Pose Fusion with Chain Pose Graphs for Automated Driving

Reading Report

HMW-Alexander, Prof. Raj Rajkumar

CMU-ECE

November 14, 2016

Basic Information

Authors

- **Christian Merfels** is with Volkswagen Group Research, Wolfsburg, and Institute of Geodesy and Geoinformation, University of Bonn, Germany.
- **Cyrill Stachniss** is with Institute of Geodesy and Geoinformation, University of Bonn, Germany.

Conference

2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)

Abstract

- A **graph-based multiple localization source fusion** approach.
  - A kind of **Non-Linear Least Squares** optimization method.
  - Different rates & Different coverages.
  - Third-party localization modules.
- **Sliding window** and **Chain Pose Graph** guarantee the real-time efficiency.
  - Sliding window size controls the number of history nodes be evaluated.
    - If size=1, then it becomes a filtering-based solution.
    - If size=all, then it becomes a batch solution.
  - Chain Pose Graph avoids the "fill-in" problem while marginalizing nodes.[1]
    - Schur Complement generally fill-in the marginalized system matrix.
    - Find a marginalization solution with minimal fill-in is NP-complete.
    - It's helpful for the Cholesky decomposition to solve the final linear solution.
- Experiments
  - A real experiment fused the LiDAR, camera, GPS and odometry.
  - A simulated experiment demonstrated the accuracy's relation to the sliding window size and the number of localization sources.

[1] At the cost of loosing constraints between non-consecutive nodes.

Abstract

- A **graph-based multiple localization source fusion** approach.
  - A kind of **Non-Linear Least Squares** optimization method.
  - Different rates & Different coverages.
  - Third-party localization modules.
- **Sliding window** and **Chain Pose Graph** guarantee the real-time efficiency.
  - Sliding window size controls the number of history nodes be evaluated.
    - If size=1, then it becomes a filtering-based solution.
    - If size=all, then it becomes a batch solution.
  - Chain Pose Graph avoids the "fill-in" problem while marginalizing nodes.[1]
    - Schur Complement generally fill-in the marginalized system matrix.
    - Find a marginalization solution with minimal fill-in is NP-complete.
    - It's helpful for the Cholesky decomposition to solve the final linear solution.
- Experiments
  - A real experiment fused the LiDAR, camera, GPS and odometry.
  - A simulated experiment demonstrated the accuracy's relation to the sliding window size and the number of localization sources.

[1] At the cost of loosing constraints between non-consecutive nodes.

Abstract

- A **graph-based multiple localization source fusion** approach.
  - A kind of **Non-Linear Least Squares** optimization method.
  - Different rates & Different coverages.
  - Third-party localization modules.
- **Sliding window** and **Chain Pose Graph** guarantee the real-time efficiency.
  - Sliding window size controls the number of history nodes be evaluated.
    - If size=1, then it becomes a filtering-based solution.
    - If size=all, then it becomes a batch solution.
  - Chain Pose Graph avoids the "fill-in" problem while marginalizing nodes.[1]
    - Schur Complement generally fill-in the marginalized system matrix.
    - Find a marginalization solution with minimal fill-in is NP-complete.
    - It's helpful for the Cholesky decomposition to solve the final linear solution.
- Experiments
  - A real experiment fused the LiDAR, camera, GPS and odometry.
  - A simulated experiment demonstrated the accuracy's relation to the sliding window size and the number of localization sources.

[1] At the cost of loosing constraints between non-consecutive nodes.

Introduction

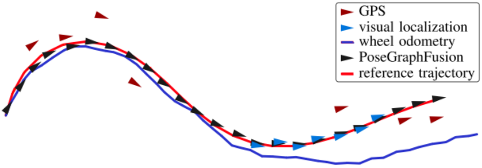
Problem & Solution

Individual localization system is not enough (accuracy and coverage), and the combination of orthogonal localization systems is more powerful.

Objective

Provides a multi-sensor data fusion approach which is decoupled from the localization algorithm.

Formulation



Related Work

Filtering-based appraoches

Kalman filter and its variants, particle filter

- Feature: Markov assumption, marginalize all older information.
- Problem: Prematurely incorporating the linearization error.

Sliding window smoothing algorithms

Compute the ML estimate by Non-Linear Least Squares optimization.

- Feature: consider past measurements
- Problem: large window size requires high computation cost.

Method: Pose Graph Fusion

Nonlinear Least Squares Problem

$$x^* = \arg\min_x \sum_{i,j} e_{ij}^T \Omega_{ij} e_{ij}$$

- $x = [x_1^T, \ldots, x_m^T]^T$ is the state vector.
- A measurement between x_i and x_j:
  - z_{ij} is the mean of this measurement
  - $\Omega_{ij} = \Sigma_{ij}^{-1}$ is the information matrix of this measurement
- An error evaluation function $e_{ij}(x_i, x_j, z_{ij})$

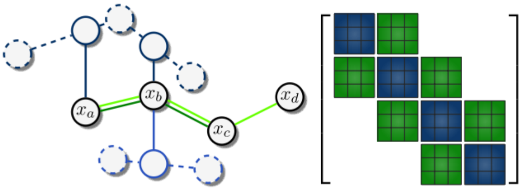
System Matrix and Coefficient Vector

$$H = \sum_{i,j} J_{ij}(x)^T \Omega_{ij} J_{ij}(x), \;\; b^T = \sum_{i,j} e_{ij}^T \Omega_{ij} J_{ij}(x)$$

Solution:

$$H\Delta x = -b, \;\; x^* = \hat{x} + \Delta x$$

Method: Pose Graph Fusion



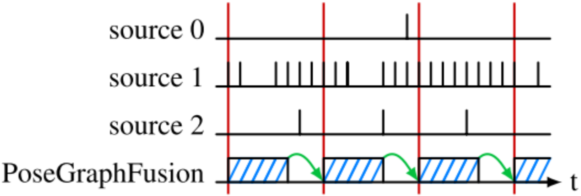
(a) A chain pose graph.

(b) Corresponding block-tridiagonal structure of the system matrix.

Fig. 3. A chain pose graph and the corresponding structure of the system matrix. The black circles are hidden nodes, the dashed blue circles are global pose measurements from two different sources, the non-dashed blue circles are observed nodes, and the green edges are odometry constraints. Note how the raw global pose measurements are interpolated (dashed blue lines) at the same timestamps as the hidden nodes to obtain the observed nodes.

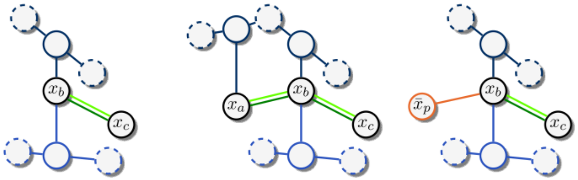
- Measurements are from different localization sources at certain time (interpolation is required)
- The odometries information represent the edge between two consecutive state nodes.

Method: Time Behavior



- The rate of fusion result is constant and is determined by the PoseGraphFusion.
- Different window size affects the computation time.

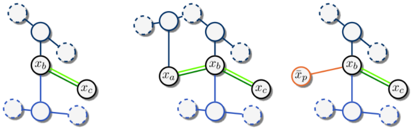
Marginalize the Older Nodes



(a) Graph $\mathcal{G}^{\text{small}}$. (b) Graph $\mathcal{G}^{\text{full}}$. (c) Graph $\mathcal{G}^{\text{prior}}$.

- (a) directly remove the older node x_a, introduce dependency error.
- (b) the original full graph.
- (c) replace x_a with a virtual prior measurement $\bar{x}_p$.

Math about the Marginalization (from full to marginalized)

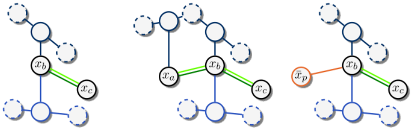


(a) Graph $\mathcal{G}^{\text{small}}$. (b) Graph $\mathcal{G}^{\text{full}}$. (c) Graph $\mathcal{G}^{\text{prior}}$.

$$H^{small}\Delta x^{small} = \begin{bmatrix} H^{small}_{bb} & H^{small}_{bc} \\ H^{small}_{cb} & H^{small}_{cc} \end{bmatrix} \Delta x^{small} = -\begin{bmatrix} b^{small}_b \\ b^{small}_c \end{bmatrix} \tag{1}$$

$$\begin{aligned} H^{full}\Delta x^{full} &= \begin{bmatrix} \bar{H}_{aa}+\hat{H}_{aa} & \hat{H}_{ab} & 0 \\ \hat{H}_{ba} & H^{small}_{bb}+\hat{H}_{bb} & H^{small}_{bc} \\ 0 & H^{small}_{cb} & H^{small}_{cc} \end{bmatrix} \Delta x^{full} \\ &= \left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & H^{small}_{bb} & H^{small}_{bc} \\ 0 & H^{small}_{cb} & H^{small}_{cc} \end{bmatrix} + \begin{bmatrix} \bar{H}_{aa}+\hat{H}_{aa} & \hat{H}_{ab} & 0 \\ \hat{H}_{ba} & \hat{H}_{bb} & 0 \\ 0 & 0 & 0 \end{bmatrix} \right) \Delta x^{full} \\ &= -\begin{bmatrix} 0 \\ b^{small}_b \\ b^{small}_c \end{bmatrix} - \begin{bmatrix} \bar{b}_a+\hat{b}_a \\ \hat{b}_b \\ 0 \end{bmatrix} = -\begin{bmatrix} \bar{b}_a+\hat{b}_a \\ b^{small}_b+\hat{b}_b \\ b^{small}_c \end{bmatrix} \end{aligned} \tag{2}$$

Math about the Marginalization (Schur Complement)



(a) Graph $\mathcal{G}^{\mathrm{small}}$. (b) Graph $\mathcal{G}^{\mathrm{full}}$. (c) Graph $\mathcal{G}^{\mathrm{prior}}$.

$$\begin{aligned}
\begin{bmatrix} \bar{H}_{aa} + \hat{H}_{aa} & \hat{H}_{ab} \\ \hat{H}_{ba} & \hat{H}_{bb} \end{bmatrix} &\Rightarrow & H^{schur} = \hat{H}_{bb} - \hat{H}_{ba}(\bar{H}_{aa} + \hat{H}_{aa})^{-1}\hat{H}_{ab} \\
b^{schur} = H^{schur}\Delta x_b &= & \hat{b}_b - \hat{H}_{ba}(\bar{H}_{aa} + \hat{H}_{aa})^{-1}(\bar{b}_a + \hat{b}_a)
\end{aligned} \tag{3}$$

$$\begin{aligned}
H^{marg} &= \begin{bmatrix} H^{small}_{bb} + H^{schur} & H^{small}_{bc} \\ H^{small}_{cb} & H^{small}_{cc} \end{bmatrix} \\
b^{marg} &= \begin{bmatrix} b^{small}_{b} + b^{schur} \\ b^{small}_{c} \end{bmatrix}
\end{aligned} \tag{4}$$

After the marginalization, the structure of the system matrix is still block-tridiagonal due to the particular design of the chain pose graph, meaning that the sparsity pattern of H is retained after marginalization without fill-in.

Math about the Virtual Prior Measurement

$$
\begin{aligned}
H^{prior} &= \begin{bmatrix} H_{bb}^{small} + \bar{H}^{prior} & H_{bc}^{small} \\ H_{cb}^{small} & H_{cc}^{small} \end{bmatrix} \\
b^{prior} &= \begin{bmatrix} b_b^{small} + \bar{b}^{prior} \\ b_c^{small} \end{bmatrix}
\end{aligned} \tag{5}
$$

$$
\begin{aligned}
e_{pb} &= \begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix} (x_b - \bar{x}_p) \\
J_{pb}(x) &= \begin{bmatrix} R & 0 \\ 0 & 1 \end{bmatrix}
\end{aligned} \tag{6}
$$

$$
\begin{aligned}
\bar{H}_p &= J_{pb}(x)^T \bar{\Omega}_p J_{pb}(x) = H^{schur} \\
\Rightarrow \bar{\Omega}_p &= (J_{pb}(x)^T)^{-1} H^{schur} (J_{pb}(x))^{-1} \\
\bar{b}_p &= J_{pb}(x)^T \bar{\Omega}_p e_{pb} = J_{pb}(x)^T \bar{\Omega}_p J_{pb}(x)(x_b - \bar{x}_p) = \bar{H}_p (x_b - \bar{x}_p) \\
\Rightarrow \bar{x}_p &= x_b - (H^{schur})^{-}1 b^{schur}
\end{aligned} \tag{7}
$$

Method Summary

- Localization sources can be fused in the form of a graph, as well as an adjacent matrix (the system matrix H)
  - Fusion is decoupled with localization.
  - Localization is done by non-linear least squares on H and b after fusion.
- The special design on the Chain Pose Graph makes it easier to be solved.
  - Different localization sources are measured at certain timestamps via interpolation
    - decouple the measurement relationship between state nodes
    - handle different rate and coverage problem.
  - Odometries only connect two consecutive nodes.
- Marginalization enables constant sliding window size, as well as constant final localization rate.
  - The Chain Pose Graph's property avoids the fill-in problem during the marginalization.
  - Compact the marginalized nodes into a virtual prior measurement.

Method Summary

- Localization sources can be fused in the form of a graph, as well as an adjacent matrix (the system matrix H)
  - Fusion is decoupled with localization.
  - Localization is done by non-linear least squares on H and b after fusion.
- The special design on the Chain Pose Graph makes it easier to be solved.
  - Different localization sources are measured at certain timestamps via interpolation
    - decouple the measurement relationship between state nodes
    - handle different rate and coverage problem.
  - Odometries only connect two consecutive nodes.
- Marginalization enables constant sliding window size, as well as constant final localization rate.
  - The Chain Pose Graph's property avoids the fill-in problem during the marginalization.
  - Compact the marginalized nodes into a virtual prior measurement.

Method Summary

- Localization sources can be fused in the form of a graph, as well as an adjacent matrix (the system matrix H)
  - Fusion is decoupled with localization.
  - Localization is done by non-linear least squares on H and b after fusion.
- The special design on the Chain Pose Graph makes it easier to be solved.
  - Different localization sources are measured at certain timestamps via interpolation
    - decouple the measurement relationship between state nodes
    - handle different rate and coverage problem.
  - Odometries only connect two consecutive nodes.
- Marginalization enables constant sliding window size, as well as constant final localization rate.
  - The Chain Pose Graph's property avoids the fill-in problem during the marginalization.
  - Compact the marginalized nodes into a virtual prior measurement.

Experiment: Real Data

Settings

- global sources:
  - source 0: coarse LiDAR scans to a globally referenced point cloud. (3rd-party)
  - source 1: GPS (3rd-party)
  - source 2: visual localization system with a globally referenced feature map.
- local source: wheel odometry.
- sliding window size: $M = 1000$
- temporal resolution: $\Delta t = 25ms$
- experiment route: $16km$ in rural and urban areas in Germany.

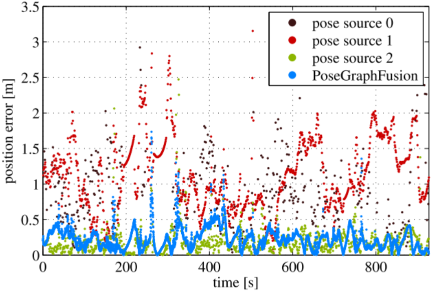
Estimation (Baseline: batch solution)

Accuracy:

- source 0 (LiDAR): $RMS = 1.06m$
- source 1 (GPS): $RMS = 1.23m$
- source 2 (Camera): $RMS = 0.28m$
- PoseGraphFusion: $RMS = 0.38m$

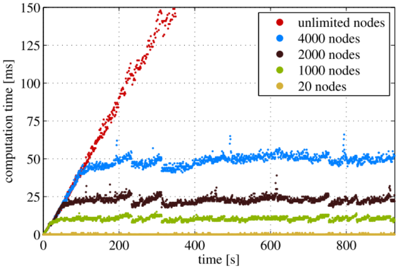
Coverage rate:

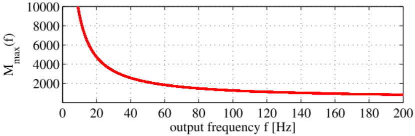
- source 0: 66.98%
- source 1 with odometry: 100%
- source 2: 97.76%
- PoseGraphFusion: 100%



Runtime Performance

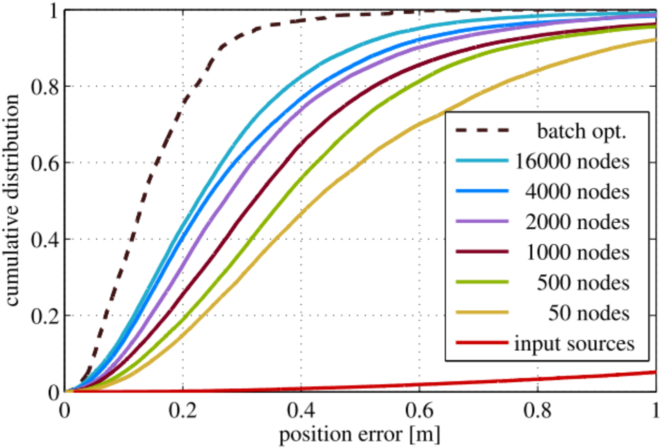
a single core of a laptop with an Intel i7-4800QM processor





Sim Data: Number of Hidden Nodes v.s. Accuracy

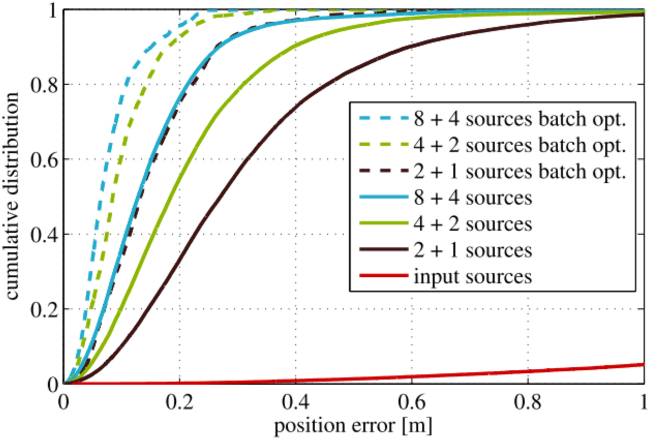
Larger sliding window size $\Rightarrow$ more accurate result



CyLab

Sim Data: Number of Pose Sources v.s. Accuracy

More pose sources ⇒ more accurate result



Conclusion Related with CORAL

- A graph-based method to fuse different localization sources:
  - Multiple GNSS w/ or w/o RTK (global)
  - Visual place recognition based on landmarks and digital map (global)
  - Visual odometry (local)
  - Encoder/IMU (local)
- The paper doesn't prove if all pose sources' coverage rates are low (one is 100%). We can test it.
- Localization frequency is controllable:
  - Relative localization: requires high-frequency and local high-accuracy. Global accuracy is not required. (small sliding window size & more relative localization modules)
  - Absolute localization: requires low-frequency and global high-accuracy. Local accuracy is not required. (large sliding window size)
- The possible to COordinate Relative and Absolute Localization:
  - The Relative localization graph is a subgraph of the Absolute localization graph.
  - The bi-directional relationship can be built via the graph.

CyLab

Conclusion Related with CORAL

- A graph-based method to fuse different localization sources:
  - Multiple GNSS w/ or w/o RTK (global)
  - Visual place recognition based on landmarks and digital map (global)
  - Visual odometry (local)
  - Encoder/IMU (local)
- The paper doesn't prove if all pose sources' coverage rates are low (one is 100%). We can test it.
- Localization frequency is controllable:
  - Relative localization: requires high-frequency and local high-accuracy. Global accuracy is not required. (small sliding window size & more relative localization modules)
  - Absolute localization: requires low-frequency and global high-accuracy. Local accuracy is not required. (large sliding window size)
- The possible to COordinate Relative and Absolute Localization:
  - The Relative localization graph is a subgraph of the Absolute localization graph.
  - The bi-directional relationship can be built via the graph.

CyLab

Conclusion Related with CORAL

- A graph-based method to fuse different localization sources:
  - Multiple GNSS w/ or w/o RTK (global)
  - Visual place recognition based on landmarks and digital map (global)
  - Visual odometry (local)
  - Encoder/IMU (local)
- The paper doesn't prove if all pose sources' coverage rates are low (one is 100%). We can test it.
- Localization frequency is controllable:
  - Relative localization: requires high-frequency and local high-accuracy. Global accuracy is not required. (small sliding window size & more relative localization modules)
  - Absolute localization: requires low-frequency and global high-accuracy. Local accuracy is not required. (large sliding window size)
- The possible to COordinate Relative and Absolute Localization:
  - The Relative localization graph is a subgraph of the Absolute localization graph.
  - The bi-directional relationship can be built via the graph.

CyLab

Conclusion Related with CORAL

- A graph-based method to fuse different localization sources:
  - Multiple GNSS w/ or w/o RTK (global)
  - Visual place recognition based on landmarks and digital map (global)
  - Visual odometry (local)
  - Encoder/IMU (local)
- The paper doesn't prove if all pose sources' coverage rates are low (one is 100%). We can test it.
- Localization frequency is controllable:
  - Relative localization: requires high-frequency and local high-accuracy. Global accuracy is not required. (small sliding window size & more relative localization modules)
  - Absolute localization: requires low-frequency and global high-accuracy. Local accuracy is not required. (large sliding window size)
- The possible to COordinate Relative and Absolute Localization:
  - The Relative localization graph is a subgraph of the Absolute localization graph.
  - The bi-directional relationship can be built via the graph.

CyLab