Ph.D. Proposal: Collaborative Localization of Robots: An Optimization on Manifolds Approach

Joseph Knuth
Advisor: Dr. Prabir Barooah

October 9, 2012

Outline

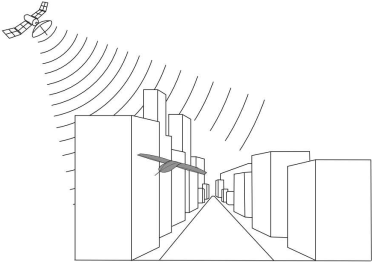
Localization is important and not always easy

Defining Localization

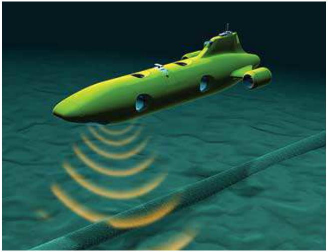
We say a robot (camera, UAV, AUV, etc.) is <u>localized</u> when an estimate of is pose (position and orientation) is available with respect to some fixed relevant reference frame.

Localization is hard when GPS measurements are not available, or only intermittently available.

(a) In urban canyon
(b) Underwater



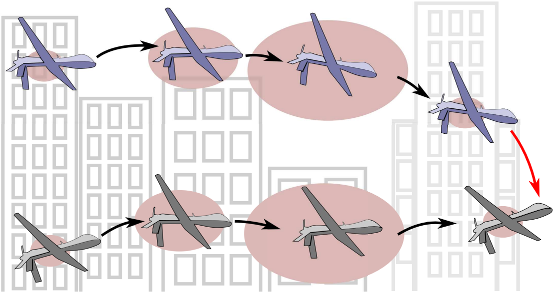
(a)



(b)

Collaborative Localization

No GPS? Try Collaboration!

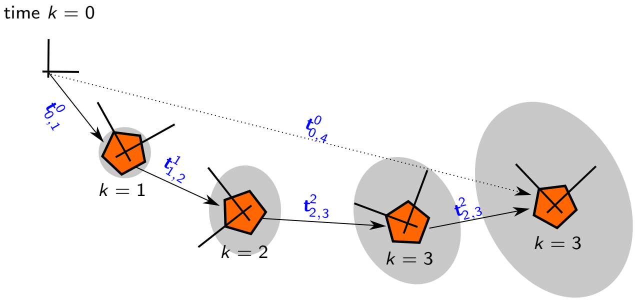


Areas of interest:

- Single Robot: How fast does uncertainty grow
- Collaborative localization methods
- Collaborative Localization: How fast does uncertainty grow

Outline

Chapter 2: Error growth in position estimation from dead reckoning



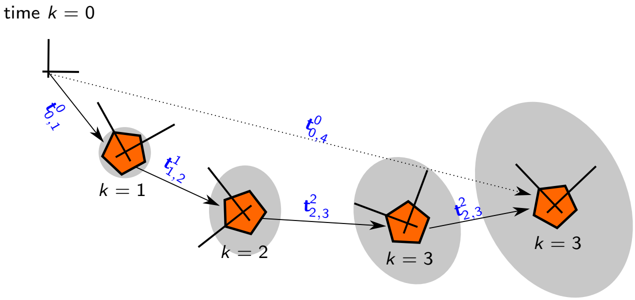
- Dead reckoning ⇒ growth in uncertainty
- But at what rate?

$$\hat{\mathbf{t}}^0_{0,1} = \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{t}}^1_{0,1}$$

$$\hat{\mathbf{t}}^0_{0,2} = \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{R}}^1_2 \, \hat{\mathbf{t}}^2_{1,2}$$

$$\hat{\mathbf{t}}^0_{0,3} = \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{R}}^1_2 \, \hat{\mathbf{t}}^2_{1,2} + \hat{\mathbf{R}}^0_1 \, \hat{\mathbf{R}}^1_2 \, \hat{\mathbf{R}}^2_3 \, \hat{\mathbf{t}}^3_{2,1}$$

Chapter 2: Error growth in position estimation from dead reckoning



- Dead reckoning $\Rightarrow$ growth in uncertainty
- But at what rate?

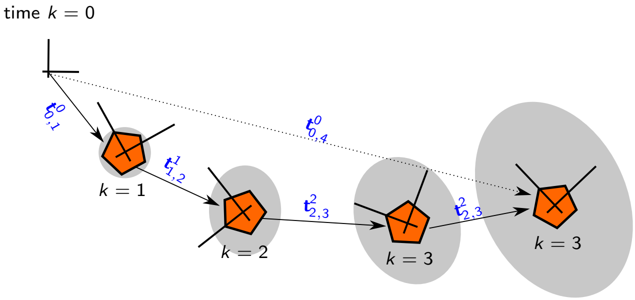
Uncertainty from: <u>1st Rotation</u>

$$\hat{\mathbf{t}}^0_{0,1} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1}$$

$$\hat{\mathbf{t}}^0_{0,2} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{t}}^2_{1,2}$$

$$\hat{\mathbf{t}}^0_{0,3} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{t}}^2_{1,2} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{R}}^2_3\ \hat{\mathbf{t}}^3_{2,1}$$

Chapter 2: Error growth in position estimation from dead reckoning



- Dead reckoning ⇒ growth in uncertainty
- But at what rate?

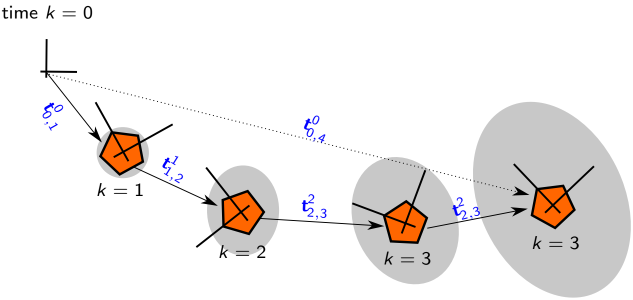
Uncertainty from: 1st Rotation 2nd Rotation

$$\hat{\mathbf{t}}^0_{0,1} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1}$$

$$\hat{\mathbf{t}}^0_{0,2} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{t}}^2_{1,2}$$

$$\hat{\mathbf{t}}^0_{0,3} = \hat{\mathbf{R}}^0_1\ \hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{t}}^2_{1,2} + \hat{\mathbf{R}}^0_1\ \hat{\mathbf{R}}^1_2\ \hat{\mathbf{R}}^2_3\ \hat{\mathbf{t}}^3_{2,1}$$

Chapter 2: Error growth in position estimation from dead reckoning



- Dead reckoning ⇒ growth in uncertainty
- But at what rate?

Uncertainty from: <u>1^{st} Rotation</u> <u>2^{nd} Rotation</u> <u>3^{rd} Rotation</u>

$$\hat{\mathbf{t}}^0_{0,1} = \hat{\mathbf{R}}^0_1\,\hat{\mathbf{t}}^1_{0,1}$$

$$\hat{\mathbf{t}}^0_{0,2} = \hat{\mathbf{R}}^0_1\,\hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\,\hat{\mathbf{R}}^1_2\,\hat{\mathbf{t}}^2_{1,2}$$

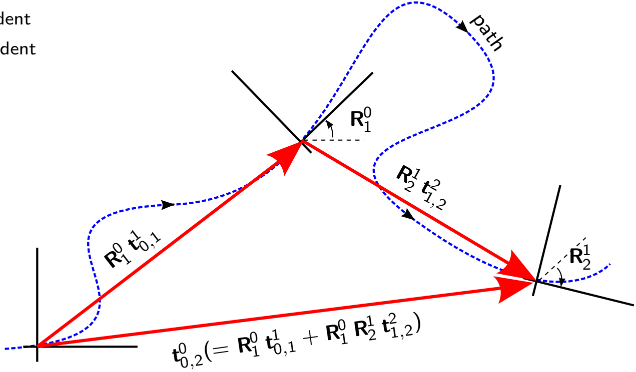
$$\hat{\mathbf{t}}^0_{0,3} = \hat{\mathbf{R}}^0_1\,\hat{\mathbf{t}}^1_{0,1} + \hat{\mathbf{R}}^0_1\,\hat{\mathbf{R}}^1_2\,\hat{\mathbf{t}}^2_{1,2} + \hat{\mathbf{R}}^0_1\,\hat{\mathbf{R}}^1_2\,\hat{\mathbf{R}}^2_3\,\hat{\mathbf{t}}^3_{2,1}$$

Assumption 1 and the Error Model

- The robot's speed is uniformly bounded.
- Independent measurements: $\forall i \neq j$
  - $\tilde{\mathbf{t}}^{i}_{i-1,i}$ and $\tilde{\mathbf{t}}^{j}_{j-1,j}$ independent
  - $\tilde{\mathbf{R}}^{i}_{i-1}$ and $\tilde{\mathbf{R}}^{j}_{j-1}$ independent
  - $\tilde{\mathbf{t}}^{i}_{i-1,i}$ and $\tilde{\mathbf{R}}^{j}_{j-1}$ independent
- $\tilde{\mathbf{R}}^{k}_{k-1}$ identically distributed and non-degenerate $\forall k$
- Additional technical assumptions on bounds of moments

Error Model:

$$\hat{\mathbf{t}}^{k}_{k-1,k} = \mathbf{t}^{k}_{k-1,k} + \tilde{\mathbf{t}}^{k}_{k-1,k}$$
$$\hat{\mathbf{R}}^{k-1}_{k} = \tilde{\mathbf{R}}^{k-1}_{k} \mathbf{R}^{k-1}_{k}$$



Main Theorem of Chapter 1: Asymptotic Bounds

Theorem

If

- *Robot moving in 3-D Euclidean space*
- *Dead reckoning used for localization*
- *Assumption 1 holds*
- $\mathbf{e}(n) := \mathbf{t}^0_{0,n} - \hat{\mathbf{t}}^0_{0,n}$

Then

1. *The bias in the position estimation error satisfies* $\|\,\mathrm{E}[\mathbf{e}(n)]\| = O(n)$*. In particular,*

$$\|\mathbf{t}^0_{0,n}\| - c_1(1-\gamma^n) \le \|\,\mathrm{E}[\mathbf{e}(n)]\| \le \|\mathbf{t}^0_{0,n}\| + c_1(1-\gamma^n) \qquad (1)$$

2. *The position error covariance satisfies* $\mathrm{Tr}\,(Cov(\mathbf{e}(n),\mathbf{e}(n))) = O(n)$*, with upper bound given by*

$$\mathrm{Tr}\,(Cov(\mathbf{e}(n),\mathbf{e}(n))) \le \overline{\alpha_0}\left(\frac{1+\gamma}{1-\gamma}\,n\right), \qquad (2)$$

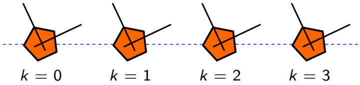
If furthermore, Cov of translation measurements is large enough, then

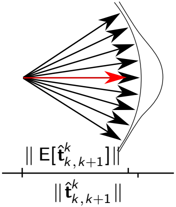
$$\mathrm{Tr}\,(Cov(\mathbf{e}(n),\mathbf{e}(n))) = \Theta(n).$$

where c_1, α_0 are functions of the statistics and $\gamma := \|\,\mathrm{E}[\tilde{\mathbf{R}}^0_1]\| < 1$.

Simple Example: Consequences

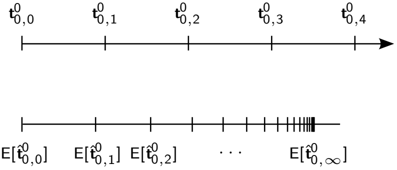
Straight Line Motion: Orientation given by $\theta^k_{k,k+1} = 0$, position $\mathbf{t}^k_{k,k+1} = [1,0]^T$ for all k





- Temporarily no error in translation
- $\hat{\mathbf{t}}^k_{k-1,k}$: correct length, wrong direction
- More uncertain direction is, the shorter $\mathrm{E}[\hat{\mathbf{t}}^k_{k-1,k}]$ becomes

True Path

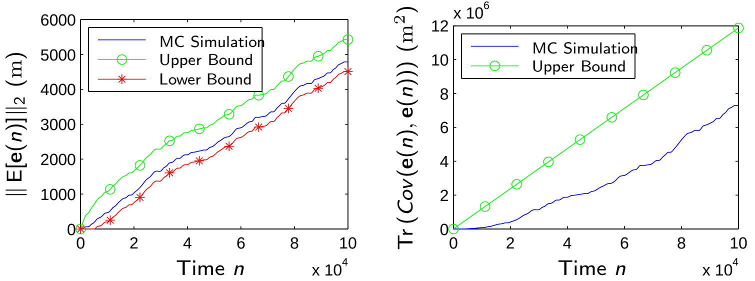


Average Estimated Path

$\| \mathrm{E}[\hat{\mathbf{t}}^0_{0,n}] \| \rightarrow$ Constant

$\| \mathrm{E}[e(n)] \| \rightarrow \| \mathbf{t}^0_{0,n} \| -$ Constant

Comparison: Theoretical bounds with Monte-Carlo Simulation



(a) (b)

- Both bias and variance growth upper bounded by $f(n) = c\,n$ for some c.
- Previously, it was stated that error growth was superlinear (Olson 2003). $O(s^{3/2})$ was claimed.
- Simulations show superlinear error growth initially. Often, $\gamma \approx 1$ and so it takes a while for geometric decay to kick in.

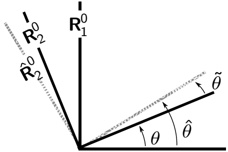
Exact Results for Straight Line & Periodic Motion

2-D Motion: Orientation given by θ, position $\in \mathbb{R}^2$

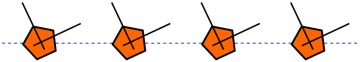
$$c := \mathsf{E}[\cos\left(\tilde{\theta} - \mathsf{E}[\tilde{\theta}]\right)]$$

$$c < 1$$

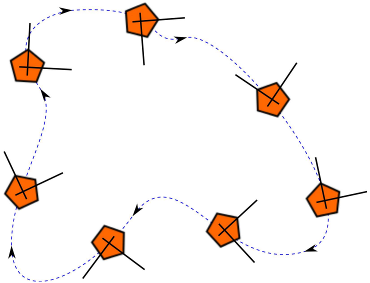
$$\underline{\mathbf{R}} := \begin{pmatrix} \cos \mathsf{E}[\tilde{\theta}] & -\sin \mathsf{E}[\tilde{\theta}] \\ \sin \mathsf{E}[\tilde{\theta}] & \cos \mathsf{E}[\tilde{\theta}] \end{pmatrix}$$



Straight Line:

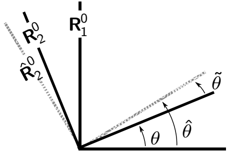


Periodic Motion:



Exact Results for Straight Line & Periodic Motion

2-D Motion: Orientation given by θ, position $\in \mathbb{R}^2$



$$c := \mathrm{E}[\cos\left(\tilde{\theta} - \mathrm{E}[\tilde{\theta}]\right)]$$

$$c < 1$$

$$\underline{\mathbf{R}} := \begin{pmatrix} \cos \mathrm{E}[\tilde{\theta}] & -\sin \mathrm{E}[\tilde{\theta}] \\ \sin \mathrm{E}[\tilde{\theta}] & \cos \mathrm{E}[\tilde{\theta}] \end{pmatrix}$$

Straight Line:

If, in addition to Assumption 1,

- Orientation and speed are constant. i.e. $\mathbf{t}_{k-1,k}^{k} = \mathbf{r}$, $\theta_k^{k-1} = 0$.
- $\tilde{\theta}$ symmetric about mean.
- $\tilde{t}$ i.i.d

Then

$$\mathrm{E}[\mathbf{e}(n)] = n\,\mathbf{r} - (I - c\underline{\mathbf{R}})^{-1}\left(I - (c\underline{\mathbf{R}})^{n}\right)\left(c\underline{\mathbf{R}}\mathbf{r} + \boldsymbol{\rho}\right),$$

$$\mathrm{Tr}\left(Cov(\mathbf{e}(n), \mathbf{e}(n))\right) = \psi n + \omega(n),$$

where the scalars ψ, $\omega(n)$ are function of the statistics and motion.

Periodic Motion:

If, in addition to Assumption 1,

- motion/statistics periodic with period p

Then

$$\mathrm{E}[\mathbf{e}(n)] = \mathbf{t}_{0,q}^{0} - \left(I - (c\underline{\mathbf{R}})^{p}\right)^{-1} \times \left(I - (c\underline{\mathbf{R}})^{\eta p}\right) w(p) - (c\underline{\mathbf{R}})^{\eta p} w(q),$$

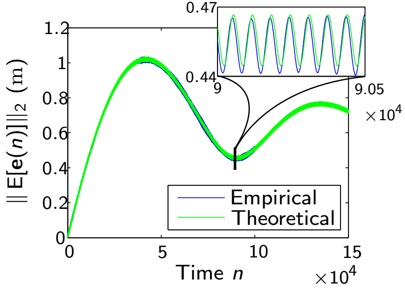
where $w(j)$ is given by

$$w(j) := \sum_{i=0}^{j-1} (c\underline{\mathbf{R}})^{i}\, \mathbf{R}_{i+1}^{0}\left(c\underline{\mathbf{R}}\ \mathbf{t}_{i,i+1}^{i+1} + \boldsymbol{\rho}_i\right)$$

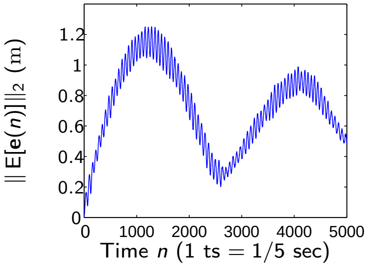
where $c, \underline{\mathbf{R}}$ are function of the motion/stat, $\eta := \lfloor n/p \rfloor$, $q := n - \eta p$.

Comparison (Circular): Monte-Carlo Simulation vs Predicted vs Experimental

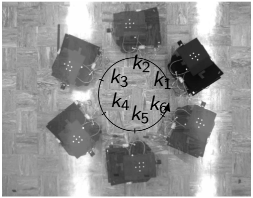
Monte-Carlo Simulation



Experimental



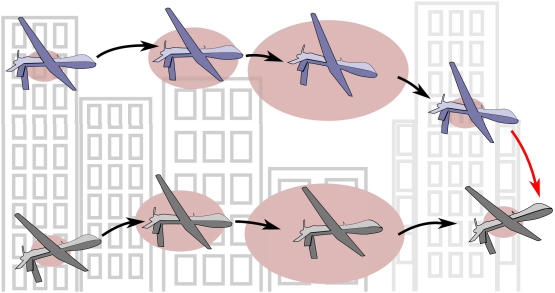
- Predictions match MC Simulation well
- The statistics of the robot could not be accurately estimated, however, the general shape matches what we predict.
- Small oscillations due to circular motion (65 ts/rotation)
- Large oscillations due to periodic motion (3020 ts/period)



Outline

Chapter 3: Collaborative localization

Consider a group of autonomous robots that lose access to GPS.



Assume:

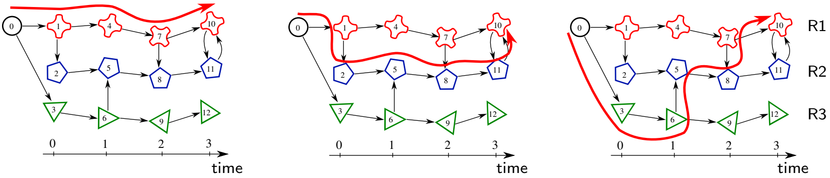
- Capable of dead reckoning (inter-time relative pose measurements)
- Pairs of robots can measure their relative pose (inter-robot relative pose measurements)
- Robots can exchange these relative pose measurements

- Dead reckoning leads to unbounded growth in localization uncertainty.
- Utilizing inter-robot relative pose measurements can help mitigate this growth.
- Existing Methods: Extended Kalman Filter, Graph Optimization*, Belief Propagation

Corresponding Graph Problem: $\mathcal{G}(k) = (\mathcal{V}(k), \mathcal{E}(k))$

(robot, time) absolute poses $\mapsto$ nodes

relative pose measurements $\mapsto$ edges



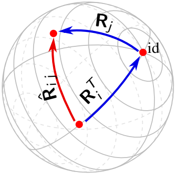
(a) (b) (c)

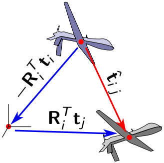
- Path (a) corresponds to the dead reckoning estimate for robot 1 at time 3.
- Path (b) and (c) are additional paths available due to the noisy inter-robot relative pose measurements.
- Each path gives a distinct estimate due to the noise in every measurement.
- Averaging over all such paths should give an improved estimate of the node variable.

Averaging through minimization of a cost function

If no noise then:

$$\hat{\mathbf{R}}_{ij} = \mathbf{R}_i^T \mathbf{R}_j$$
$$\hat{\mathbf{t}}_{ij} = \mathbf{R}_i^T (\mathbf{t}_j - \mathbf{t}_i)$$





Cost Function

$$f(\{\mathbf{R}_i, \mathbf{t}_i\}_{i \in V(k)}) := \frac{1}{2} \sum_{(i,j) \in \mathcal{E}(k)} \left(d^2(\hat{\mathbf{R}}_{ij}, \mathbf{R}_i^T \mathbf{R}_j) + \|\hat{\mathbf{t}}_{ij} - \mathbf{R}_i^T (\mathbf{t}_j - \mathbf{t}_i)\|^2 \right)$$

Riemannian Distance: $d(p, q) = \sqrt{-\frac{1}{2} \operatorname{Tr}\left(\log^2(p^{-1} q)\right)}, \ p, q \in SO(3)$

How to minimize f: Gradient Decent on a Manifold

Manifold (M)

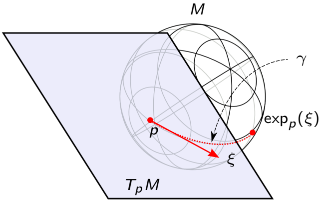
A manifold of dim. n is a topological space that is locallaly like $\mathbb{R}^n$.

Geodesic (γ)

Geodesic is a generalization of the notaiton of "straigh line". For a Riemannian manifold, a geodesic is the shortest curve between two points.

Parallel Transport ($\exp_p$)

Parallel trasport: Given a tangent vector v at a point $p \in M$, $\exp_p(v)$ is a new point $q \in M$ found by moving in the direction of v along a geodesic.



Consider a function $f : M \to \mathbb{R}$

- There is an equivalent of the vector space graident for Manifolds: grad f
- Move in the negative direction of the grandient to minimize f
- Gradient Descent: We know how to do this, see (Absil et al., 2008)
  - Update Law:
    $p_{k+1} = \exp_p(-\eta_k \text{grad} f(p))$
  - Step size: η_k comes from line search on Manifold

Gradient Decent on the Product Manifold

Our Manifold: $(SO(3) \times \mathbb{R}^3)^n$
Given a collection of manifold $M_1, M_2, \ldots, M_n$ and corresponding Riemannian metrics $< \cdot \,|\, \cdot >^1_{(\cdot)}, \ldots, < \cdot \,|\, \cdot >^n_{(\cdot)}$, define the product Riemannian manifold/metric

$$M = M_1 \times M_2 \times \cdots \times M_n \qquad < \eta \,|\, \xi >_p = \sum_{i=1}^{n} < \eta_i \,|\, \xi_i >^i_{p_i}$$

for all $p = (p_1, p_2, \ldots, p_n)$ and all $(\eta_1, \ldots, \eta_n), (\xi_1, \ldots, \xi_n) \in T_p M$.

When we define our product Riemannian manifold as above, the following fact holds.

Fact

The gradient of the cost function at $p = (\mathbf{R}_1, \mathbf{t}_1, \ldots, \mathbf{R}_n, \mathbf{t}_n) \in \left(SO(3) \times \mathbb{R}^3\right)^n$ is

$$\text{grad}\, f(p) = (\text{grad}\, f(\mathbf{R}_1), \text{grad}\, f(\mathbf{t}_1), \ldots, \text{grad}\, f(\mathbf{R}_n), \text{grad}\, f(\mathbf{t}_n))$$

where, for $i = 1, \ldots, n = |\mathcal{V}(k)|$,

$$\text{grad}\, f(\mathbf{R}_i) = \sum_{e \in \mathcal{E}(k)} \text{grad}\, g_e(\mathbf{R}_i) \qquad \text{grad}\, f(\mathbf{t}_i) = \sum_{e \in \mathcal{E}(k)} \text{grad}\, g_e(\mathbf{t}_i)$$

Gradient Decent on the Product Manifold: Update Law

Theorem

The parallel transport map at a point $p = (\mathbf{R}_1, \mathbf{t}_1, \ldots, \mathbf{R}_n, \mathbf{t}_n) \in (SO(3) \times \mathbb{R}^3)^n$*, denoted by* $\exp_p$*, is given by*

$$\exp_p(\xi) = (\mathbf{R}_1 \exp(\mathbf{R}_1^T \xi_{\mathbf{R}_1}), \mathbf{t}_1 + \xi_{\mathbf{t}_1}, \ldots, \mathbf{R}_n \exp(\mathbf{R}_n^T \xi_{\mathbf{R}_n}), \mathbf{t}_n + \xi_{\mathbf{t}_n})$$

where $\xi = (\xi_{\mathbf{R}_1}, \xi_{\mathbf{t}_1}, \ldots, \xi_{\mathbf{R}_n}, \xi_{\mathbf{t}_n})$ *is an element of the tangent space* $T_p\left[(SO(3) \times \mathbb{R}^3)^n\right] = T_{\mathbf{R}_1} SO(3) \times \cdots \times T_{\mathbf{t}_n}\mathbb{R}^3$.

To minimize the cost function, iteratively move in the direction of the negative gradient using the parallel transport map. i.e.

$$p_{k+1} = \exp_{p_k}(-\eta_k \text{grad}\, f(p_k)).$$

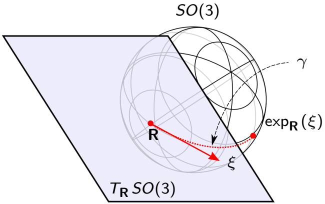
We choose η_k as the *Armijo step size*.

Minimizing the Cost Function: Gradient Decent on the Product Manifold

Input: $\mathcal{G}(k)$, measurements, inital guess
Output: $\{(\mathbf{R}_i, \mathbf{t}_i)\}_{i \in \mathcal{V}(k)}$

repeat
  foreach $i \in \mathcal{V}(k)$ **do**
    Compute grad $f(\hat{\mathbf{R}}_i)$
    Compute grad $f(\hat{\mathbf{t}}_i)$
  end
  Determine $\eta^{(A)}$, the Armijo step size
  foreach $i \in \mathcal{V}(k)$ **do**
    $\hat{\mathbf{R}}_i \rightarrow \hat{\mathbf{R}}_i \exp\left(-\eta^{(A)} \hat{\mathbf{R}}_i^T \operatorname{grad} f(\hat{\mathbf{R}}_i)\right)$
    $\hat{\mathbf{t}}_i \rightarrow \hat{\mathbf{t}}_i - \eta^{(A)} \operatorname{grad} f(\hat{\mathbf{t}}_i)$
  end
until $\|\operatorname{grad} f\| < \varepsilon$

- Parallel transport is how we move in the direction of a tangent vector.
- Parallel transport on the product manifold is the product of parallel transport on the individual manifolds.
- For $\mathbf{R} \in SO(3)$ and $\xi \in T_{\mathbf{R}} SO(3)$: $\exp_{\mathbf{R}}(\xi) = \mathbf{R} \exp(\mathbf{R}^T \xi)$.



Convergence to a critical point of the cost function is guaranteed by (Absil et al., 2008).

Heterogeneous Measurements

Previously, only inter-robot relative pose measurements were considered. We can do better.

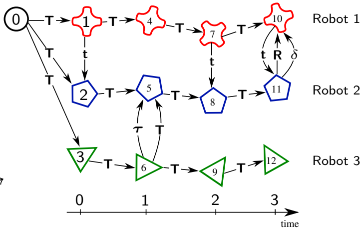
$$f(\{\mathbf{T}\}_{\mathcal{V}(k)}) := \frac{1}{2} \sum_{(i,j)=e\in\mathcal{E}(k)} g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j)$$

$$g_e(\mathbf{R}_i, \mathbf{t}_i, \mathbf{R}_j, \mathbf{t}_j) =$$

$$\begin{cases} d^2(\hat{\mathbf{R}}_{ij}, \mathbf{R}_i^T \mathbf{R}_j) & \text{if Orientation} \\ \|\hat{\mathbf{t}}_{ij} - \mathbf{R}_i^T(\mathbf{t}_j - \mathbf{t}_i)\|^2 & \text{if Position} \\ \|\left(\hat{\boldsymbol{\tau}}_{ij}\|\mathbf{t}_j - \mathbf{t}_i\|\right) - \mathbf{R}_i^T(\mathbf{t}_j - \mathbf{t}_i)\| & \text{if Bearing} \\ \|\left(\hat{\delta}_{ij} - \|\mathbf{t}_j - \mathbf{t}_i\|\right)\|^2 & \text{if Distance} \end{cases}$$

Allow inter-robot relative measurements to be any combination of the following types:

- Relative Orientation
- Relative Position
- Relative Bearing
- Relative Distance



New fully labeled, time varying graph. The edge labels now indicate the type of measurement.

Orientation Measurements

$$\text{grad}\, g_e(\mathbf{R}_h) = \begin{cases} -2\mathbf{R}_h\Big(\log(\mathbf{R}_h^T\mathbf{R}_v\hat{\mathbf{R}}_{u\,v}^T) & \text{if } h = u \\ -2\mathbf{R}_h\Big(\log(\mathbf{R}_h^T\mathbf{R}_u\hat{\mathbf{R}}_{u\,v}) & \text{if } h = v \\ 0 & \text{otherwise} \end{cases} \qquad \text{grad }\, g_e(\mathbf{t}_h) = 0.$$

Position Measurements

$$\text{grad}\, g_e(\mathbf{R}_h) = -2\mathbf{R}_h\Big(\mathbf{R}_h^T(\mathbf{t}_v - \mathbf{t}_u)\hat{\mathbf{t}}_{u\,v}^T - \hat{\mathbf{t}}_{u\,v}(\mathbf{t}_v - \mathbf{t}_u)^T\mathbf{R}_h\Big)\mathrm{I}_u(h)$$

$$\text{grad }\, g_e(\mathbf{t}_h) = 2I_{uv}(h)(\mathbf{t}_u + \mathbf{R}_u\hat{\mathbf{t}}_{u\,v} - \mathbf{t}_v).$$

Bearing Measurements

$$\text{grad }\, g_e(\mathbf{R}_h) = -2\mathbf{R}_h\Big(\mathbf{R}_h^T(\mathbf{t}_v - \mathbf{t}_h)\hat{\boldsymbol{\tau}}_{u\,v}^T\|\mathbf{t}_u - \mathbf{t}_v\| - \hat{\boldsymbol{\tau}}_{u\,v}\|\mathbf{t}_v - \mathbf{t}_u\|(\mathbf{t}_v - \mathbf{t}_u)^T\mathbf{R}_h\Big)\mathrm{I}_u(h)$$

$$\text{grad }\, g_e(\mathbf{t}_h) = -4I_{uv}(h)[(\mathbf{t}_v - \mathbf{t}_u) - \|\mathbf{t}_v - \mathbf{t}_u\|\mathbf{R}_u\hat{\boldsymbol{\tau}}_{u\,v}]$$

Distance Measurements

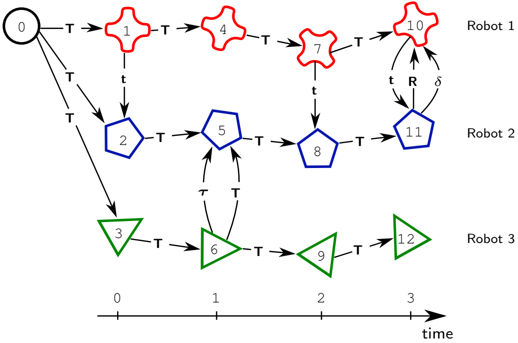
$$\text{grad}\, g_e(\mathbf{R}_h) = 0 \qquad \text{grad}\, g_e(\mathbf{t}_h) = -2I_{uv}(h)\frac{(\hat{\delta}_{u\,v} - \|\mathbf{t}_v - \mathbf{t}_u\|)}{\|\mathbf{t}_v - \mathbf{t}_u\|}(\mathbf{t}_v - \mathbf{t}_u).$$

where $I_{uv}(h) = 1$ if $h = u$, -1 if $h = v$ and 0 otherwise and I_u is an indicator function.

How to Distribute the Computation

At each time step every robot *i* performs the following:

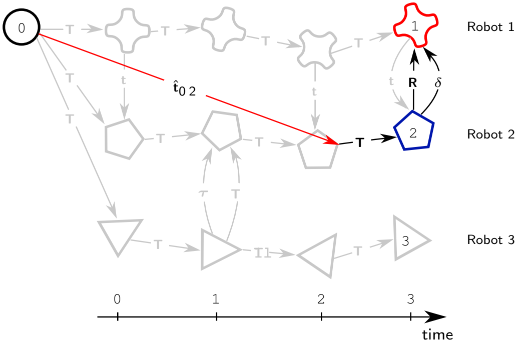
- Estimate current pose using past estimate and inter-time measurement.
- Broadcast/Receive absolute pose estimate and inter-robot measurements to/from all neighbors.
- Run the centralized algorithm on local subgraph using centralized algorithm.
- Only robot *i*'s new estimate of its global pose is retained.



How to Distribute the Computation

At each time step every robot *i* performs the following:

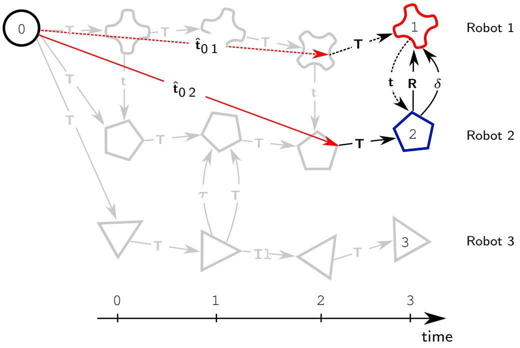
- Estimate current pose using past estimate and inter-time measurement.
- Broadcast/Receive absolute pose estimate and inter-robot measurements to/from all neighbors.
- Run the centralized algorithm on local subgraph using centralized algorithm.
- Only robot *i*'s new estimate of its global pose is retained.



How to Distribute the Computation

At each time step every robot *i* performs the following:

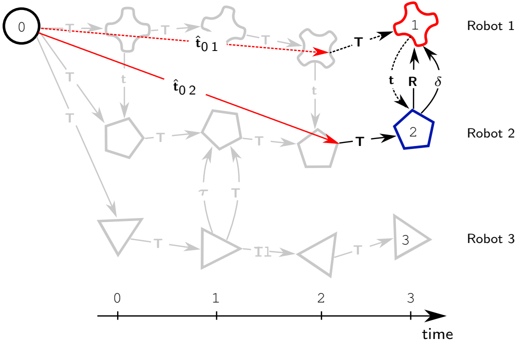
- Estimate current pose using past estimate and inter-time measurement.
- Broadcast/Receive absolute pose estimate and inter-robot measurements to/from all neighbors.
- Run the centralized algorithm on local subgraph using centralized algorithm.
- Only robot *i*'s new estimate of its global pose is retained.



How to Distribute the Computation

At each time step every robot *i* performs the following:

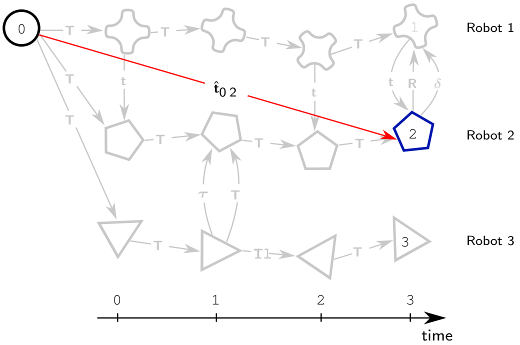
- Estimate current pose using past estimate and inter-time measurement.
- Broadcast/Receive absolute pose estimate and inter-robot measurements to/from all neighbors.
- Run the centralized algorithm on local subgraph using centralized algorithm.
- Only robot *i*'s new estimate of its global pose is retained.



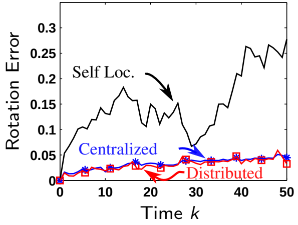
How to Distribute the Computation

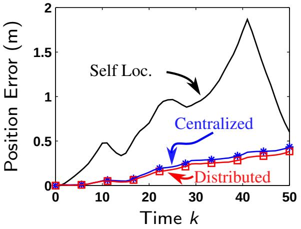
At each time step every robot *i* performs the following:

- Estimate current pose using past estimate and inter-time measurement.
- Broadcast/Receive absolute pose estimate and inter-robot measurements to/from all neighbors.
- Run the centralized algorithm on local subgraph using centralized algorithm.
- Only robot *i*'s new estimate of its global pose is retained.



Simulations: Centralized vs Distributed Algorithm



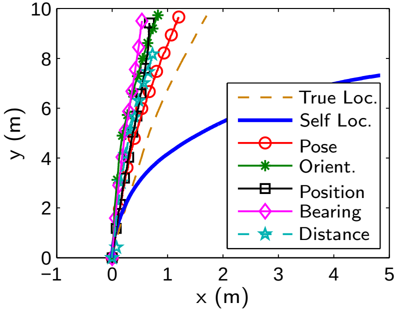


The Distributed algorithm does not solve the centralized problem, but how close can it get?
Simulation:

- Simulated 5 robots moving along random zig-zag paths.
- Single run, not Monte-Carlo.
- Surprisingly, distributed does as well as centralized.
- Both outperform dead reckoning.

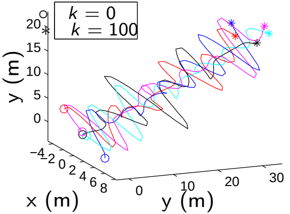
This motivates the use of the distributed algorithm for all future simulations.

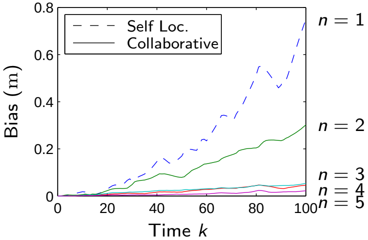
Experimental Results: Distributed Collaborative Localization



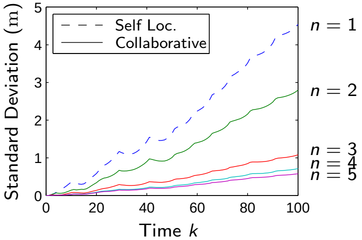
- Experiments were conducted using two Pioneer P3-DX robots equipped with cameras and targets.
- The true path (found using the overhead camera), estimated path using self localization, and estimated path using the distributed collaborative localization algorithm for each measurement type are all reported.
- A distinct improvement in localization accuracy is seen when collaborative localization is performed.

Simulations: Distributed Collaborative Localization

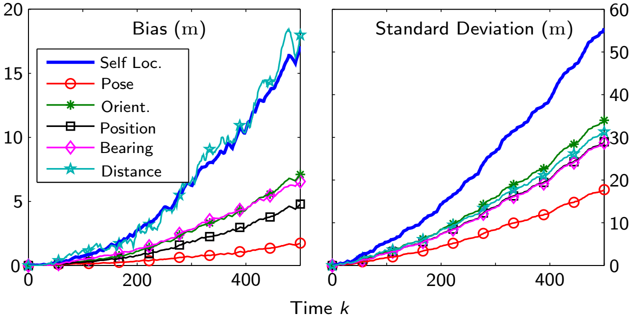




- 5 robots move along random zig-zag paths.
- Inter-robot relative pose measurements are available.
- Neighbor relations are determined by distance.
- Noise: Orientation - Von Mises-Fisher, Position - Zero-Mean Normal



Simulations: Distributed Collaborative Localization



- Simulated 5 robots utilizing various types of inter-robot relative measurements (relative pose, orientation, position, bearing and distance).
- All measurement types lead to improved estimate.
- Full relative pose performs the best (most information)
- Bearing nearly as good as full position (and much easier to get)

Outline

- Chapter 1: Motivation
- Chapter 2: Error Growth
- Chapter 3: Collaborative Localization Algorithm
- Chapter 4: Comparisons
- Chapter 6: Future Work

Chapter 4 Comparison with existing Work

Graph SLAM/Standard Pose Graph:

- Utilizes the same graph presented here.
- Requires the ability to recognize and label landmarks. (see Lu, 1997; Duckett, 2002; Olson, 2006; Grisetti, 2009)
- Cost function dependent on the parameterization of SO(3) that is chosen. (see Kummerle, 2011; Tiggs, 2000; Konolige, 2010; Lourakis, 2009)

(Extended) Kalman Filter:

- Problem becomes very complicated in $3-D$
- Linearization required (see Roumeliotis and Bekey, 2002; Karam et al., 2006; Sharma and Taylor, 2008).
- Requires absolute orientation measurements (Roumeliotis 2002, Roumeliotis and Rekleitis, 2004)
- Requires exact knowledge of the absolute orientation (see Sanderson, 1998; Barooah et al., 2010).

Method 1: Standard Pose Graph Optimization (Parameterize SO(3))

Given graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \ell)$, node variables $\{(\mathbf{R}_i, \mathbf{t}_i)\}_n$ and orientation and position measurements $\left\{\hat{\mathbf{R}}_{ij}\right\}_{m_1}$, $\{\hat{\mathbf{t}}_{ij}\}_{m_2}$.

Definitions

- $\mathbf{q}_i = q(\mathbf{R}_i)$
- $C(\mathbf{q}_i) = \mathbf{R}_i$
- $\hat{\mathbf{q}}_{ij} = q(\hat{\mathbf{R}}_{ij})$
- $\mathbf{q}_i = [\bar{\mathbf{q}}_i{}^T \; q_4]^T$
- $\mathbf{q} \boxplus \bar{\mathbf{p}} = (\mathbf{p} \otimes \mathbf{q})$

State Variable

$$X = \begin{bmatrix} \mathbf{t}_1 \\ \mathbf{q}_1 \\ \vdots \end{bmatrix}$$

Cost Function

$$f(X) = \sum_{e \in \mathcal{E}} g_e(X)^T P_e g_e(X), \;\; P_e > 0$$

$$g_e(X) = \begin{cases} \mathbf{q}_i^{-1} \otimes \mathbf{q}_j \otimes \hat{\mathbf{q}}_{ij}^{-1} - id & \ell(e) = \mathbf{R} \\ C(\mathbf{q}_i)^T(\mathbf{t}_j - \mathbf{t}_i) - \hat{\mathbf{t}}_{ij} & \ell(e) = \mathbf{t} \end{cases}$$

Minimize f: Levenberg-Marquardt

$$P = \text{diag}(P_1, P_2, \dots)$$

$$g(X) = \begin{bmatrix} g_1(X) \\ g_2(X) \\ \vdots \end{bmatrix}$$

$$J = \left[\left. \frac{\partial g_i(X \boxplus \Delta X)}{\partial \Delta X_j} \right|_{\Delta X = 0} \right]_{ij}$$

$$H = J^T P J$$

$$b = -J^T P g(X)$$

$$(H + \lambda I)\Delta X = b$$

$$X_{k+1} = X_k \boxplus \Delta X$$

Method 2: Indirect (error-state) Kalman Filter

Definitions

$$\mathbf{q}_k^i = q(\mathbf{R}_u^T) \text{ s.t. } (i,k) \mapsto u \qquad C(\mathbf{q}_k^i) = \mathbf{R}_u^T$$

$$\mathbf{q}_{k,k+1}^i = q(\hat{\mathbf{R}}_{u\,v}^T) \text{ s.t. } (i,k) \mapsto u,\ (i,k+1) \mapsto v \qquad \text{Cross}(a,b) = \lfloor a \times \rfloor b$$

State Vector

$$X_k^i = \begin{bmatrix} \mathbf{t}_k^i \\ \mathbf{q}_k^i \end{bmatrix} \qquad X_k = \begin{bmatrix} X_k^1 \\ \vdots \\ X_k^r \end{bmatrix}$$

Error States

$$\tilde{\mathbf{t}}_k^i = \mathbf{t}_k^i - \hat{\mathbf{t}}_k^i$$

$$\delta\mathbf{q}_k^i = \mathbf{q}_k^i \otimes \hat{\mathbf{q}}_k^i \approx \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta}_k^i \\ 1 \end{bmatrix}$$

Indirect KF State

$$\tilde{X}_k^i = \begin{bmatrix} \tilde{\mathbf{t}}_k^i \\ \delta\boldsymbol{\theta}_k^i \end{bmatrix} \qquad \tilde{X}_k = \begin{bmatrix} \tilde{X}_k^1 \\ \vdots \\ \tilde{X}_k^r \end{bmatrix}$$

Integrator

$$\begin{aligned} \mathbf{q}_{k+1}^i &= (\mathbf{q}_{k,k+1}^i)^{-1} \otimes \mathbf{q}_k^i \\ \mathbf{t}_{k+1}^i &= \mathbf{t}_k^i + C(\mathbf{q}_k^i)^T \mathbf{t}_{k,k+1}^i \end{aligned} \quad \rightarrow \quad \begin{aligned} \hat{\mathbf{q}}_{k+1}^i &= (\overbrace{\tilde{\mathbf{q}}_{k,k+1}^i \otimes \mathbf{q}_{k,k+1}^i}^{\hat{\mathbf{q}}_{k,k+1}^i \leftarrow \text{measurement}})^{-1} \otimes \hat{\mathbf{q}}_k^i \\ \hat{\mathbf{t}}_{k+1}^i &= \hat{\mathbf{t}}_k^i + C(\hat{\mathbf{q}}_k^i)^T \underbrace{(\mathbf{t}_{k,k+1}^i + \tilde{\mathbf{t}}_{k,k+1}^i)}_{\text{measurement} \rightarrow \hat{\mathbf{t}}_{k,k+1}^i} \end{aligned}$$

Method 2: Indirect (error-state) Kalman Filter

Linearized Error State Equations

$$\begin{aligned}\tilde{\mathbf{t}}^i_{k+1} &= \tilde{\mathbf{t}}^i_k - C(\hat{\mathbf{q}}^i_k)\lfloor\tilde{\mathbf{t}}^i_{k,k+1}\times\rfloor\delta\boldsymbol{\theta}^i_k - C(\hat{\mathbf{q}}^i_k)^T\tilde{\mathbf{t}}^i_{k,k+1}\\ \delta\boldsymbol{\theta}^i_{k+1} &= C(\hat{\mathbf{q}}^i_{k,k+1})^T\delta\boldsymbol{\theta}^i_k + C(\hat{\mathbf{q}}^i_{k,k+1})\delta\boldsymbol{\theta}^i_{k,k+1}\end{aligned} \quad\rightarrow\quad \tilde{X}_{k+1|k} = F\tilde{X}_{k|k} + G\eta$$

Measurement Model (inter-robot relative position measurements)

$$\begin{aligned} z_{ij} &= C(\mathbf{q}^i_{k+1})(\mathbf{t}^j_{k+1} - \mathbf{t}^i_{k+1}) + \xi\\ &\mathrm{E}[\xi] = 0,\ \mathrm{E}[\xi\xi^T] = R\\ \tilde{z}_{ij} &= z_{ij} - \hat{z}_{ij}\\ &\approx H\tilde{X}_{k+1|k} + \xi\end{aligned}$$

$$\begin{aligned} H &= e_i^T \otimes H_i + e_j^T \otimes H_j\\ H_i &= \begin{bmatrix} C(\hat{\mathbf{q}}^i_{k+1}) & \lfloor C(\hat{\mathbf{q}}^i_{k+1})(\hat{\mathbf{t}}^j_{k+1} - \hat{\mathbf{t}}^i_{k+1})\times\rfloor\end{bmatrix}\\ H_j &= \begin{bmatrix} C(\hat{\mathbf{q}}^i_{k+1}) & 0\end{bmatrix}\end{aligned}$$

Update (reset)

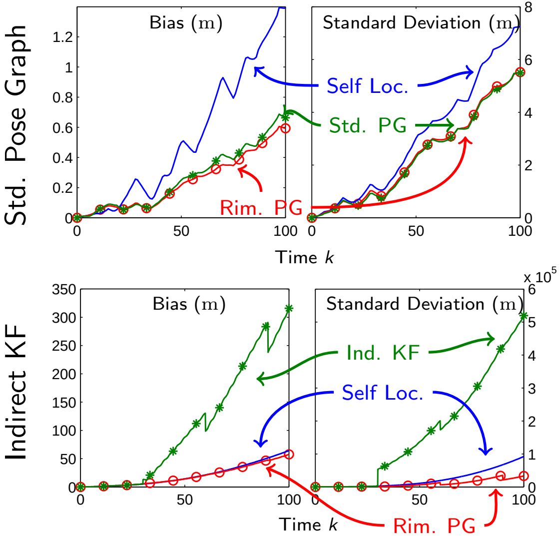
$$\begin{aligned} S &= HP_{k+1|k}H^T + R\\ K &= PH^TS^{-1}\\ \Delta X &= K\tilde{z}\end{aligned}$$

$$\Delta X = \begin{bmatrix}\Delta\mathbf{t}^1\\ \delta\boldsymbol{\theta}^1\\ \vdots\end{bmatrix}$$

$$\delta\boldsymbol{\theta}^i \xrightarrow{\text{1-1}} \delta\mathbf{q}^i$$

$$\begin{aligned}\hat{\mathbf{q}}^i_{k+1|k+1} &= \delta\mathbf{q}^i \otimes \hat{\mathbf{q}}^i_{k+1|k}\\ \hat{\mathbf{t}}^i_{k+1|k+1} &= \hat{\mathbf{t}}^i_{k+1|k} + \Delta\mathbf{t}^1\end{aligned}$$

Simulations: Comparison vs State of the Art



- Accurate orientation measurements ⇒ Std. PG better.
- Inaccurate orientation measurements ⇒ Rim. PG better.
- Frequent measurements ⇒ KF better (Accurate Cov leads to better est.)
- Infrequent measurements ⇒ Rim. PG better. (small angle approximation violated)

Summary

Advantages

- Directly applicable to 3-D pose estimation
- Able to handle a time-varying neighbor relationship in the distributed setting
- Solution independent of parameterization of SO(3)
- Able to utilize heterogeneous measurement types (of the relative position, orientation, bearing, distance, or any combination thereof)
- Useful when time between measurements is large, or error in orientation measurements is large.

Disadvantages

- No way of utilizing statistical information about the measurements when available
- No indication of the accuracy of the estimate

Outline

- Chapter 1: Motivation
- Chapter 2: Error Growth
- Chapter 3: Collaborative Localization Algorithm
- Chapter 4: Comparisons
- Chapter 6: Future Work

Future Work: Maximum Likelihood

Idea: We need a systematic way to weight measurements

1. Define the probability of seeing a certain measurement given the node variables.
2. Density w.r.t. orientation measurements must be defined on the $SO(3)$ manifold.
3. Density w.r.t. bearing measurements must be defined on S^2.
4. Cost function given by negative log-likelihood function.
5. Minimizing cost function gives max likelihood est. of node variables.

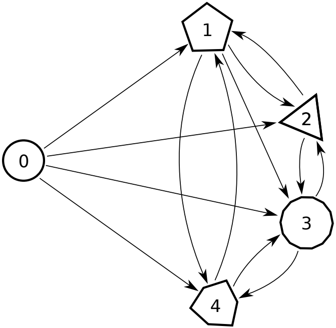
Possible density for $SO(3)$: Wrapped Gaussian Distribution

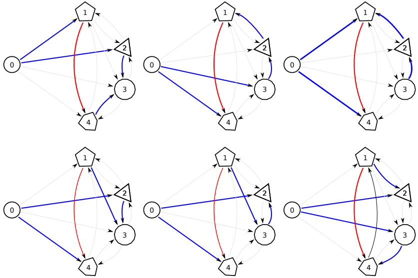
- Gaussian-like: Solution to heat equation on SO(2)
- (Approximate) max likelihood problem tractable
- Observations:
  - Single mode at mean
  - Distribution for axis of rotation uniform
  - Does not (necessarily) give rise to normally distributed parameterization

Future Work: Outlier Rejection

Goal: Given enough redundant measurements, reject outliers.
Idea: Rejecting Pose/Orientation Measurements

- Consider a random subset of all cycles.
- Cost of a cycle given by $d(id, \hat{\mathbf{R}}_{ij} \ldots \hat{\mathbf{R}}_{kj})$.
- Cost of and edge e given by $min\{\text{cost of cycle c} \mid e \in c\}$.
- Assume edge costs are Normally distributed and apply Grubbs' test for outliers.





Education

Education:

PhD Mechanical Engineering, University of Florida — 2011 - Present
Department of Mechanical and Aerospace Engineering, Gainesville, FL.

MS Mechanical Engineering, University of Florida — 2008 - 2010
Department of Mechanical and Aerospace Engineering, Gainesville, FL.

BS Computer Engineering, University of Illinois at Urbana-Champaign — 2003 - 2007
Department of Electrical and Computer Engineering, Urbana, IL.

Areas of study:

Major: Nonlinear/Adaptive Control Theory, Stochastic Control, Robot Geometry, Dynamics, Random Dynamical Systems
Mathematics: Analysis, Measure Theory, Probability Theory, Partial Differential Equations, Optimal Estimation, Statistics, Differential Geometry

Publications

Journal Articles:

- J. Knuth and P. Barooah, "Error Growth in Position Estimation from Noisy Relative Pose Measurements." submitted to Robotics and Autonomous Systems, 2012
- J. Knuth and P.Barooah, "Distributed Collaborative 3D Pose Estimation of Robots from Heterogeneous Relative Measurements: an Optimization on Manifold Approach." submitted to International Journal of Robotics Research, 2012

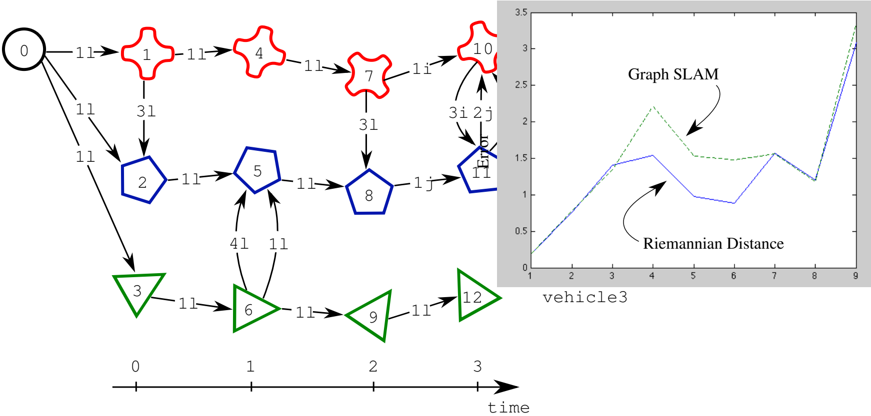
In Conference:

- J. Knuth and P. Barooah, "Maximum-likelihood localization of a camera network from heterogeneous relative measurements", submitted to American Control Conference, 2013
- J. Knuth and P. Barooah, "Collaborative localization with heterogeneous inter-robot measurements by Riemannian optimization", submitted to IEEE international Conference on Robots and Automation, 2013
- J. Knuth and P. Barooah, "Collaborative 3D localization of robots from relative pose measurements using gradient descent on manifolds", IEEE international Conference on Robots and Automation, 2012
- J. Knuth and P. Barooah, "Distributed collaborative localization of multiple vehicles from relative pose measurements", 47th Annual Allerton Conference on Communication, Control and Computing, September 30- October 2, 2009, Urbana-Champaign, IL.
- L. Erickson, J. Knuth, J. OKane, and S. LaValle, Probabilistic localization with a blind robot, in iEEE International Conference on Robotics and Automation, pp. 18211827, May 2008.

Additional Slides

- GS vs Riemannian Dist.
- Tangent Plane
- Inner-Product
- Matrix Exponential
- Armijo Step Size
- $S(1)$ - The Circle
- Parameterizations of $SO(3)$

Graph SLAM Cost Function vs. Riemannian Dist. Cost Function



Tangent Planes

Definition

Given a manifold M, the tangent plane T_pM at a point $p \in M$ consists of vectors $\xi \in T_pM$ s.t.

- $\xi : C^\infty(M) \to \mathbb{R}$
- ξ acts as a derivation of $C^\infty(M)$ evaluated at p
  i.e. for $f, g \in C^\infty(M)$

$$\xi(fg) = (\xi f)g(p) + f(p)(\xi g)$$

Definition (alt)

Let $\gamma : [0,1] \to M$ be a parameterized path on M s.t. $\gamma(0) = p \in M$. Then $\frac{d}{dt}\gamma(t)|_{t=0} \in T_pM$. Considering all such paths characterizes T_pM.

Inner-Product

An inner-product space $\mathbb{H}$ is a complex vector space equipped with an inner product

$$< \mid >: \mathbb{H} \times \mathbb{H} \to \mathbb{C}$$

such that for $\alpha, \beta \in \mathbb{C}$, $x, y, u, z \in \mathbb{H}$

- $< u|\alpha x + \beta y >= \alpha < u|x > +\beta < u|y >$
- $\overline{< x|y >} =< y|x >$
- $< z|z >\geq 0 \ \ \forall z \in \mathbb{H}$ and $< z|z >= 0$ iff $z = 0$

A norm on $\mathbb{H}$ is given by $||z|| = \sqrt{< z|z >}$. If $\mathbb{H}$ is complete w.r.t to this norm, then $\mathbb{H}$ is a Hilbert Space.

The Matrix Exponential

For $X \in \mathbb{R}^{n \times n}$,

$$\exp(X) = \sum_{k=0}^{\infty} \frac{1}{k!} X^k$$

Armijo Step Size

To minimize the cost function, iteratively move in the direction of the negative gradient using the parallel transport map. i.e.

$$p_{k+1} = \exp_{p_k}(-\eta_k \operatorname{grad} f(p_k)).$$

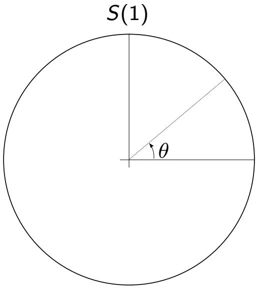
We choose η_k as the Armijo step size $\eta_t^{(A)} = \beta^{N_k}\alpha$, where N_k is the smallest nonnegative integer such that

$$f(p_k) - f(exp_{p_k}(\beta^{N_k}\alpha \operatorname{grad} f(p_k))) \geq \sigma\beta^{N_k}\alpha\|\operatorname{grad} f(p_k)\|,$$

for scalar tuning parameters $\alpha > 0$, $\beta, \sigma \in (0, 1)$.

EX: $S(1)$ - The Circle

$S(1) \subset \mathbb{R}^2$ is a 1-D manifold.

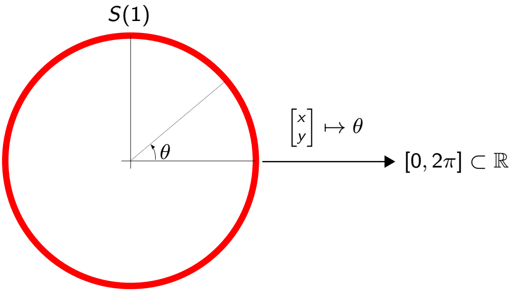


Question: What are the charts?

- If we try to use only one chart whose domain is all of S(1) we cannot find a homeomorphism.
  - $f^{-1}(0) = f^{-1}(2\pi)$
  - If we remove 2π f is not continuous.
- Instead, we break $S(1)$ into pieces (at least 2).

EX: $S(1)$ - The Circle

$S(1) \subset \mathbb{R}^2$ is a 1-D manifold.

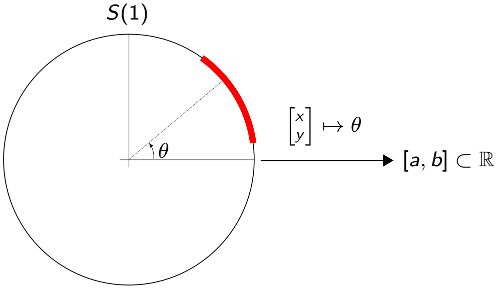


Question: What are the charts?

- If we try to use only one chart whose domain is all of S(1) we cannot find a homeomorphism.
  - $f^{-1}(0) = f^{-1}(2\pi)$
  - If we remove 2π f is not continuous.
- Instead, we break $S(1)$ into pieces (at least 2).

EX: $S(1)$ - The Circle

$S(1) \subset \mathbb{R}^2$ is a 1-D manifold.



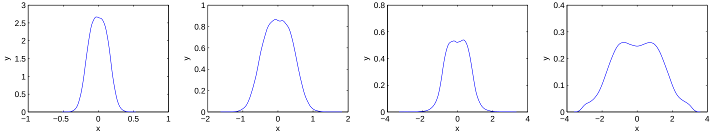
Question: What are the charts?

- If we try to use only one chart whose domain is all of S(1) we cannot find a homeomorphism.
  - $f^{-1}(0) = f^{-1}(2\pi)$
  - If we remove 2π f is not continuous.
- Instead, we break $S(1)$ into pieces (at least 2).

Comparison: Parameterizations of $SO(3)$

In each of the methods mentioned above, the space $SO(3)$ must be represented by a map to $\mathbb{R}^3$. A distribution is then assumed on this map. However this can be misleading. Consider the following example.

We generate 100,000 samples from a Wrapped Gaussian distribution on the group $SO(3)$, then use kernel density estimation to find the pdf of the Euler angles (3-2-1). The pdf for one angle for multiple variances is shown below.



(l) (m) (n) (o)

Clearly the distribution is not Gaussian, and is in fact multi-modal.

Method 1: Indirect Kalman Filter

Propagation

$X_{k+1} = f(X_k) + g(\eta_{k+1})$, $\mathrm{E}[\eta_{k+1}] = 0$, $\mathrm{E}[\eta_{k+1}\eta_{k+1}^T] = Q$

$\hat{X}_{k+1|k} = f(\hat{X_{k|k}})$

KF State: $\tilde{X}_{k+1|k} = X_{k+1} + \hat{X}_{k+1|k}$

Linearized SS Model: $\tilde{X}_{k+1|k} \approx F\tilde{X}_{k|k} + G\eta_{k+1}$

$P_{k+1|} = FP_{k|k}F^T + GQG^T$

Note:

(i) $\hat{\tilde{X}}_{0|0} = 0$ by assumption.

(ii) $\hat{\tilde{X}}_{k|k} = 0 \Rightarrow \hat{\tilde{X}}_{k+1|k} = 0$.

Update

$z = h(X_{k+1}) + \xi_{k+1}$, $\mathrm{E}[\xi_{k+1|k}] = 0$, $\mathrm{E}[\xi_{k+1}\xi_{k+1}^T] = R$

$\tilde{z} = z - \hat{z} \approx H\tilde{X}_{k+1|k}$

$\tilde{r} = \tilde{z} - H\hat{\tilde{X}}_{k+1|k} = \tilde{z} =: r$

$S = HP_{k+1|k}H^T + R$

$K = P_{k+1|k}H^TS^{-1}$

$\Delta X = Kr$

$\hat{\tilde{X}}_{k+1|k+1} = \hat{\tilde{X}}_{k+1|k} + \Delta X$

Reset

Reset $\hat{\tilde{X}}_{k+1|k+1} = 0$

Best est of X_k

$= \hat{\tilde{X}}_{k+1|k+1} + \hat{X}_{k+1|k}$

$= 0 + \hat{X}_{k+1|k+1}$

$\Rightarrow \hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + \Delta X$

Full Assumption 1:

1. The robot's speed is uniformly bounded. More specifically, there exists a constant $\tau > 0$ such that $\|\mathbf{t}^k_{k-1,k}\| \leq \tau$.
2. The translation measurement errors $\tilde{\mathbf{t}}^k_{k-1,k}$ form a sequence of independent random vectors, with mean $\mathbf{b}_k := \mathrm{E}[\tilde{\mathbf{t}}^k_{k-1,k}]$ and covariance $\mathbf{P}_k := Cov(\tilde{\mathbf{t}}^k_{k-1,k}, \tilde{\mathbf{t}}^k_{k-1,k})$ that are uniformly bounded. That is, there exist scalar constants b, $\underline{p}$, $\overline{p}$ such that $0 \leq \|\mathbf{b}_k\| \leq b$ and $0 \leq \underline{p} \leq \mathrm{Tr}\left(\mathbf{P}_k\right) \leq \overline{p} < \infty$ for all k.
3. The rotation measurement errors $\tilde{\mathbf{R}}^k_{k+1}$ form a sequence of independent random matrices. The rotation and translation measurement errors $\tilde{\mathbf{R}}^{j-1}_j$ and $\tilde{\mathbf{t}}^k_{k-1,k}$ are mutually independent if $j \neq k$, and possibly dependent when $j = k$, with $\mathrm{E}[\tilde{\mathbf{R}}^{k-1}_k \; \tilde{\mathbf{t}}^k_{k-1,k}] =: \boldsymbol{\rho}_k \in \mathbb{R}^d$. There exists a scalar ρ such that $\|\boldsymbol{\rho}_k\| \leq \rho$ for all k.
4. The relative translation measurement errors $\{\tilde{\mathbf{t}}^k_{k-1,k}\}^\infty_{k=1}$ are uniformly absolutely integrable, i.e., there exists a scalar β so that $\beta_k \leq \beta < \infty$ for all k where $\beta_k := \mathrm{E}\,\|\tilde{\mathbf{t}}^k_{k-1,k}\|$.
5. The rotation measurement errors $\tilde{\mathbf{R}}^k_{k+1}$ are identically distributed, so that each $\tilde{\mathbf{R}}^k_{k+1}$ has the same distribution as that of some matrix $\tilde{\mathbf{R}} \in SO(d)$, $d \in \{2, 3\}$. Moreover, $\tilde{\mathbf{R}}$ is not degenerate, i.e., its pdf (probability distribution function) is not concentrated on a set of measure zero.

$$\overline{\alpha_0} = \max \left\{ (\tau^2 + 2\tau b + \overline{p} + b^2), (\tau + \frac{\beta}{\gamma})(\tau + b) \right\}. \tag{3}$$

$$c = \frac{\gamma\tau + \beta}{1 - \gamma} \tag{4}$$

Very Large:

$$\underline{p} \geq 2b\tau + \tau^2 + 2\frac{(\tau + \rho/\gamma)(\tau + b)}{1 - \gamma}, \tag{5}$$

$$\psi = 2c\mathbf{r}^T \left(I - c\underline{\mathbf{R}}\right)^{-1} \underline{\mathbf{R}}\mathbf{r} + \text{Tr}\left(\mathbf{P} + \mathbf{b}\mathbf{b}^T\right) + (2\mathbf{b}^T + \mathbf{r}^T)(I - c\underline{\mathbf{R}})^{-1}\boldsymbol{\rho}$$

$$\begin{aligned}\omega(n) = {} & \mathbf{r}^T(I - c\underline{\mathbf{R}})^{-2}\left(I - 4c\underline{\mathbf{R}} + 2(c\underline{\mathbf{R}})^2 + 2(c\underline{\mathbf{R}})^{n+1}\right)\mathbf{r} - 2\mathbf{b}^T(I - c\underline{\mathbf{R}})^{-2}\left(I - (c\underline{\mathbf{R}})^n\right)\boldsymbol{\rho} \\ & + \mathbf{b}^T(I - c\underline{\mathbf{R}})^{-1}\left[I - (c\underline{\mathbf{R}})^n\right]\mathbf{r} - \mathbf{r}^T(I - c\underline{\mathbf{R}})^{-2}\left[I - (c\underline{\mathbf{R}})^n\right]\boldsymbol{\rho} \\ & - \left\| \left[(I - c\underline{\mathbf{R}})^{-1}(I - (c\underline{\mathbf{R}})^n)(c\underline{\mathbf{R}}\mathbf{r} + \boldsymbol{\rho})\right] \right\|_2^2\end{aligned}$$

Method 1: Indirect Kalman Filter

Propagation

$X_{k+1} = f(X_k) + g(\eta_{k+1})$, $\mathrm{E}[\eta_{k+1}] = 0$, $\mathrm{E}[\eta_{k+1}\eta_{k+1}^T] = Q$

$\hat{X}_{k+1|k} = f(\hat{X_{k|k}})$

KF State: $\tilde{X}_{k+1|k} = X_{k+1} + \hat{X}_{k+1|k}$

Linearized SS Model: $\tilde{X}_{k+1|k} \approx F\tilde{X}_{k|k} + G\eta_{k+1}$

$P_{k+1|} = FP_{k|k}F^T + GQG^T$

Note:

(i) $\hat{\tilde{X}}_{0|0} = 0$ by assumption.

(ii) $\hat{\tilde{X}}_{k|k} = 0 \Rightarrow \hat{\tilde{X}}_{k+1|k} = 0$.

Update

$z = h(X_{k+1}) + \xi_{k+1}$, $\mathrm{E}[\xi_{k+1|k}] = 0$, $\mathrm{E}[\xi_{k+1}\xi_{k+1}^T] = R$

$\tilde{z} = z - \hat{z} \approx H\tilde{X}_{k+1|k}$

$\tilde{r} = \tilde{z} - H\hat{\tilde{X}}_{k+1|k} = \tilde{z} =: r$

$S = HP_{k+1|k}H^T + R$

$K = P_{k+1|k}H^T S^{-1}$

$\Delta X = Kr$

$\hat{\tilde{X}}_{k+1|k+1} = \hat{\tilde{X}}_{k+1|k} + \Delta X$

Reset

Reset $\hat{\tilde{X}}_{k+1|k+1} = 0$

Best est of X_k

$= \hat{\tilde{X}}_{k+1|k+1} + \hat{X}_{k+1|k}$

$= 0 + \hat{X}_{k+1|k+1}$

$\Rightarrow \hat{X}_{k+1|k+1} = \hat{X}_{k+1|k} + \Delta X$