Bundle Adjustment

Sparse Estimation in Multi-View Geometry

Manmohan Krishna Chandraker

CSE 252C, Fall 2004, UCSD

Overview

- Recap
- Exploiting structure in bundle adjustment
- Partitioned Levenberg-Marquardt
- Solving sparse linear systems
- Banded linear systems

Notation

For an entity $\mathbf{q}$,

- $\mathbf{q}$ - true value of the entity
- $\bar{\mathbf{q}}$ - measured value of the entity
- $\widehat{\mathbf{q}}$ - estimated value of the entity

Gauss-Newton

- Given measurement $\mathbf{X} \in \mathbb{R}^N$ and function f, estimate parameter $\widehat{\mathbf{P}} \in \mathbb{R}^M$ such that

$$\|\epsilon\| = \|f(\widehat{\mathbf{P}}) - \mathbf{X}\|$$

  is minimized.
- Solution :
  - Initial guess : $\mathbf{P}_0$.
  - Assume linearity aroung $\mathbf{P}_0$.
  - $\epsilon_0 = \mathbf{X} - f(\mathbf{P}_0)$.
  - Find $\mathbf{P}_1 = \mathbf{P}_0 + \boldsymbol{\Delta}$ to minimize $\|\mathbf{X} - f(\mathbf{P}_1)\| = \|\mathbf{X} - f(\mathbf{P}_0) - \mathbf{J}\boldsymbol{\Delta}\| = \|\epsilon_0 - \mathbf{J}\boldsymbol{\Delta}\|$.

Jacobian

- $\mathbf{J} = \dfrac{\partial \widehat{\mathbf{X}}}{\partial \mathbf{P}}$ evaluated at $\widehat{\mathrm{P}}$.

$$\mathbf{J} = \begin{bmatrix} \frac{\partial \widehat{\mathbf{X}}_1}{\partial \mathbf{P}_1} & \frac{\partial \widehat{\mathbf{X}}_1}{\partial \mathbf{P}_2} & \cdots & \frac{\partial \widehat{\mathbf{X}}_1}{\partial \mathbf{P}_M} \\ \vdots & \ddots & & \vdots \\ \frac{\partial \widehat{\mathbf{X}}_N}{\partial \mathbf{P}_1} & \frac{\partial \widehat{\mathbf{X}}_N}{\partial \mathbf{P}_2} & \cdots & \frac{\partial \widehat{\mathbf{X}}_N}{\partial \mathbf{P}_M} \end{bmatrix}_{N \times M}$$

- Update equations:

$$\mathbf{J}\boldsymbol{\Delta} = \epsilon$$

- Normal equations:

$$\mathbf{J}^\top \mathbf{J}\boldsymbol{\Delta} = \mathbf{J}^\top \epsilon$$

Levenberg-Marquardt

- Augmented normal equations :

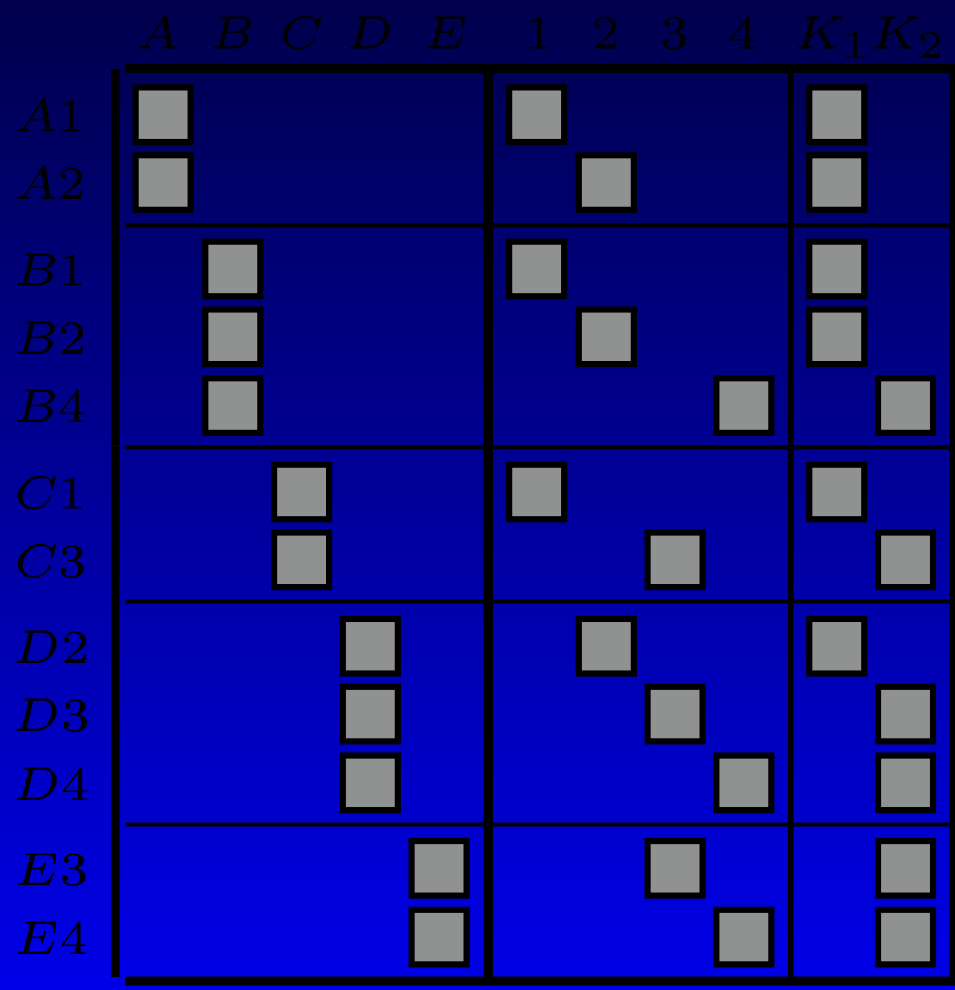
$$(\mathbf{J}^\top \mathbf{J} + \lambda \mathbf{I})\boldsymbol{\Delta} = \mathbf{J}^\top \epsilon$$

- $\mathcal{O}(n^3)$ in number of parameters

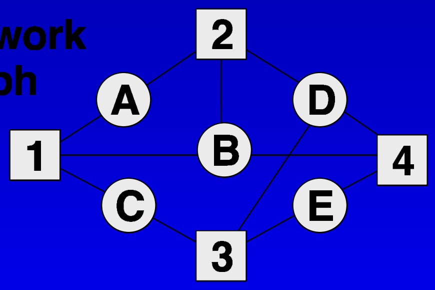
- Homography computation
  - Noiseless case
  - Noisy correspondences
- Bundle adjustment

Bundle Adjustment : Jacobian

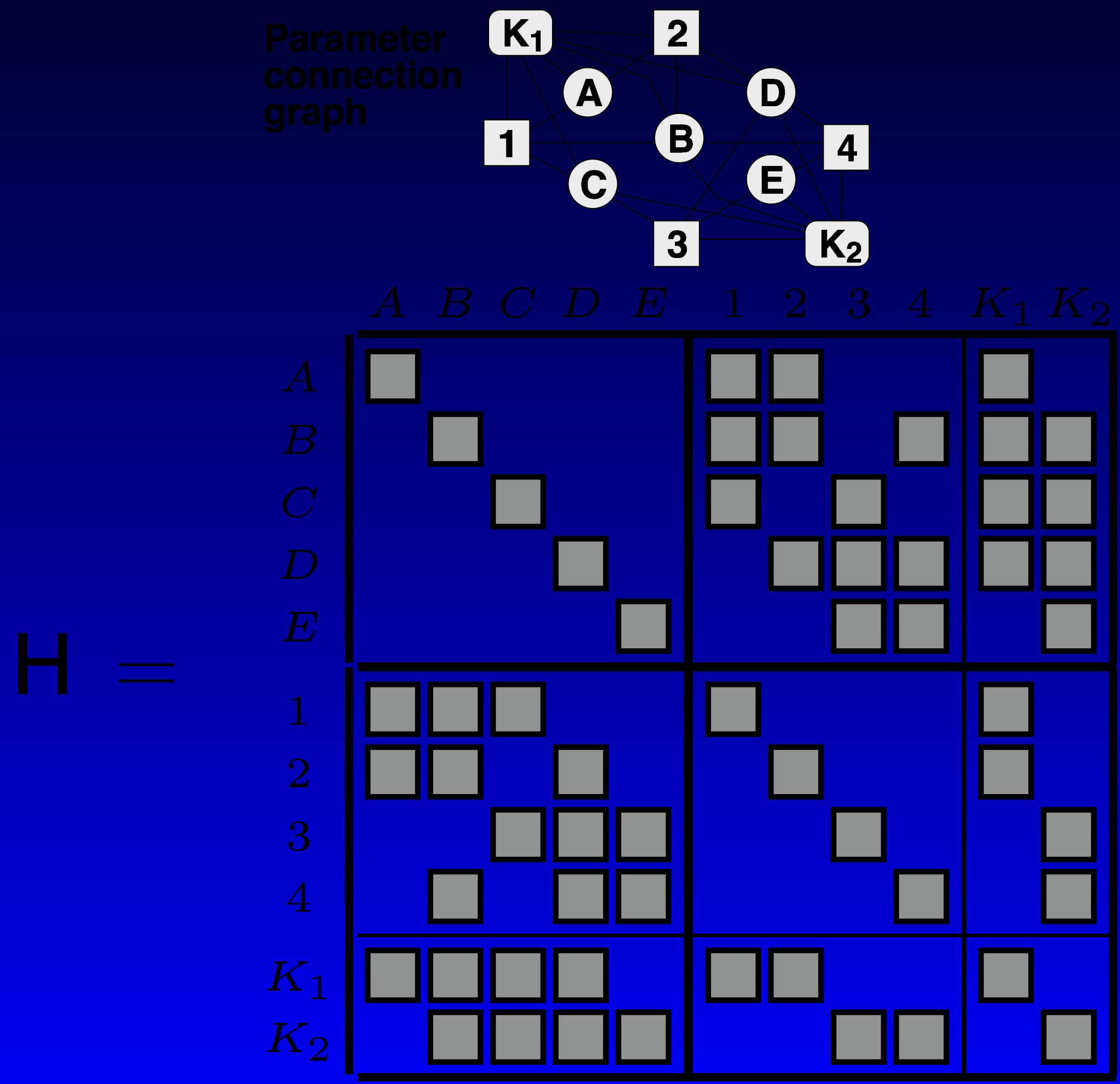
$\mathbf{J} =$



Network graph



Primary & Secondary Structure



Exploiting Structure

Observations:

- Hessian coarsely divided into 4 blocks

$$\mathbf{H} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}.$$

- Hessian is symmetric.
- Blocks $\mathbf{A}$ and $\mathbf{D}$ are sparse and block diagonal.

Exploiting Partitioning

- Schur's complement.
- Reduced bundle system.

Exploiting symmetry

- A square, non-singular matrix $\mathbf{A}$ can be factorized as $\mathbf{LDU}$ where $\mathbf{L}$ is lower triangular with unit diagonal entries, $\mathbf{U}$ is upper triangular and $\mathbf{D}$ is a diagonal matrix.
- If $\mathbf{A}$ is also symmetric, $\mathbf{U} = \mathbf{L}^\top$.
- If $\mathbf{A}$ is also positive definite, $\mathbf{D} = \mathbf{I}$.

Symmetric System Solver

Given $\mathbf{A}\mathbf{x} = \mathbf{L}\mathbf{D}\mathbf{L}^\top\mathbf{x} = \mathbf{b}$, solve for $\mathbf{x}$

- *Forward substitution:* Solve $\mathbf{L}\mathbf{x}' = \mathbf{b}$ in order for components of $\mathbf{x}'$

$$\mathbf{x}'_i = \mathbf{b}_i - \sum_{j=1}^{i-1} L_{ij}\mathbf{x}'_j.$$

- *Scaling:* Solve $\mathbf{D}\mathbf{x}'' = \mathbf{x}'$.
- *Back-substitution:* Solve $\mathbf{L}^\top\mathbf{x} = \mathbf{x}''$ in order for components of $\mathbf{x}$

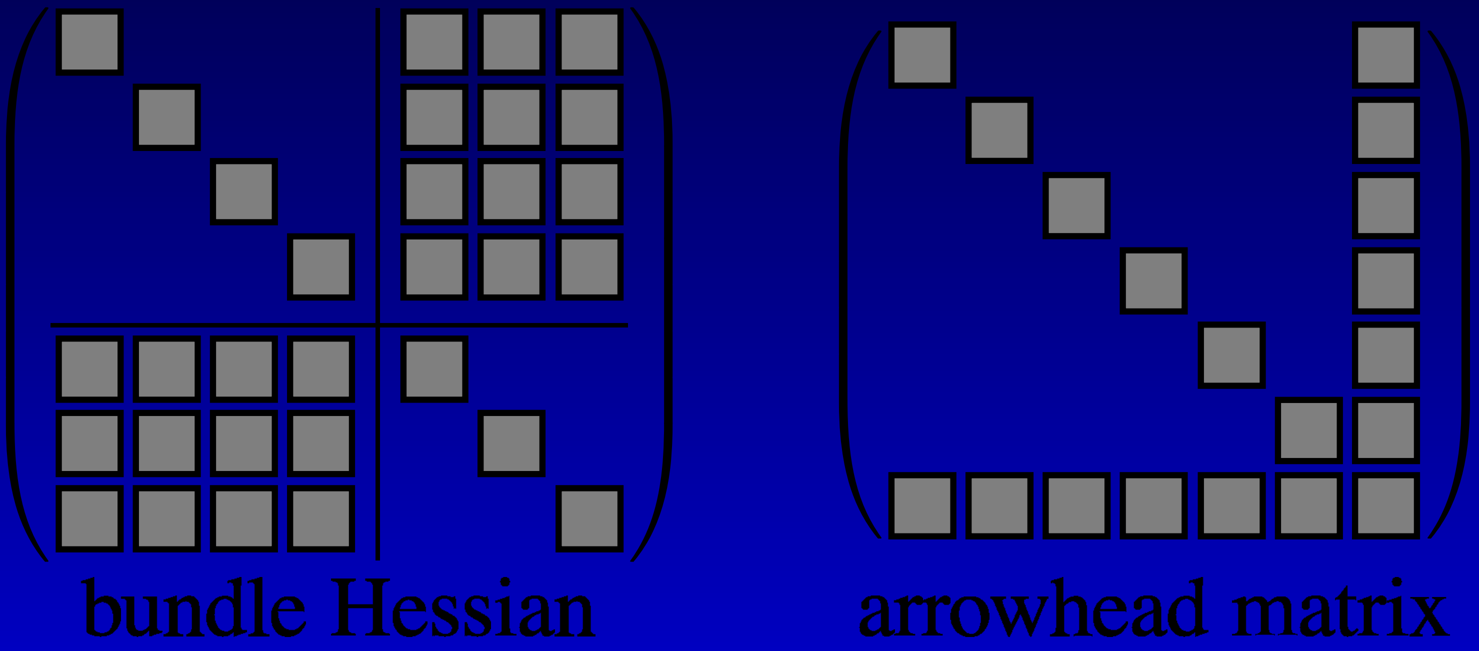
$$\mathbf{x}_i = \mathbf{x}''_i - \sum_{j=i+1}^{n} L_{ji}\mathbf{x}_j.$$

Sparse Factorization

- Arrowhead matrices.
- Block tridiagonal systems.
- Divide and conquer - recursive partitioning.

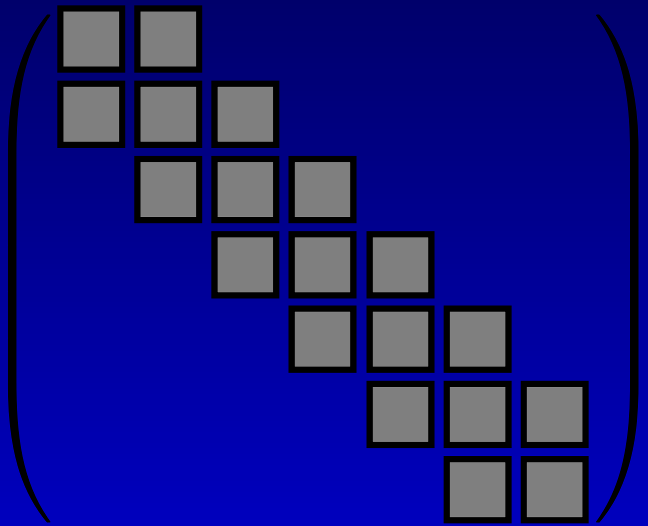
Arrowhead Matrices

Trivial LDL-decomposition.



Block Tridiagonal Systems

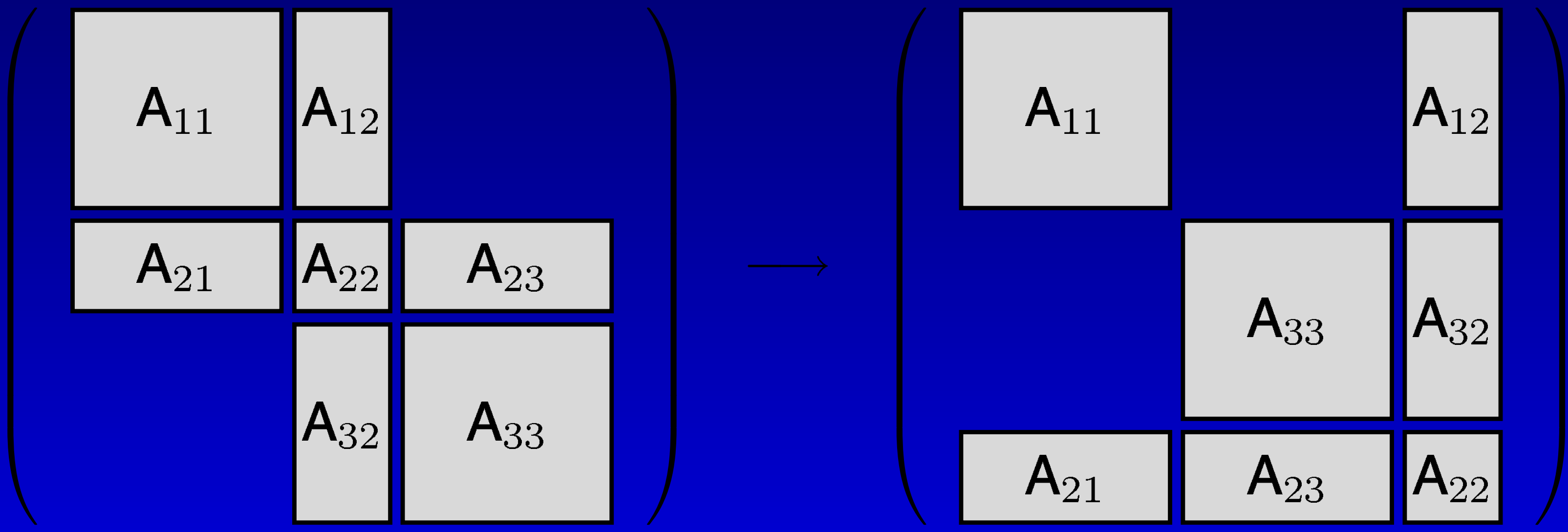
- **L** and **U** factors are also block tridiagonal.
- Reduced system obtained by recursive 2×2 Schur complementation.



block tridiagonal matrix

Recursive Partitioning

- Construct elimination graph
- Find a separating vertex cut
- Re-order into connected components, separating ones last.



Partitioned L-M

- Parameter vector, $\mathbf{P} = \left[\mathbf{a}^\top, \mathbf{b}^\top\right]^\top \in \mathbb{R}^M$
- Measurement vector, $\mathbf{X} \in \mathbb{R}^N$ (given).

- **Objective :** Find $\mathbf{P}$ that minimizes squared Mahalanobis distance,

$$\|\epsilon\|^2_{\Sigma_{\mathbf{X}}} = \|\epsilon^\top \Sigma_{\mathbf{X}}^{-1} \epsilon\|$$

where $\epsilon = \mathbf{X} - \widehat{\mathbf{X}}$.

Partitioned Jacobian

- Jacobian sub-matrices

$$\mathbf{A} = \left[\partial\widehat{\mathbf{X}}/\partial\mathbf{a}\right] \qquad \mathbf{B} = \left[\partial\widehat{\mathbf{X}}/\partial\mathbf{b}\right]$$

- Update equation

$$\mathbf{J}\boldsymbol{\Delta} = [\mathbf{A}|\mathbf{B}]\begin{pmatrix} \boldsymbol{\Delta}_{\mathbf{a}} \\ \boldsymbol{\Delta}_{\mathbf{b}} \end{pmatrix} = \epsilon$$

- Normal equations

$$\mathbf{J}^\top\mathbf{J}\boldsymbol{\Delta} = \mathbf{J}^\top\epsilon$$

Normal Equations

- Normal equations

$$\left[\begin{array}{c|c} \mathbf{A}^\top\mathbf{A} & \mathbf{A}^\top\mathbf{B} \\ \hline \mathbf{B}^\top\mathbf{A} & \mathbf{B}^\top\mathbf{B} \end{array}\right] \begin{pmatrix} \boldsymbol{\Delta}_\mathbf{a} \\ \boldsymbol{\Delta}_\mathbf{b} \end{pmatrix} = \begin{pmatrix} \mathbf{A}^\top\epsilon \\ \mathbf{B}^\top\epsilon \end{pmatrix}$$

- Augmented normal equations

$$\begin{bmatrix} \mathbf{U}^* & \mathbf{W} \\ \mathbf{W}^\top & \mathbf{V}^* \end{bmatrix} \begin{pmatrix} \boldsymbol{\Delta}_\mathbf{a} \\ \boldsymbol{\Delta}_\mathbf{b} \end{pmatrix} = \begin{pmatrix} \epsilon_\mathbf{A} \\ \epsilon_\mathbf{B} \end{pmatrix}$$

Solution

- Pre-multiply by $\begin{bmatrix} \mathbf{I} & -\mathbf{W}{\mathbf{V}^*}^{-1} \\ \mathbf{0} & \mathbf{I} \end{bmatrix}$ to get

$$\begin{bmatrix} \mathbf{U}^* - \mathbf{W}{\mathbf{V}^*}^{-1}\mathbf{W}^\top & \mathbf{0} \\ \mathbf{W}^\top & \mathbf{V}^* \end{bmatrix} \begin{pmatrix} \boldsymbol{\Delta}_\mathbf{a} \\ \boldsymbol{\Delta}_\mathbf{b} \end{pmatrix} = \begin{pmatrix} \epsilon_\mathbf{A} - \mathbf{W}{\mathbf{V}^*}^{-1}\epsilon_\mathbf{B} \\ \epsilon_\mathbf{B} \end{pmatrix}$$

- Solve for $\boldsymbol{\Delta}_\mathbf{a}$:

$$(\mathbf{U}^* - \mathbf{W}{\mathbf{V}^*}^{-1}\mathbf{W}^\top)\boldsymbol{\Delta}_\mathbf{a} = \epsilon_\mathbf{A} - \mathbf{W}{\mathbf{V}^*}^{-1}\epsilon_\mathbf{B}$$

- Back-substitute for $\boldsymbol{\Delta}_\mathbf{b}$:

$$\mathbf{V}^*\boldsymbol{\Delta}_\mathbf{b} = \epsilon_\mathbf{B} - \mathbf{W}^\top\boldsymbol{\Delta}_\mathbf{a}$$

Update

- $\mathbf{P}_{\text{new}} = \mathbf{P} + \left(\boldsymbol{\Delta}_{\mathbf{a}}^{\top}, \boldsymbol{\Delta}_{\mathbf{b}}^{\top}\right)^{\top}$.
- $\epsilon_{\text{new}} = f(\mathbf{P}_{\text{new}}) - \mathbf{X}$.
- If $\|\epsilon_{\text{new}}\| \leq \|\epsilon\|$:

$$\lambda \leftarrow \lambda / 10$$

  else:

$$\lambda \leftarrow \lambda * 10$$

- Re-iterate until convergence.

Sparse Levenberg-Marquardt

- Measurement vector, $\mathbf{X} = (\mathbf{X}_1^\top, \cdots, \mathbf{X}_n^\top)^\top \in \mathbb{R}^N$

- Independent measurements $\Rightarrow$ block-diagonal covariance matrix

$$\Sigma_{\mathbf{X}} = \begin{bmatrix} \Sigma_{\mathbf{X}_1} & & & \\ & \Sigma_{\mathbf{X}_2} & & \\ & & \ddots & \\ & & & \Sigma_{\mathbf{X}_n} \end{bmatrix}$$

Sparseness

- Parameter vector,
$\mathbf{P} = (\mathbf{a}^\top, \mathbf{b}_1^\top, \cdots, \mathbf{b}_n^\top)^\top \in \mathbb{R}^M$.

- *Sparseness assumption* : $\widehat{\mathbf{X}}_i$ depends on $\mathbf{a}$ and $\mathbf{b}_i$ only

$$\frac{\partial \widehat{\mathbf{X}}_i}{\partial \mathbf{b}_j} = 0, \text{ for } i \neq j$$

Sparse Jacobian

- Jacobian sub-matrices

$$\mathbf{A}_i = \left[\partial\widehat{\mathbf{X}}_i/\partial\mathbf{a}\right] \qquad \mathbf{B}_i = \left[\partial\widehat{\mathbf{X}}_i/\partial\mathbf{b}_i\right]$$

- Error vector : $\epsilon = \mathbf{X} - \widehat{\mathbf{X}} = (\epsilon_1^\top, \cdots, \epsilon_n^\top)^\top$.
- Update equations :

$$\mathbf{J}\boldsymbol{\Delta} = \left[\begin{array}{c|cccc} \mathbf{A}_1 & \mathbf{B}_1 & & & \\ \mathbf{A}_2 & & \mathbf{B}_2 & & \\ \vdots & & & \ddots & \\ \mathbf{A}_n & & & & \mathbf{B}_n \end{array}\right] \left(\begin{array}{c} \boldsymbol{\Delta}_{\mathbf{a}} \\ \hline \boldsymbol{\Delta}_{\mathbf{b}_1} \\ \vdots \\ \boldsymbol{\Delta}_{\mathbf{b}_n} \end{array}\right) = \begin{pmatrix} \epsilon_1 \\ \vdots \\ \epsilon_n \end{pmatrix}$$

Solution

$$\mathbf{U} = \sum_i \mathbf{A}_i^\top \mathbf{A}_i$$

$$\mathbf{V} = \mathrm{diag}(\mathbf{V}_1, \cdots, \mathbf{V}_n), \quad \text{where } \mathbf{V}_i = \mathbf{B}_i^\top \mathbf{B}_i$$

$$\mathbf{W} = [\mathbf{W}_1, \cdots, \mathbf{W}_n], \quad \text{where } \mathbf{W}_i = \mathbf{A}_i^\top \mathbf{B}_i$$

$$\epsilon_{\mathbf{B}} = (\epsilon_{\mathbf{B}_1}^\top, \cdots, \epsilon_{\mathbf{B}_n}^\top)^\top, \quad \text{where } \epsilon_{\mathbf{B}_i} = \mathbf{B}_i^\top \epsilon_i$$

$$\epsilon_{\mathbf{A}} = \sum_i \mathbf{A}_i^\top \epsilon_i \qquad \mathbf{Y}_i = \mathbf{W}_i {\mathbf{V}_i^*}^{-1}$$

- Compute $\mathbf{\Delta_a}$:
  $(\mathbf{U}^* - \sum_i \mathbf{Y}_i \mathbf{W}_i^\top)\mathbf{\Delta_a} = \epsilon_{\mathbf{A}} - \sum_i \mathbf{Y}_i \epsilon_{\mathbf{B}_i}$.
- Compute $\mathbf{\Delta_b}$: $\mathbf{\Delta}_{\mathbf{b}_i} = {\mathbf{V}_i^*}^{-1}(\epsilon_{\mathbf{B}_1} - \mathbf{W}_1^\top \mathbf{\Delta_a})$.

Sparseness Advantage

The matrix $\mathbf{V}^*$ is block diagonal, so

- Inversion is easy.
- The update equations can be solved one block at a time.
- Each step of the algorithm is linear, so total complexity is $\mathcal{O}(n)$.

2D Homography Estimation

- **Measurements:** $(\mathbf{x}_i, \mathbf{x}'_i)$
- Measurements are noisy versions of true points $(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}'_i)$.

- **Objective:** Minimize

$$\sum_i d(\mathbf{x}_i, \bar{\mathbf{x}}_i)^2 + d(\mathbf{x}'{}_i, \mathbf{H}\bar{\mathbf{x}}_i)^2$$

w. r. t. parameter vector $\mathbf{P} = (\mathbf{h}, \bar{\mathbf{x}}_1, \cdots, \bar{\mathbf{x}}_n)^\top$.

Sparsity Structure

- Estimated measurement vector,

$$f(\mathbf{P}) = (\widehat{\mathbf{X}}_1^\top, \widehat{\mathbf{X}}_2^\top, \cdots, \widehat{\mathbf{X}}_n^\top)^\top$$

- *Sparseness assumption:* $\widehat{\mathbf{X}}_i$ depends only on $\mathbf{H}$ and $\widehat{\mathbf{x}}_i$ and is independent of $\widehat{\mathbf{x}}_j$ for $j \neq i$.
- Sparse Jacobian

$$\mathbf{J}\boldsymbol{\Delta} = \left[\begin{array}{c|cccc} \mathbf{A}_1 & \mathbf{B}_1 & & & \\ \mathbf{A}_2 & & \mathbf{B}_2 & & \\ \vdots & & & \ddots & \\ \mathbf{A}_n & & & & \mathbf{B}_n \end{array}\right] \begin{pmatrix} \boldsymbol{\Delta}_{\mathbf{a}} \\ \hline \boldsymbol{\Delta}_{\mathbf{b}_1} \\ \vdots \\ \boldsymbol{\Delta}_{\mathbf{b}_n} \end{pmatrix}$$

More ....

- $\widehat{\mathbf{X}}_i = (\widehat{\mathbf{x}}_i^\top, \mathbf{H}\widehat{\mathbf{x}}_i^\top)^\top = (\widehat{\mathbf{x}}_i^\top, \widehat{\mathbf{x}'}_i^\top)^\top$
- $\widehat{\mathbf{x}}_i$ is independent of $\mathbf{h}$:

$$\mathbf{A}_i = \frac{\partial \widehat{\mathbf{X}}_i}{\partial \mathbf{h}} = \begin{bmatrix} \mathbf{0} \\ \partial \widehat{\mathbf{x}'}_i^\top / \partial \mathbf{h} \end{bmatrix}$$

- Similarly,

$$\mathbf{B}_i = \frac{\partial \widehat{\mathbf{X}}_i}{\partial \widehat{\mathbf{x}}_i} = \begin{bmatrix} \mathbf{I} \\ \partial \widehat{\mathbf{x}'}_i^\top / \partial \widehat{\mathbf{x}}_i \end{bmatrix}$$

Bundle Adjustment

- m cameras, n points.
- $\mathbf{x}_{ij}$: image of i-th point by j-th camera.
- **Measurement:** $\mathbf{X} = (\mathbf{X}_1, \cdots, \mathbf{X}_n)^\top$ where $\mathbf{X}_i = (\mathbf{x}_{i1}^\top, \mathbf{x}_{i2}^\top, \cdots, \mathbf{x}_{im}^\top)^\top$.
- **Parameter vector:** $(\mathbf{a}^\top, \mathbf{b}^\top)^\top$
  - Camera parameters, $\mathbf{a} = (\mathbf{a}_1^\top, \cdots, \mathbf{a}_m^\top)^\top$.
  - 3D point parameters, $\mathbf{b} = (\mathbf{b}_1^\top, \cdots, \mathbf{b}_n^\top)^\top$.
- **Objective:** Minimize total reprojection error

Sparsity Structure

- $\mathbf{x}_{ij}$ depends only on j-th camera

$$\frac{\partial \widehat{\mathbf{x}}_{ij}}{\partial \mathbf{a}_k} = 0 \text{ for } j \neq k$$

- Jacobian submatrix

$$\mathbf{A}_i = \left[\partial \widehat{\mathbf{X}}_i / \partial \mathbf{a}\right] = \mathrm{diag}(\mathbf{A}_{i1}, \cdots, \mathbf{A}_{im})$$

where $\mathbf{A}_{ij} = \partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{a}_j$.

Sparsity Structure (contd.)

- Image $\mathbf{x}_{ij}$ depends only on i-th 3D point, $\mathbf{X}_i$

$$\frac{\partial \widehat{\mathbf{x}}_{ij}}{\partial \mathbf{b}_k} = 0 \text{ for } j \neq k$$

- Jacobian submatrix

$$\mathbf{B}_i = \left[\partial \widehat{\mathbf{X}}_i / \partial \mathbf{b}_i\right] = \left[\mathbf{B}_{i1}^\top, \cdots, \mathbf{B}_{im}^\top\right]^\top$$

where $\mathbf{B}_{ij} = \partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{b}_i$.

Sparsity Structure (contd.)

- Completely uncorrelated measurements :
  - $\Sigma_{\mathbf{X}} = \mathrm{diag}(\Sigma_{\mathbf{X}_1}, \cdots, \Sigma_{\mathbf{X}_n})$.
  - $\Sigma_{\mathbf{X}_i} = \mathrm{diag}(\Sigma_{\mathbf{x}_{i1}}, \cdots, \Sigma_{\mathbf{x}_{im}})$.

Sparse Jacobian

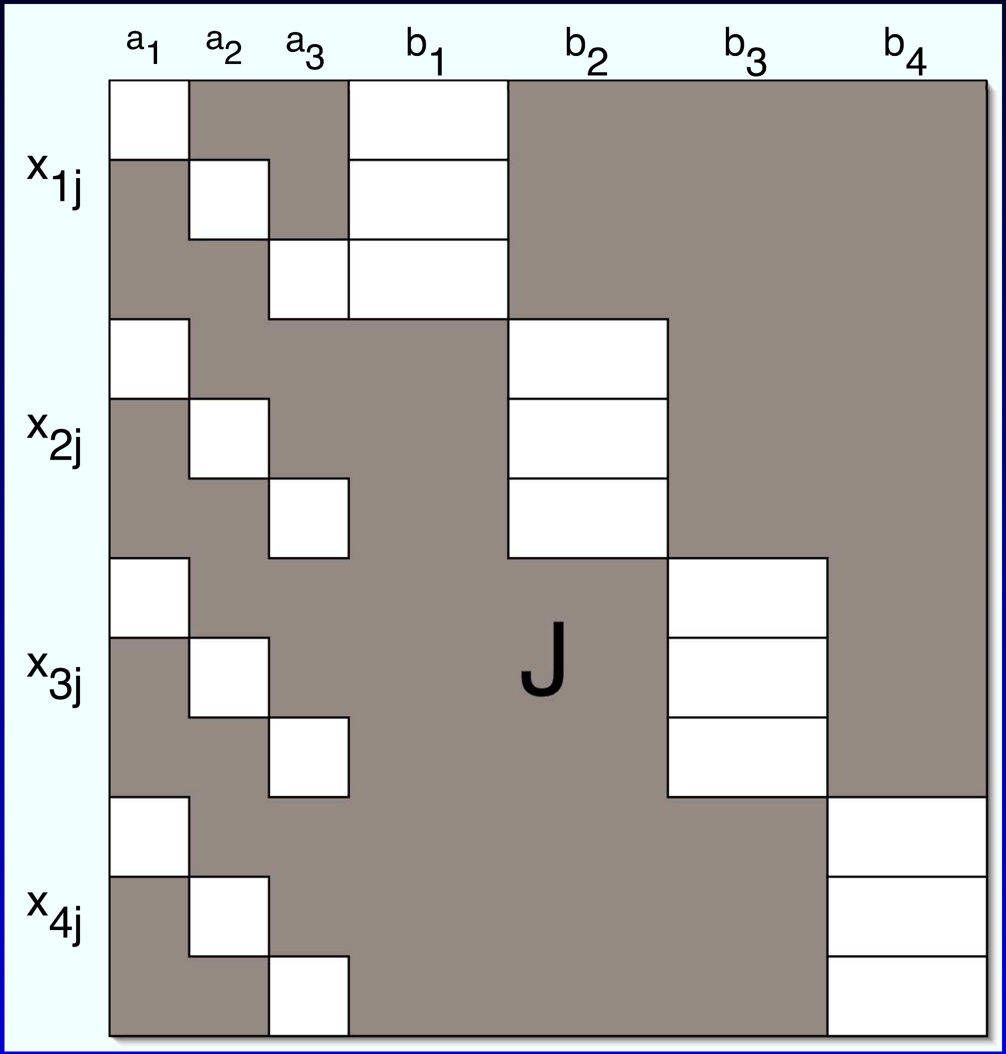


Figure 1: Bundle adjustment Jacobian: 3 cameras, 4 features

Sparse Hessian

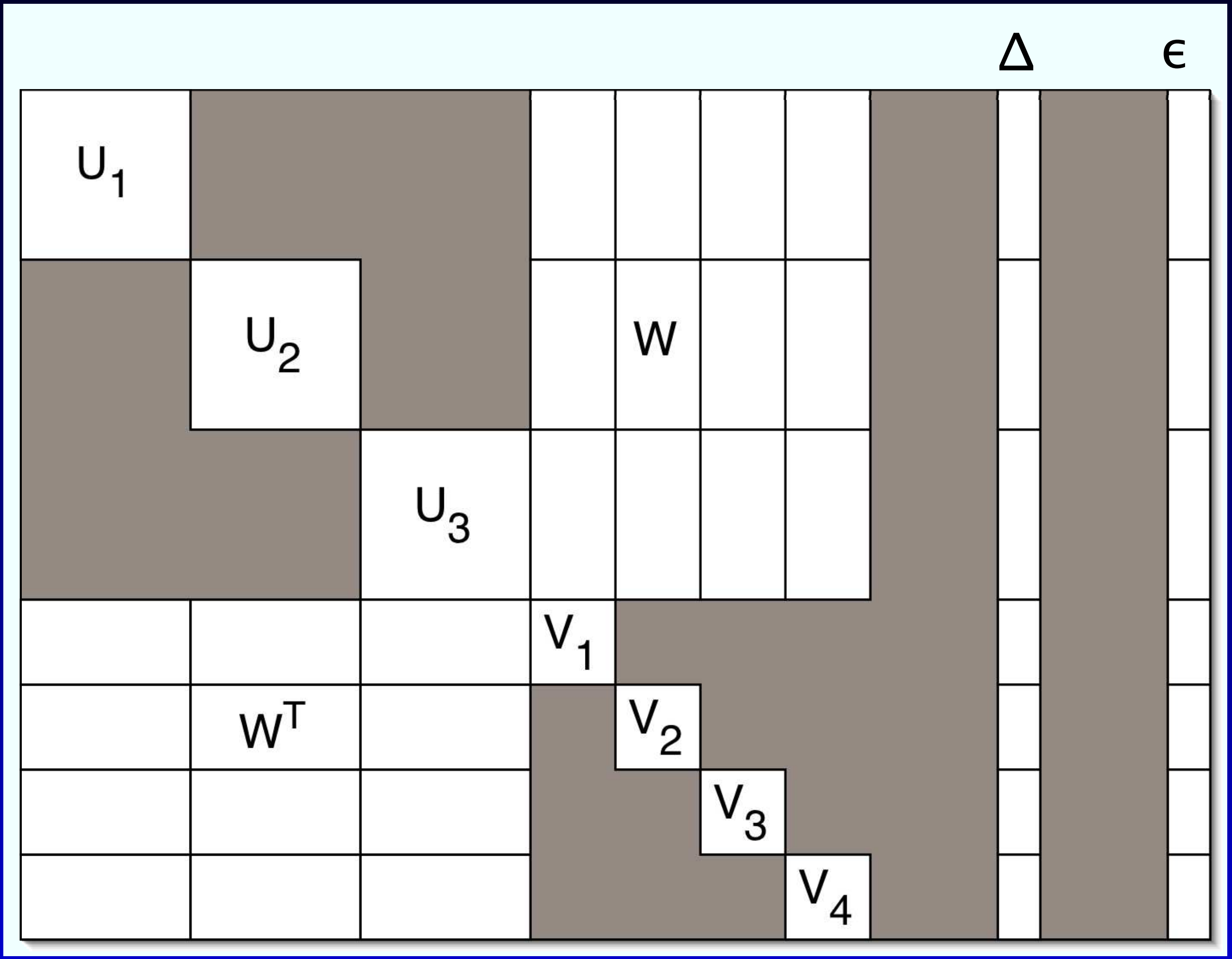


Figure 2: Bundle adjustment Hessian: 3 cameras and 4 features

Algorithm: Bundle Adjustment

- Compute error vector $\epsilon_{ij} = \mathbf{x}_{ij} - \widehat{\mathbf{x}}_{ij}$ and derivative matrices

$$\mathbf{A}_{ij} = [\partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{a}_j] \quad \mathbf{B}_{ij} = [\partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{b}_j]$$

- Compute, for $i = 1, \cdots, n$ and $j = 1, \cdots, m$:

$$\mathbf{U}_j = \sum_i \mathbf{A}_{ij}^\top \mathbf{A}_{ij} \qquad \mathbf{V}_i = \sum_i \mathbf{B}_{ij}^\top \mathbf{B}_{ij}$$

$$\epsilon_{\mathbf{a}_j} = \sum_i \mathbf{A}_{ij}^\top \epsilon_{ij} \qquad \epsilon_{\mathbf{b}_i} = \sum_i \mathbf{B}_{ij}^\top \epsilon_{ij}$$

$$\mathbf{W}_{ij} = \mathbf{A}_{ij}^\top \mathbf{B}_{ij} \qquad \mathbf{Y}_{ij} = \mathbf{W}_{ij} {\mathbf{V}_i^*}^{-1}$$

Algorithm (contd.)

- Compute $\mathbf{\Delta_a} = (\mathbf{\Delta}_{\mathbf{a}_1}^\top, \cdots, \mathbf{\Delta}_{\mathbf{a}_m}^\top)^\top$ from

$$\mathbf{S}\mathbf{\Delta_a} = (\epsilon_1^\top, \cdots, \epsilon_m^\top)^\top$$

  S is $m \times m$ block matrix, with block

$$\mathbf{S}_{jk} = -\sum_i \mathbf{Y}_{ij}\mathbf{W}_{ik}^\top + \mathbf{U}_j^* \delta_{jk}$$

  $\epsilon_j = \epsilon_{\mathbf{a}_j} - \sum_i \mathbf{Y}_{ij}\epsilon_{\mathbf{b}_i}$.

- Back-substitute for $\mathbf{\Delta_b} = (\mathbf{\Delta}_{\mathbf{b}_1}^\top, \cdots, \mathbf{\Delta}_{\mathbf{b}_n}^\top)^\top$

$$\mathbf{\Delta}_{\mathbf{b}_i} = \mathbf{V}_i^{*-1}(\epsilon_{\mathbf{b}_i} - \sum_j \mathbf{W}_{ij}^\top \mathbf{\Delta}_{\mathbf{a}_j})$$

Missing Data

- Each point is visible only in some arbitrary subset of views.

- Suppose i-th point not visible in j-th image ($\mathbf{x}_{ij}$ not defined).
- Simply ignore terms subscripted by ij.

Sparse Tracks

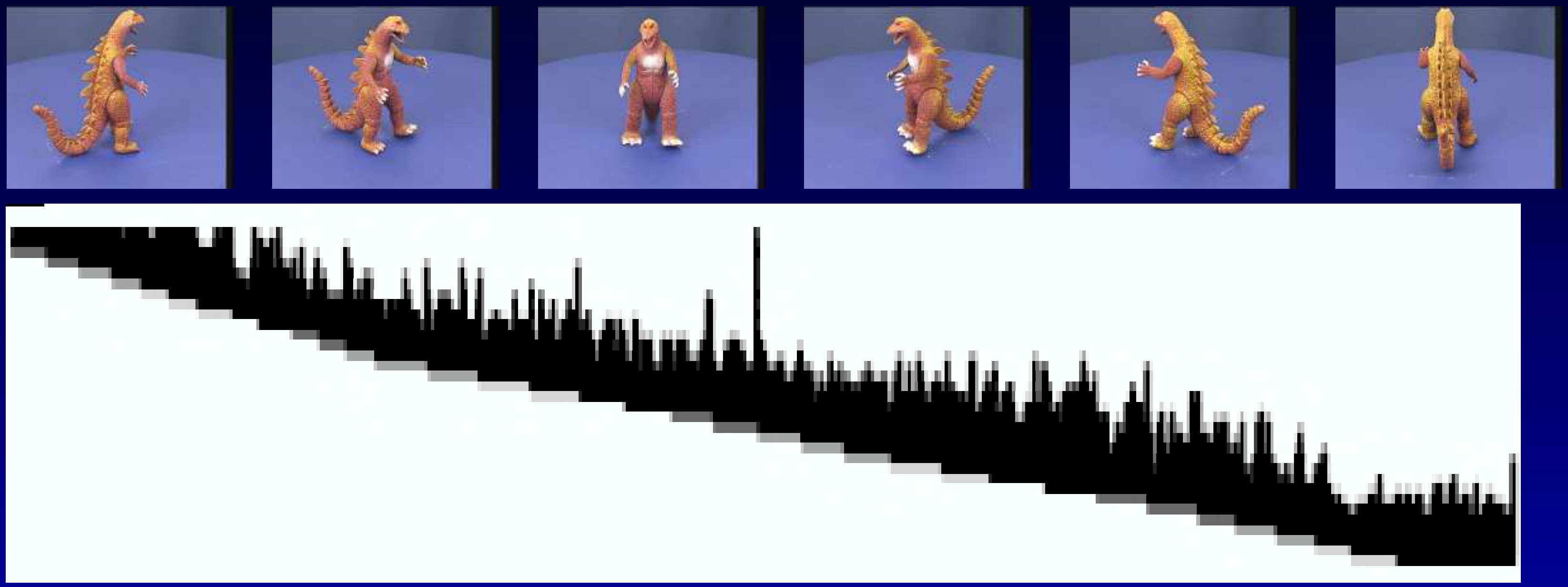


Figure 3: *Track Lifetimes* For a tracking sequence, most frames survive only a fraction of the length of the total sequence.

Banded Normal Equations

- Limited bandwidth tracking leads to banded structure for matrix $\mathbf{S}$.
- Update equations are of the form

$$\mathbf{S}\boldsymbol{\Delta}_{\mathbf{a}} = \epsilon$$

where $\mathbf{S}$ is symmetric, positive definite.

Banded Structure

Block $\mathbf{S}_{jk} \neq [\mathbf{0}] \Leftrightarrow$ some point is visible in both j-th and k-th images.

Proof:

- $\mathbf{S}_{jk} = -\sum_i \mathbf{W}_{ij} {\mathbf{V}_i^*}^{-1} \mathbf{W}_{ik}^\top$.
- $\mathbf{W}_{ij} = [\partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{a}_j]^\top [\partial \widehat{\mathbf{x}}_{ij} / \partial \mathbf{b}_i]$.

- $\mathbf{W}_{ij} \neq [\mathbf{0}] \Leftrightarrow$ feature i is visible to camera j.
- $\mathbf{S}_{jk} \neq [\mathbf{0}] \Leftrightarrow (\mathbf{W}_{ij} \neq [\mathbf{0}]$ and $\mathbf{W}_{ik} \neq [\mathbf{0}])$.

Banded Sparse Systems

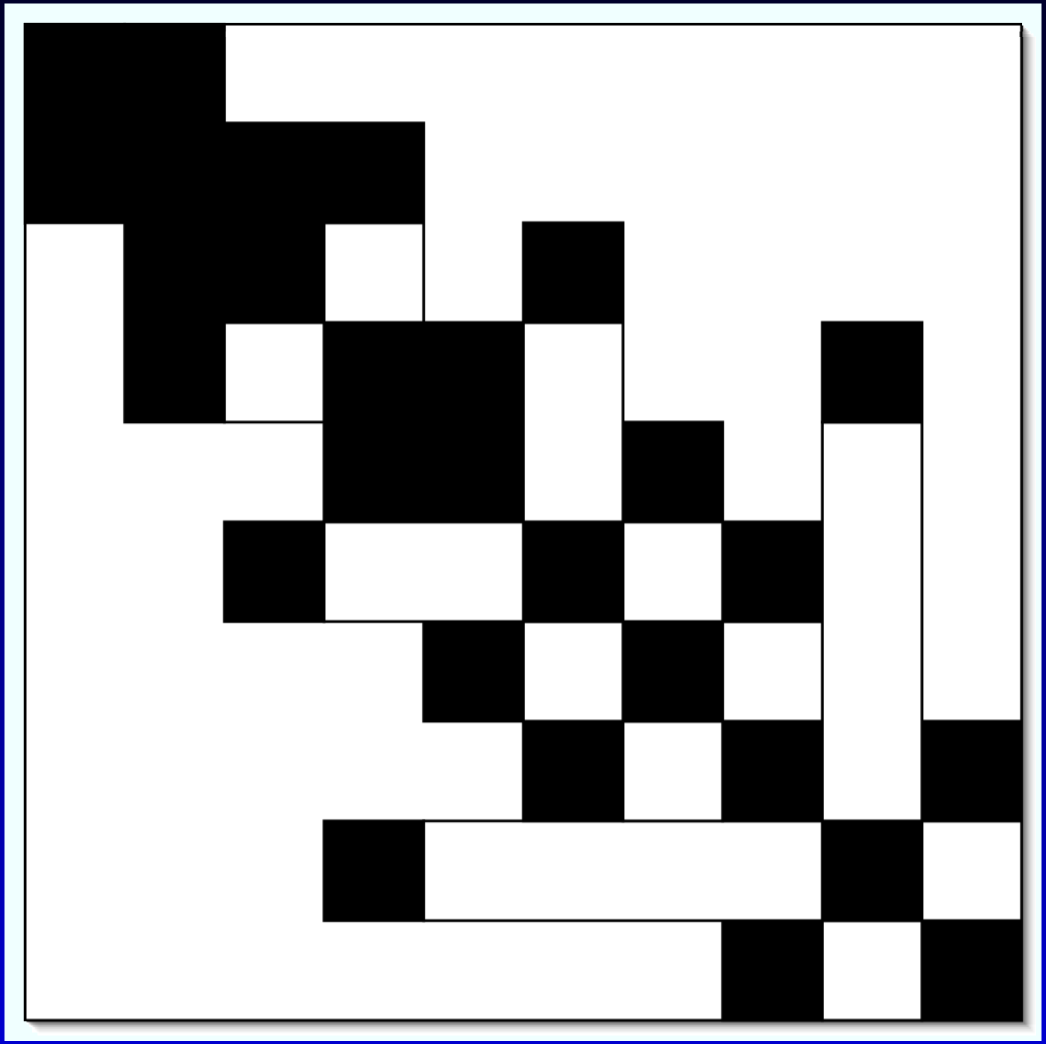


Figure 4: A banded sparse matrix. Black shows non-zero entries.

Skyline Structure

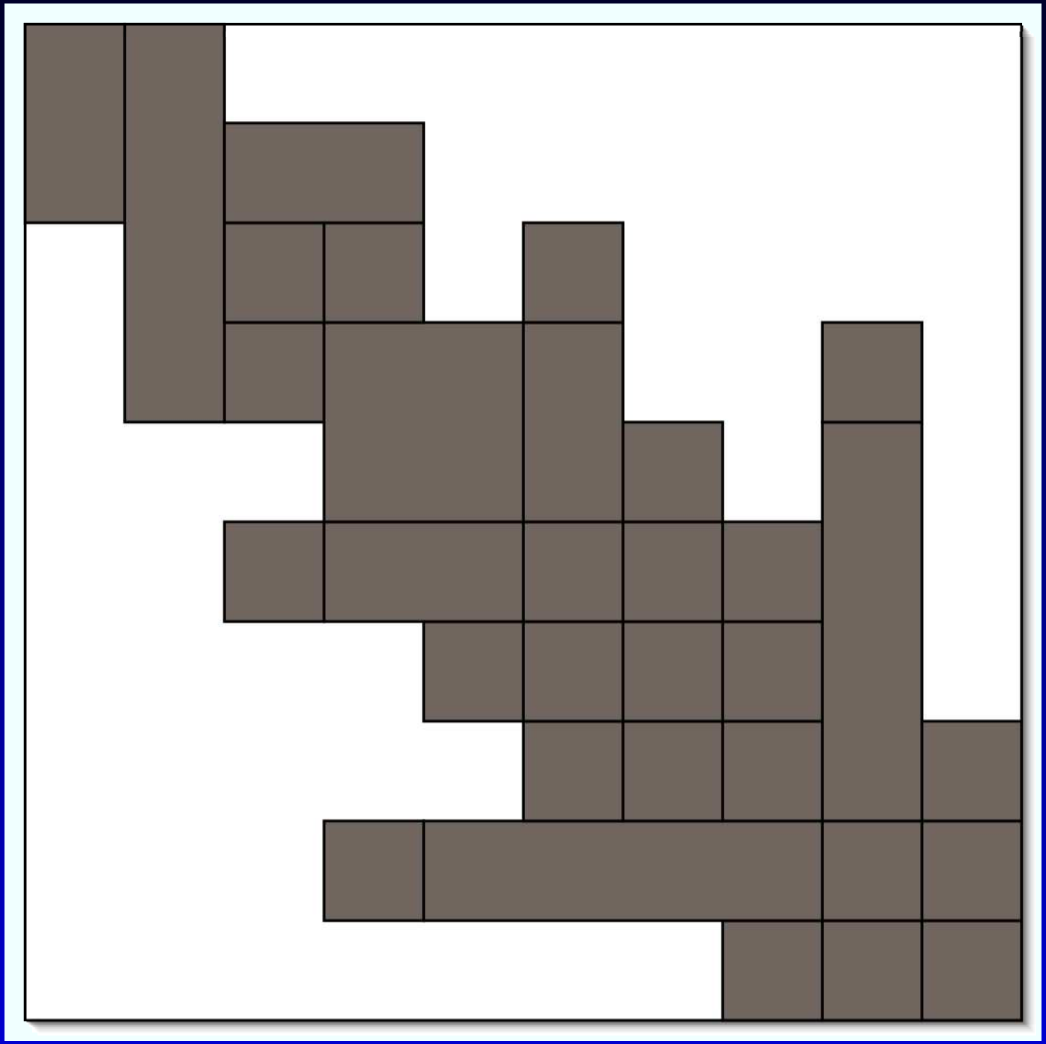


Figure 5: The skyline structure of a sparse matrix. All non-zero entries lie within the shaded region.

Skyline Decomposition

Let $\mathbf{A}$ *be a symmetric matrix such that* $\mathbf{A}_{ij} = 0$ *for* $j < m_i$. *Let* $\mathbf{A} = \mathbf{L}\mathbf{D}\mathbf{L}^\top$. *Then* $L_{ij} = 0$ *for* $j < m_i$.

Skyline structure of a matrix is preserved under LDL decomposition.

Solving Banded Systems

- *Forward substitution* : $\mathbf{x}'_i = \mathbf{b}_i - \sum_{j=m_i}^{i-1} L_{ij}\mathbf{x}'_j.$

Let $L_{ij} = 0$ for $i > m_j$. Then,

- *Back substitution* : $\mathbf{x}_i = \mathbf{x}''_i - \sum_{j=i+1}^{m_j} L_{ji}\mathbf{x}_j$

Summary

- Structure and sparsity help in optimization.
- Bundle adjustment is inherently sparse at various levels.
- Carefully structured Levenberg-Marquardt algorithm can exploit these sparsities at all levels.