CVPR 2014 Tutorial on Visual SLAM
Large Scale – Reducing Computational Cost

Michael Kaess
kaess@cmu.edu

The Robotics Institute
Carnegie Mellon University
Large-Scale Visual SLAM

Computational cost grows with time

Two approaches to reduce cost:

• Formulation
  – Keyframes
  – Submaps
  – Reduced pose graph

• Simplification
  – Cut old data
  – Sparsification
Large-Scale Visual SLAM

Computational cost grows with time

Two approaches to reduce cost:

• Formulation
  – Keyframes
  – Submaps
  – Reduced pose graph

• Simplification
  – Cut old data
  – Sparsification
Reduced pose graph

- Key-frame approach
- Reuses existing poses
- Grows with explored space, not time
- Partitions the environment
  - Maintains a set of poses that cover all the partitions
Reduced Pose Graph (step n) - Construction

In general, not revisiting exactly same poses

Standard pose graph:
Reduced Pose Graph (step n+1)

In general, not revisiting exactly same poses

Corresponds to a constraint between $x_i$ and $x_j$

Standard pose graph:

New pose is added
Reduced Pose Graph (step n+2)

Avoiding inconsistency

Second loop closure to $x_j$ to avoid double use of constraint

Standard pose graph:
Reduced Pose Graph (step n+3)

Avoiding inconsistency

Constraint between $x_i$ and $x_j$ added

Omitting short odometry links

Standard pose graph:
Long-term Visual Mapping

MIT Stata Center Dataset (publicly available)

- Duration: 6 months
- Operation time: 9 hours
- Distance travelled: 11 km (about 7 miles)
- VO keyframes: 630K
Reduced Pose Graph – Second Floor

iSAM optimizes reduced pose graph
Comparison of full vs reduced pose graph

- 4 Hours of data

- **Reduced pose graph**
  - # Poses 1363
  - Mean error 0.44m

- **Full pose graph**
  - # Poses 28520
  - Mean error 0.37m
Timing (approx. 9 hours of mission)

- SLAM Optimize
- Loop Proposal
- Registration
- Feature Extraction
- Feature Matching

No. SLAM pose nodes vs. Exploration Time [mins]

Reduced Pose Graph

[Data source: Johannsson, Kaess, Fallon, Leonard, ICRA 13]
Reduced Pose Graph – 10 Floors

iSAM optimizes reduced pose graph
Reduced Pose Graph

Map of 10 floors

- Accelerometer used to detect elevator transitions
- iSAM optimizes RPG to achieve real-time

[Johannsson, Kaess, Fallon, Leonard, ICRA 13]
Large-Scale Visual SLAM

Computational cost grows with time

Two approaches to reduce cost:

• **Formulation**
  – Keyframes
  – Submaps
  – Reduced pose graph

• **Simplification**
  – Cut old data
  – **Sparsification**
Sparsification: Factor Graph Node Removal

• Control complexity of performing inference in graph
  – Long-term multi-session SLAM
  – Reduces the size of graph
  – Storage and transmission

• Graph maintenance
  – Forgetting old views
Sparsification: Factor Graph Node Removal

Remove node from graph $\rightarrow$ marginalize variable from distribution

Original Graph

Dense Node Removal (Marginalization)  Sparse-Approximate Node Removal
Generic Linear Constraint Node Removal

(a) Original Graph

(b) Target Info.

(c) Chow-Liu Tree

(d) Root Shift

(e) Final Graph
Sparse Approximate GLC: Ensuring Conservative Approximations

- Chow-Liu Tree minimizes KLD

\[
D_{KL} \left( \mathcal{N}^{-1} (\eta_t, \Lambda_t) \| \mathcal{N}^{-1} (\tilde{\eta}_t, \tilde{\Lambda}_t) \right) = \frac{1}{2} \left( \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) + \frac{\ln |\Lambda_t|}{\ln |\tilde{\Lambda}_t|} - \text{dim}(\eta_t) \right)
\]
Sparse Approximate GLC: Ensuring Conservative Approximations

- Chow-Liu Tree minimizes KLD

\[
D_{KL} \left( \mathcal{N}^{-1}(\eta_t, \Lambda_t) \| \mathcal{N}^{-1}(\tilde{\eta}_t, \tilde{\Lambda}_t) \right) = \frac{1}{2} \left( \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) + \frac{\ln |\Lambda_t|}{\ln |\tilde{\Lambda}_t|} - \dim(\eta_t) \right)
\]

- Often results in a slightly overconfident estimate
Sparse Approximate GLC: Ensuring Conservative Approximations

• Why care about overconfident estimates?
  – Overconfidence in pose or obstacle location $\rightarrow$ unsafe paths
  – Overconfidence in pose or landmark location $\rightarrow$ failed data association
Sparse Approximate GLC: Ensuring Conservative Approximations

• Propose method to ensure conservative approximation
• Start with CLT which minimizes the KLD and then numerically adjust it to produce a conservative estimate
Sparse Approximate GLC: Ensuring Conservative Approximations

- Constrained convex optimization problem
- Minimize the KLD

\[
D_{KL} \left( \mathcal{N}^{-1}(\eta_t, \Lambda_t) \| \mathcal{N}^{-1}(\tilde{\eta}_t, \tilde{\Lambda}_t) \right) = \frac{1}{2} \left( \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) + \frac{\ln |\Lambda_t|}{\ln |\tilde{\Lambda}_t|} - \dim(\eta_t) \right)
\]

\[
f_{KL}(\tilde{\Lambda}_t) = \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) - \ln |\tilde{\Lambda}_t|
\]

- Subject to conservative constraint (difference is PSD)

\[
\tilde{\Sigma} \geq \Sigma \quad \Leftrightarrow \quad \Lambda \geq \tilde{\Lambda}
\]
Sparse Approximate GLC: Ensuring Conservative Approximations

- Constrained convex optimization problem
- Minimize the KLD

\[ D_{KL} \left( \mathcal{N}^{-1}(\eta_t, \Lambda_t) \| \mathcal{N}^{-1}(\tilde{\eta}_t, \tilde{\Lambda}_t) \right) = \frac{1}{2} \left( \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) + \frac{\ln |\Lambda_t|}{\ln |\tilde{\Lambda}_t|} - \dim(\eta_t) \right) \]

\[ f_{KL}(\tilde{\Lambda}_t) = \text{tr}(\tilde{\Lambda}_t \Lambda_t^{-1}) - \ln |\tilde{\Lambda}_t| \]

- Subject to conservative constraint (difference is PSD)

\[ \tilde{\Sigma} \geq \Sigma \iff \Lambda \geq \tilde{\Lambda} \]
Sparse Approximate GLC: Ensuring Conservative Approximations

- Start with the Chow-Liu Tree

- Consider three methods
  - Covariance Intersection
  - Weighted Factors
  - Weighted Eigenvalues

- Convex semidefinite programs
Chow-Liu Tree Approximation

- All proposed methods start with the CLT

\[ \mathcal{N}^{-1}(\eta_t, \Lambda_t) \approx \mathcal{N}^{-1}(\tilde{\eta}_t, \tilde{\Lambda}_{\text{CLT}}) = \prod_i p(x_i | x_{p(i)}) \]

\[ \tilde{\Lambda}_{\text{CLT}} = \sum_i \Psi_i \]
Covariance Intersection

[Julier and Uhlmann, 1997]

- Convex combination of correlated factors

\[
\Lambda_{CI} = \sum_i w_i \Psi_i
\]

\[
\text{minimize}_{\mathbf{w}} \quad f_{KL}(\Lambda_{CI}(\mathbf{w}))
\]

subject to \(\sum_i w_i = 1\)
Weighted Factors

- Replace constraint that weights sum to one

\[ \Lambda \approx \tilde{\Lambda}_{WF} = \Psi_1 + \Psi_2 \]

\[ \tilde{\Lambda}_{WF}(\mathbf{w}) = \sum_i w_i \Psi_i \]

minimize \( f_{KL}(\tilde{\Lambda}_{WF}(\mathbf{w})) \)

subject to

- \( 0 \leq w_i \leq 1, \forall i \)
- \( \Lambda_t \geq \tilde{\Lambda}_{WF}(\mathbf{w}) \)
Weighted Eigenvalues

- Modify each eigenvalue of each factor

\[
\tilde{\Lambda}_{\text{WEV}}(\mathbf{w}) = \sum_{i} \sum_{j=1}^{q_i} w_j^i \lambda_j^i \mathbf{u}_j^i \mathbf{u}_j^i \mathbf{T} = \sum_k w_k \lambda_k \mathbf{u}_k \mathbf{u}_k^\mathbf{T}
\]

minimize \[
f_{KL}(\tilde{\Lambda}_{\text{WEV}}(\mathbf{w}))
\]
subject to
\[
0 \leq w_k \leq 1, \ \forall k
\]
\[
\Lambda_t \geq \tilde{\Lambda}_{\text{WEV}}(\mathbf{w})
\]
## Conservative GLC: Experimental Results

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Node Types</th>
<th>Factor Types</th>
<th># Nodes</th>
<th># Factors</th>
</tr>
</thead>
<tbody>
<tr>
<td>Intel Lab</td>
<td>3-DOF pose</td>
<td>3-DOF odom., 3-DOF laser scan-matching</td>
<td>910</td>
<td>4,454</td>
</tr>
<tr>
<td>Killian Court</td>
<td>3-DOF pose</td>
<td>3-DOF odom., 3-DOF laser scan-matching</td>
<td>1,941</td>
<td>2,191</td>
</tr>
<tr>
<td>Victoria Park</td>
<td>3-DOF pose, 2-DOF Im.</td>
<td>3-DOF odom., 2-DOF landmark observation</td>
<td>7,120</td>
<td>10,609</td>
</tr>
<tr>
<td>Duderstadt Center</td>
<td>6-DOF pose</td>
<td>6-DOF odom., 6-DOF laser scan-matching</td>
<td>552</td>
<td>1,774</td>
</tr>
<tr>
<td>EECS Building</td>
<td>6-DOF pose</td>
<td>6-DOF odom., 6-DOF laser scan-matching</td>
<td>611</td>
<td>2,134</td>
</tr>
<tr>
<td>USS Saratoga</td>
<td>6-DOF pose</td>
<td>6-DOF odom., 5-DOF mono-vis., 1-DOF depth</td>
<td>1,513</td>
<td>5,433</td>
</tr>
</tbody>
</table>

![Diagrams](image_url)
Conservative GLC: Experimental Results

- Remove percentage of evenly spaced nodes from each graph
- CI very conservative
- WF and WEV approach performance of CLT
  - for most graphs
- Room improvement for Intel
  - Higher density of connectivity
  - All factor same strength