Incremental Inference and Applications

Michael Kaess
kaess@mit.edu

Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Overview

• Incremental Inference
  – Background
  – iSAM1: Matrices
  – iSAM2: Factor Graphs

• Applications
  – Ship Hull Inspection
  – Concurrent Filtering and Smoothing
  – Dense Visual SLAM
The Mapping Problem (t=0)
The Mapping Problem (t=1)

Odometry measurement

Robot

Landmark 1

Landmark measurement

Landmark 1

Landmark 2
The Mapping Problem (t=n-1)

Odometry measurement

Landmark measurement

Landmark 1

Landmark 2

Robot
The Mapping Problem (t=n)

Odometry measurement

Landmark measurement
Factor Graph Representation

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

- **Odometry measurement**
- **Landmark measurement**

Robot pose

Landmark position
Sequence of Factor Graphs!

$t=0$

$t=1$

$t=n-1$
Overview

• Incremental Inference
  – Background
  – iSAM1: Matrices
  – iSAM2: Factor Graphs

• Applications
  – Ship Hull Inspection
  – Concurrent Filtering and Smoothing
  – Dense Visual SLAM
Nonlinear Least-Squares

Repeatedly solve linearized system

\[
\text{argmin}_\Theta \sum_i \|h_i(\Theta)\|_\Xi^2
\]

Many exact or approximate solutions

[Lu&Milios 97, Konolige 04, Folkesson&Christensen 04, Eustice et al. 05, Frese 06, Olson et al. 06, Dellaert&Kaess 06, Grisetti et al. 10]

\[
\text{argmax}_\Theta \prod_i p_i(\Theta)
\]

Gaussian noise

poses landmarks
Solving the Linear Least-Squares System

Solve:  \( \text{argmin}_{\theta} \| A\theta - b \|^2 \)

Normal equations
\[ A^T A \theta = A^T b \]

Matrix factorization
\[ A^T A = R^T R \]

Measurement Jacobian

Information matrix

Square root information matrix
Retaining Sparsity: Variable Ordering

Fill-in depends on elimination order:

\[ A^T A \]

Default ordering (poses, landmarks)

Ordering based on COLAMD heuristic [Davis04]
(best order: NP hard)
Matrix – Square Root Factorization

- QR on $A$: Numerically Stable

- Cholesky on $A^TA$: Faster
iSAM [Kaess, Ranganathan, Dellaert, TRO 08]

Solving a growing system:
- Exact/batch (quickly gets expensive)
- Approximations
- Incremental Smoothing and Mapping (iSAM)

Key idea:
- Append to existing matrix factorization
- “Repair” using Givens rotations

Periodic batch steps for
- Relinearization
- Variable reordering (to keep sparsity)
Example from real sequence:
Square root inf. matrix
Side length: 21000 variables
Dense: 1.7GB, sparse: 1MB

How to avoid periodic batch steps?
Overview

• Incremental Inference
  – Background
  – iSAM1: Matrices
  – iSAM2: Factor Graphs

• Applications
  – Ship Hull Inspection
  – Concurrent Filtering and Smoothing
  – Dense Visual SLAM
Matrix vs. Graph

Measurement Jacobian

Factor Graph

Information Matrix

Markov Random Field

Square Root Inf. Matrix
Matrix vs. Graph

Measurement Jacobian

Factor Graph

Information Matrix

Markov Random Field

Square Root Inf. Matrix

Bayes Tree
iSAM2: Bayes Tree [Kaess et al., WAFR 10/IJRR 12]

Insight: Can perform inference in tree structure

Two stage process:
- Variable elimination converts factor graph to Bayes net
- Discovering cliques provides Bayes tree

- “The Bayes Tree: An Algorithmic Foundation for Probabilistic Robot Mapping”
  M. Kaess, V. Ila, R. Roberts, and F. Dellaert.
  WAFR 2010

- “iSAM2: Incremental Smoothing and Mapping Using the Bayes Tree”
  IJRR 2012
iSAM2: Variable Elimination – Small Example

• Choose ordering: \( l_1, l_2, x_1, x_2, x_3 \)

• Eliminate one node at a time

\[
p(l_1, x_1, x_2) = p(l_1 | x_1, x_2) \cdot p(x_1, x_2)
\]
iSAM2: Variable Elimination – Small Example

• Choose ordering: $l_1, l_2, x_1, x_2, x_3$
• Eliminate one node at a time

\[
p(l_1, x_1, x_2) = p(l_1 | x_1, x_2) \cdot p(x_1, x_2)
\]
iSAM2: Variable Elimination – Small Example

• Choose ordering: \( l_1, l_2, x_1, x_2, x_3 \)
• Eliminate one node at a time

\[
p(l_2, x_3) = p(l_2 | x_3) \ p(x_3)
\]
iSAM2: Variable Elimination – Small Example

• Choose ordering: $l_1, l_2, x_1, x_2, x_3$

• Eliminate one node at a time

\[ p(x_1, x_2) = p(x_1 | x_2) p(x_2) \]
iSAM2: Variable Elimination – Small Example

- Choose ordering: $l_1, l_2, x_1, x_2, x_3$
- Eliminate one node at a time

$$p(x_2, x_3) = p(x_2 | x_3) \ p(x_3)$$
iSAM2: Variable Elimination – Small Example

• Choose ordering: $l_1, l_2, x_1, x_2, x_3$

• Eliminate one node at a time

$p(x_3)$
iSAM2: Bayes Tree Data Structure [Kaess et al., IJRR 12]

Step 1

Step 2: Find cliques in reverse elimination order:

\[
R = \begin{bmatrix}
    l_1 & l_2 & x_1 & x_2 & x_3 \\
    & l_1 & x_1 & x_2 & x_3 \\
    & & l_2 & x_2 & x_3 \\
    & & & x_2 & x_3 \\
    & & & & x_3
\end{bmatrix}
\]
iSAM2: Bayes Tree Data Structure [Kaess et al., IJRR 12]

Step 1

Step 2: Find cliques in reverse elimination order:

\[ P(x_j | S_j) \propto \exp \left\{ -\frac{1}{2\sigma^2} (x_j + rS_j - d)^2 \right\} \]
Bayes Tree vs. Junction Tree/Clique Tree

BT = direct(ed) result from elimination
More intuitive, directly encodes square root inform. factor, but also less general: reflects an ordering

Chordal Bayes Net and cliques

Bayes Tree

Junction Tree
iSAM2: Bayes Tree Example  [Kaess et al., IJRR12]

How to update with new measurements / add variables?

Manhattan dataset (Olson)
iSAM2: Updating the Bayes Tree [Kaess et al., IJRR12]

Add new factor between $x_1$ and $x_3$
iSAM2: Updating the Bayes Tree [Kaess et al., IJRR12]

Add new factor between $x_1$ and $x_3$
Incremental Variable Reordering

For a small loop, what constitutes a “good” ordering?

Include loop closing into cut

Loop closing not part of cut

Include loop closing into cut

Loop closing not part of cut

Trajectory

Affected by next update

Bayes tree
Incremental Variable Reordering

Most recent variable at the end

expected to make future updates cheaper

• Force most recent variables to the end
• Find best ordering for remaining variables

Using constrained version of COLAMD algorithm (CCOLAMD)
Variable Reordering – Constrained COLAMD

Greedy approach
Arbitrary placement of newest variable

Constrained Ordering
Newest variables forced to the end

Number of affected variables:
low                                      high

Much cheaper!
iSAM2: Incremental Update + Var. Ordering

Variable ordering changes incrementally during update
- Not understood in matrix version
- Sparse matrix data structure not suitable

Large savings in computation
Variable Reordering – Fill-in

Incremental ordering still yields good overall ordering

- Only slightly more fill-in than batch COLAMD ordering
- Constrained ordering is worse than naïve/greedy:
  - Suboptimal ordering because of partial constraint, but cheaper to update!
Recovering Only Variables That Change

Again good quality and low cost are achievable:
iSAM2: Fluid Relinearization  [Kaess et al., IJRR12]

Relinearize select variables only

- Changes in map estimates are often local
- Most variables do not need to be updated
- Can be combined with updates

![Graph showing cumulative time vs. time step for different datasets]

City 10000 dataset

iSAM1: Kaess et al., TRO 08
iSAM2: Kaess et al., IJRR 12
SPA: Konolige et al., IROS 2010
HOG-Man: Grisetti et al., ICRA 2010
iSAM2: Bayes Tree for Manhattan Sequence

\[ x_0 \]

Kaess et al., IJRR 12
Beyond Gaussian Noise

• E.g.: robust estimators
• Usually Levenberg-Marquardt, but cannot be done incrementally
• Solution: Powell’s Dog Leg

• “An Incremental Trust-Region Method for Robust Online Sparse Least-Squares Estimation”
  D.M. Rosen, M. Kaess, and J.J. Leonard.
  ICRA 2012

• “Robust Incremental Online Inference Over Sparse Factor Graphs: Beyond the Gaussian Case”
  D.M. Rosen, M. Kaess, and J.J. Leonard.
  ICRA 2013
Overview

- Incremental Inference
  - Background
  - iSAM1: Matrices
  - iSAM2: Factor Graphs

- Applications
  - Ship Hull Inspection
  - Concurrent Filtering and Smoothing
  - Dense Visual SLAM
Notable Applications of iSAM

LG Electronics
LSM-100
Scanner Mouse

Spheres miniature satellites onboard ISS

Ship Hull Inspection / U.S. Navy
Autonomous In-Water Ship Hull Inspection

- Safety and security purposes
- Currently done by divers

[Hover et al., IJRR 2012]

Joint work with
  H. Johannsson, B. Englot, F. Hover, J. Leonard (MIT);
  A. Kim, P. Ozog, R. Eustice (Univ. of Michigan)
Hovering Autonomous Underwater Vehicle

Equipped with

- 8 Thrusters (full 6DOF)
- Ring laser gyro
- Sonars:
  - Doppler Velocity Log (DVL)
  - Multi-beam sonar
- Both sonars are actuated
- Camera + light strobe

HAUV 1b
Hovering Autonomous Underwater Vehicle

Equipped with
- 8 Thrusters (full 6DOF)
- Ring laser gyro
- Sonars:
  - Doppler Velocity Log (DVL)
  - Multi-beam sonar
- Both sonars are actuated
- Camera + light strobe

HULS3
Operating on Large Ships

Camera footprint

HAUV

Sonar footprint
Autonomous Underwater Inspection

• Goal
  – Ensure full coverage
  – Re-acquire targets
  – Change detection

• Localization Challenge
  – No GPS underwater
  – Long Base Line (LBL) difficult to use in harbors

• Solution: Improve navigation using onboard sonar and/or camera
  – No additional infrastructure required
Imaging and Profiling Sonar (DIDSON)

Imaging Sonar (28 deg lens)

Profiling Sonar (1 deg lens)
Imaging Sonar Registration

Frame A

Frame B

Diff time: 357.204648
dx: 0.000000
dy: 1.000000
dt: 0.000000

Score: 0.110801
Total: 78.557976
Count: 709
X: -0.417236
Y: -0.663587
T: 6.091231
CXX=1000: 0.007336
CYX=1000: 0.069000
CTT=1000: 0.003550
CXY=1000: -0.017557
CTX=1000: 0.004267
CYT=1000: -0.014364
Cur time: 13082555204.359943
Ref time: 13082555561.564591
Diff time: -357.204648
DX: 0.000000
DY: 1.000000
DT: 0.000000
Camera Registration [Univ. of Michigan]
Recent Experiments

USNS Red Cloud, Newport News (290m)

USCGC Seneca, Boston (80m)

SS Curtiss, San Diego (180m)
This is a video attachment to IJRR paper

"Advanced Perception, Navigation and Planning for Autonomous In-Water Ship Hull Inspection"

Franz S. Hover, Ryan M. Eustice, Ayoung Kim, Brendan Englot, Hordur Johannsson, Michael Kaess, and John J. Leonard

Visual SLAM with Waypoint Navigation
(2011 Apr. USCGC Seneca)

F. Hover, B. Englot, H. Johannsson, M. Kaess and J. Leonard are with the Massachusetts Institute of Technology (MIT), Cambridge, MA 02139, USA fhover, benglot, hordurj, kaess, jleonardg@mit.edu.

R. Eustice and A. Kim are with the University of Michigan, Ann Arbor, Michigan 48109, USA feustice, ayoungkg@umich.edu.
Profiling Sonar (Complex Areas)

SS Curtiss, San Diego

USCGC Seneca, Boston
Coverage Planning [Englot&Hover, ISRR 11]
Overview

• Incremental Inference
  – Background
  – iSAM1: Matrices
  – iSAM2: Factor Graphs

• Applications
  – Ship Hull Inspection
  – Concurrent Filtering and Smoothing
  – Dense Visual SLAM
Real-time Updates at High Frame Rates?

- IMU data at 100 Hz
- iSAM2 timing difficult to predict...

- Idea: Exploit Bayes tree to parallelize fast and slow updates
Navigation Community: Filtering

- Estimate current pose $X_t$
- Objective:
  
  $$\hat{X}_t = \arg\max_{X_t} p(X_t \mid Z)$$

- Update:

  $$p(X'_t \mid Z) = \int_{X_t} p(X_t, X'_t \mid Z)$$
Robotic Community: Smoothing

- Estimate all poses, current and past
- Objective:
  \[ \hat{X} = \arg \max_X p(X \mid Z) \]
- Update:
  \[ p(X' \mid Z) = p(X, X'_t \mid Z) \]

Map of Intel Labs
Parallelizing Filtering and Smoothing

- Factorization based on suitable variable ordering:
  \[ p(X | Z) = q(X) = q(X_R | X_s) q(X_s) q(X_t | X_s) \]

- Corresponding Bayes tree:

- Allows concurrent updates to filter and smoother!
Advancing the Separator

- Filter retains important states
- Ordering leads to clique that becomes new separator
Overview

• Incremental Inference
  – Background
  – iSAM1: Matrices
  – iSAM2: Factor Graphs

• Applications
  – Ship Hull Inspection
  – Concurrent Filtering and Smoothing
  – Dense Visual SLAM
Kintinuous [Whelan, Johannsson, Kaess, Leonard, McDonald, ICRA 13]

With National Univ. Ireland, Maynooth

Contributions

- Spatially extended / moving cube
- Real-time triangulation
- Color in real-time (second cube)
- Robustness: Dense color-based tracking in addition to depth-based tracking
“Robust Real-Time Visual Odometry for Dense RGB-D Mapping”
Whelan, Johannsson, Kaess, Leonard, McDonald
ICRA 2013
http://youtu.be/sxKgP2rpFzc
Kintinuous: Stairs in MIT Stata Center
Summary and Conclusion

• iSAM2
  – General nonlinear optimizer
  – Fully incremental updates in graph structure
  – Incremental ordering to retain small cliques / sparsity

• Efficient incremental inference enables many cool applications!

• Open source implementations:
  – iSAM: http://people.csail.mit.edu/kaess/isam
  – GTSAM: http://borg.cc.gatech.edu
Questions?

Variable Elimination

OmNomNom

Courtesy of Daniel Kohlsdorf (Georgia Tech)