^{-1}\mathbf{B}^\top}\right)$

Schur complement

Schur Complement & Marginalization



- Many times we may wish to forget/eliminate unimportant variables (to focus resources on what matters to us, reduce size of linear system, save memory, etc)

- How to eliminate (forget) some variables "without" loss of information?

✗ Naïvely discarding variables and their measurements → loss of information

✓ Proper way: *Marginalize* them out

Schur Complement & Marginalization

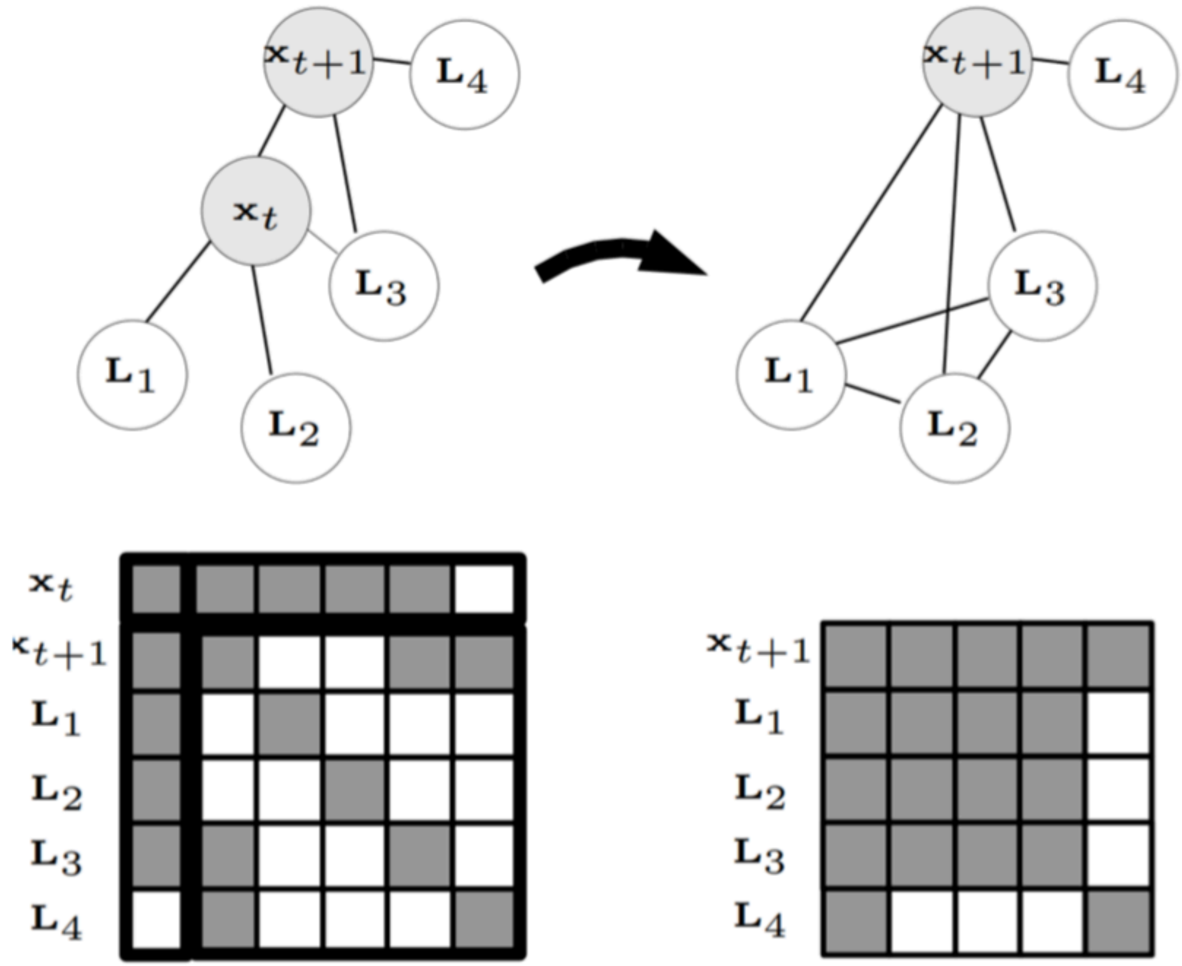
- What does marginalization/Schur complement do to the sparsity pattern of information matrix?

- Eliminating (marginalizing out) a variable creates non-zero off-diagonals (called *fill-in*) in the information matrix between all of its "neighbours" (i.e., those variables that had a non-zero off-diagonal with the eliminated variable in the information matrix)

- In graph terms, elimination creates a clique between the neighbours of the eliminated node

⇒ Loss of sparsity!

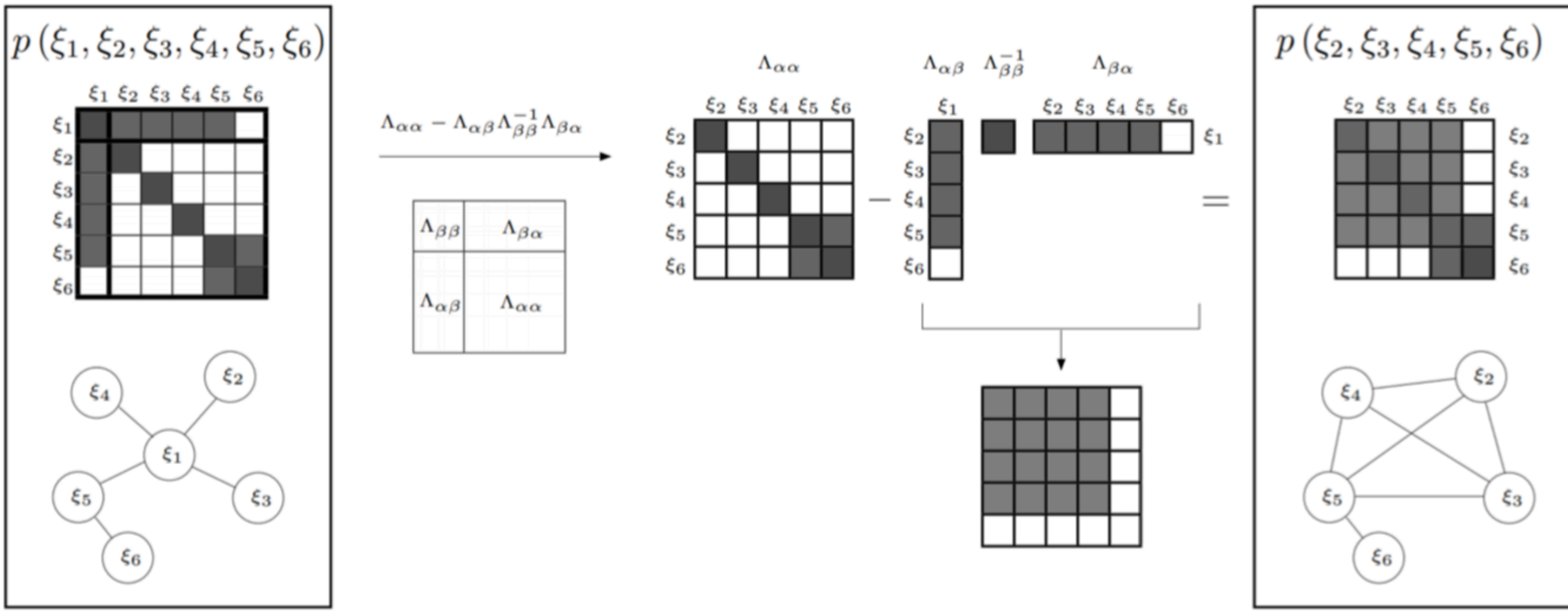
Marginalization: Example 1



Credit: *Eustice et al.*

Marginalization: Example 2

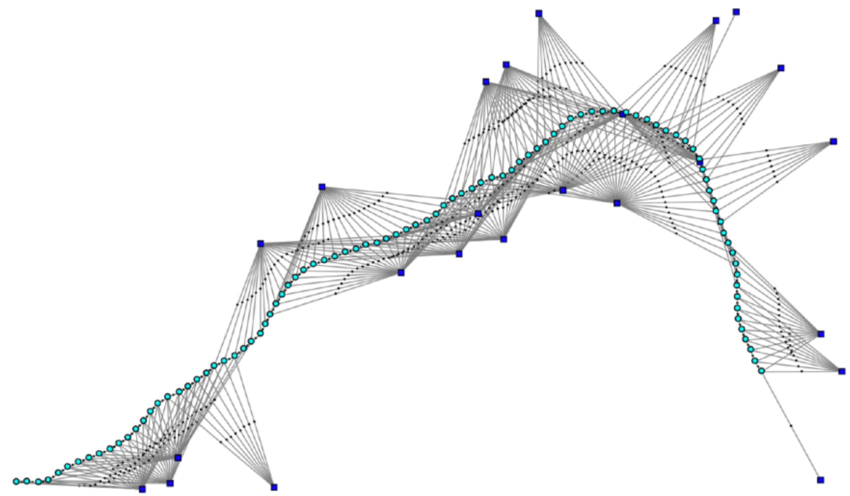
Marginalize ξ_1



Credit: *Walter et al.*

Smoothing and Filtering

MAP or Full smoothing (estimate entire trajectory and map)
- Many variables but
- Information matrix $\mathbf{J}^\top \Sigma^{-1} \mathbf{J}$ is sparse



Fixed-lag smoothing (estimate only variables in a time window)
- Use Schur complement to marginalize out old states (hence less variables)
- Information matrix after Schur complement is denser

Filtering (estimate only current pose and landmarks)
- Use Schur complement to marginalize out ALL old states (hence few variables)
- Information matrix after Schur complement is typically dense

Kalman filter,
Extended Kalman Filter