A Certifiably Correct Algorithm for Synchronization over the Special Euclidean Group

David M. Rosen[1] Luca Carlone[1] Afonso S. Bandeira[2]
John J. Leonard[1]

[1]Massachusetts Institute of Technology

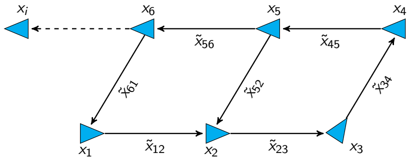
[2]New York University

Workshop on the Algorithmic Foundations of Robotics
December 18, 2016

The Special Euclidean synchronization problem

Given:

- Unknown group elements $x_1, \ldots, x_n \in \mathrm{SE}(d)$
- Noisy relative measurements $\tilde{x}_{ij} \approx x_i^{-1} x_j$

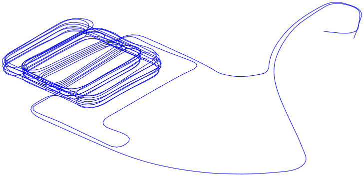


Find: An estimate $\hat{x} = (\hat{x}_1, \ldots, \hat{x}_n) \in \mathrm{SE}(d)^n$ for the hidden states

Examples: Pose-graph SLAM, camera motion estimation, sensor network localization, etc...

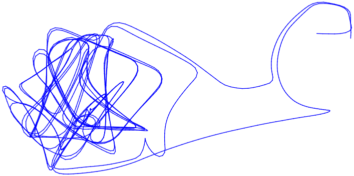
The problem: This is a *high-dimensional*, *nonconvex* maximum-likelihood estimation $\Rightarrow$ Computationally hard in general

Example

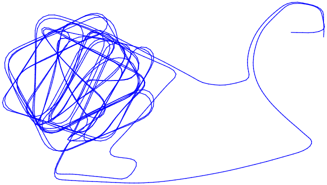


Optimal estimate

Suboptimal critical point



Suboptimal critical point



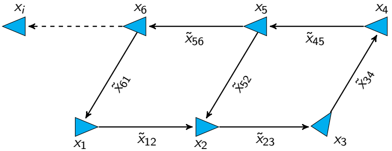
Suboptimal critical point

Contributions

In this work, we develop:

- A *convex relaxation* of SE(d) synchronization whose minimizer provides an *exact* MLE for non-adversarial noise
- A specialized, structure-exploiting optimization method to solve this relaxation efficiently

Payoff: SE-Sync, a *certifiably correct* algorithm for special Euclidean synchronization

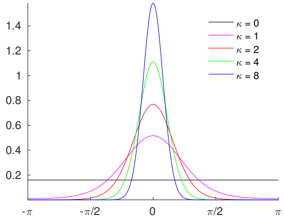


Assume the observation model:

$$\bar{t}_{ij} = R_i^T(t_j - t_i) + \delta t_{ij}, \quad \delta t_{ij} \sim \mathcal{N}(0, \tau_{ij}^{-1} I_d)$$
$$\bar{R}_{ij} = R_i^T R_j \cdot \delta R_{ij}, \quad \delta R_{ij} \sim \text{Langevin}(I_d, \kappa_{ij})$$

Here $R \sim \text{Langevin}(M, \kappa)$ means:

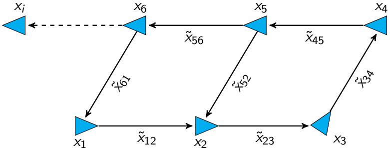
$$p(R; M, \kappa) = \frac{1}{c_d(\kappa)} \exp\left(\kappa \operatorname{tr}(M^T R)\right)$$



Forming the maximum-likelihood estimation

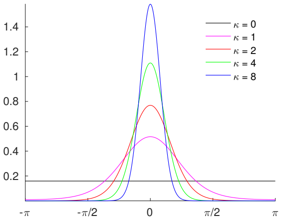
Assume the observation model:

$$\bar{t}_{ij} = R_i^T (t_j - t_i) + \delta t_{ij}, \quad \delta t_{ij} \sim \mathcal{N}(0, \tau_{ij}^{-1} I_d)$$
$$\bar{R}_{ij} = R_i^T R_j \cdot \delta R_{ij}, \quad \delta R_{ij} \sim \text{Langevin}(I_d, \kappa_{ij})$$



Here $R \sim \text{Langevin}(M, \kappa)$ means:

$$p(R; M, \kappa) = \frac{1}{c_d(\kappa)} \exp\left(\kappa \operatorname{tr}(M^\mathsf{T} R)\right)$$



Then:

$$\hat{x}_{\text{MLE}} = \underset{\substack{t_i \in \mathbb{R}^d \\ R_i \in \text{SO}(d)}}{\operatorname{argmin}} \sum_{(i,j) \in \vec{\mathcal{E}}} \kappa_{ij} \left\| R_j - R_i \tilde{R}_{ij} \right\|_F^2 + \tau_{ij} \left\| t_j - t_i - R_i \tilde{t}_{ij} \right\|_2^2$$

Simplifying the maximum-likelihood estimation

$$p^*_{\text{MLE}} = \min_{\substack{t_i \in \mathbb{R}^d \\ R_i \in \text{SO}(d)}} \sum_{(i,j) \in \vec{\mathcal{E}}} \kappa_{ij} \left\| R_j - R_i \tilde{R}_{ij} \right\|_F^2 + \tau_{ij} \left\| t_j - t_i - R_i \tilde{t}_{ij} \right\|_2^2$$

Solving for $t \triangleq (t_1, \ldots, t_n)$ in terms of $R \triangleq (R_1, \ldots, R_n)$ using a generalized Schur complement...

Simplifying the maximum-likelihood estimation

$$p^*_{\text{MLE}} = \min_{\substack{t_i \in \mathbb{R}^d \\ R_i \in \text{SO}(d)}} \sum_{(i,j) \in \vec{\mathcal{E}}} \kappa_{ij} \left\| R_j - R_i \tilde{R}_{ij} \right\|_F^2 + \tau_{ij} \left\| t_j - t_i - R_i \tilde{t}_{ij} \right\|_2^2$$

Solving for $t \triangleq (t_1, \ldots, t_n)$ in terms of $R \triangleq (R_1, \ldots, R_n)$ using a generalized Schur complement...

[Lots of algebra...]

Simplifying the maximum-likelihood estimation

$$p^*_{\text{MLE}} = \min_{\substack{t_i \in \mathbb{R}^d \\ R_i \in \text{SO}(d)}} \sum_{(i,j)\in\vec{\mathcal{E}}} \kappa_{ij} \left\| R_j - R_i \tilde{R}_{ij} \right\|_F^2 + \tau_{ij} \left\| t_j - t_i - R_i \tilde{t}_{ij} \right\|_2^2$$

Solving for $t \triangleq (t_1, \ldots, t_n)$ in terms of $R \triangleq (R_1, \ldots, R_n)$ using a generalized Schur complement:

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

$$\tilde{Q} = L(\tilde{G}^\rho) + \tilde{T}^\mathsf{T} \Omega^{\frac{1}{2}} \Pi \Omega^{\frac{1}{2}} \tilde{T}$$

Simplifying the maximum-likelihood estimation

$$p^*_{\text{MLE}} = \min_{\substack{t_i \in \mathbb{R}^d \\ R_i \in \text{SO}(d)}} \sum_{(i,j)\in\vec{\mathcal{E}}} \kappa_{ij} \left\| R_j - R_i \tilde{R}_{ij} \right\|_F^2 + \tau_{ij} \left\| t_j - t_i - R_i \tilde{t}_{ij} \right\|_2^2$$

Solving for $t \triangleq (t_1, \ldots, t_n)$ in terms of $R \triangleq (R_1, \ldots, R_n)$ using a generalized Schur complement:

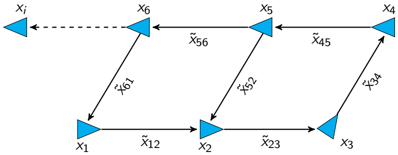
Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^{\mathsf{T}} R)$$

$$\tilde{Q} = L(\tilde{G}^\rho) + \tilde{T}^{\mathsf{T}} \Omega^{\frac{1}{2}} \Pi \Omega^{\frac{1}{2}} \tilde{T}$$

Payoffs:

- Simplified MLE is over (*compact*) *rotations* only
- Data matrices $L(\tilde{G}^\rho)$, $\tilde{T}$, Ω, Π have *simple interpretations* in G (see paper)



Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\mathrm{MLE}} = \min_{R \in \mathrm{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

$$R^\mathsf{T} R = \begin{pmatrix} I_d & * & \cdots & * \\ * & I_d & & * \\ \vdots & & \ddots & \vdots \\ * & * & \cdots & I_d \end{pmatrix} \succeq 0.$$

Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \text{tr}(\tilde{Q} R^\mathsf{T} R)$$

$$R^\mathsf{T} R = \begin{pmatrix} I_d & * & \cdots & * \\ * & I_d & & * \\ \vdots & & \ddots & \vdots \\ * & * & \cdots & I_d \end{pmatrix} \succeq 0.$$

$$\therefore \quad R^\mathsf{T} R \in \left\{ Z \in \mathcal{S}_+^{dn} \mid \text{BlockDiag}_d(Z) = (I_d, \ldots, I_d) \right\} \triangleq \mathcal{C}$$

Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\mathrm{MLE}} = \min_{R \in \mathrm{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

$$R^\mathsf{T} R = \begin{pmatrix} I_d & * & \cdots & * \\ * & I_d & & * \\ \vdots & & \ddots & \vdots \\ * & * & \cdots & I_d \end{pmatrix} \succeq 0.$$

$$\therefore \quad R^\mathsf{T} R \in \left\{ Z \in \mathcal{S}_+^{dn} \mid \mathrm{BlockDiag}_d(Z) = (I_d, \ldots, I_d) \right\} \triangleq \mathcal{C}$$

But $\mathcal{C}$ is a *spectrahedron*, and therefore **convex**

Forming the semidefinite relaxation

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \operatorname{tr}(\tilde{Q} R^\mathsf{T} R)$$

Semidefinite relaxation

$$p^*_{\text{SDP}} = \min_{Z \in \mathcal{S}_+^{dn}} \operatorname{tr}(\tilde{Q} Z)$$

$$\text{s.t.} \quad \operatorname{BlockDiag}_d(Z) = (I_d, \ldots, I_d)$$

$$R^\mathsf{T} R = \begin{pmatrix} I_d & * & \cdots & * \\ * & I_d & & * \\ \vdots & & \ddots & \vdots \\ * & * & \cdots & I_d \end{pmatrix} \succeq 0.$$

$$\therefore \quad R^\mathsf{T} R \in \left\{ Z \in \mathcal{S}_+^{dn} \mid \operatorname{BlockDiag}_d(Z) = (I_d, \ldots, I_d) \right\} \triangleq \mathcal{C}$$

But $\mathcal{C}$ is a *spectrahedron*, and therefore **convex**

$\Rightarrow$ *Expanding MLE's feasible set to* $\mathcal{C}$ gives a **convex relaxation**

The Main Idea

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \text{tr}(\tilde{Q} R^\mathsf{T} R)$$

Semidefinite relaxation

$$p^*_{\text{SDP}} = \min_{Z \in \mathcal{S}_+^{dn}} \text{tr}(\tilde{Q} Z)$$

$$\text{s.t.} \quad \text{BlockDiag}_d(Z) = (I_d, \dots, I_d)$$

Payoffs:

- $p^*_{\text{SDP}} \leq p^*_{\text{MLE}}$ (suboptimality lower bound)
- $Z^* = R^\mathsf{T} R$ with $R \in \text{SO}(d)^n \Rightarrow$ R is MLE

The Main Idea

Simplified ML estimation

$$p^*_{\text{MLE}} = \min_{R \in \text{SO}(d)^n} \text{tr}(\tilde{Q} R^\mathsf{T} R)$$

Semidefinite relaxation

$$p^*_{\text{SDP}} = \min_{Z \in \mathcal{S}^{dn}_+} \text{tr}(\tilde{Q} Z)$$

$$\text{s.t.} \quad \text{BlockDiag}_d(Z) = (I_d, \dots, I_d)$$

Payoffs:

- $p^*_{\text{SDP}} \leq p^*_{\text{MLE}}$ (suboptimality lower bound)
- $Z^* = R^\mathsf{T} R$ with $R \in \text{SO}(d)^n \Rightarrow$ R is MLE

Proposition

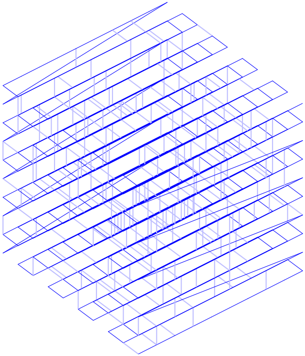
Let $\underline{Q}$ be the data matrix constructed using the true relative transforms $\underline{x}_{ij}$. There exists a constant $\beta \triangleq \beta(\underline{Q}) > 0$ such that, if $\|\tilde{Q} - \underline{Q}\|_2 < \beta$, then:

(*i*) *The semidefinite relaxation has a unique solution Z^*, and*

(*ii*) *$Z^* = {R^*}^\mathsf{T} R^*$, where $R^* \in \text{SO}(d)^n$ is a minimizer of MLE.*

Question: How do noise and problem size affect performance?

Test: Simulate random grid-world, varying κ, τ, and n:

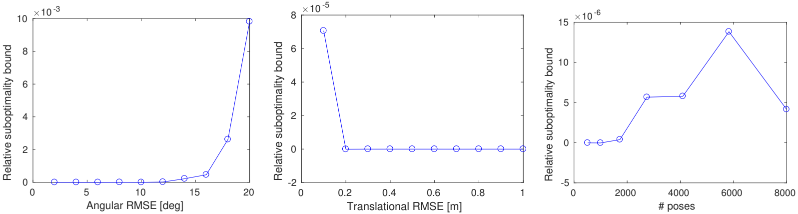


$$\bar{t}_{ij} = R_i^T (t_j - t_i) + \delta t_{ij}, \qquad \delta t_{ij} \sim \mathcal{N}(0, \tau^{-1} I_d)$$

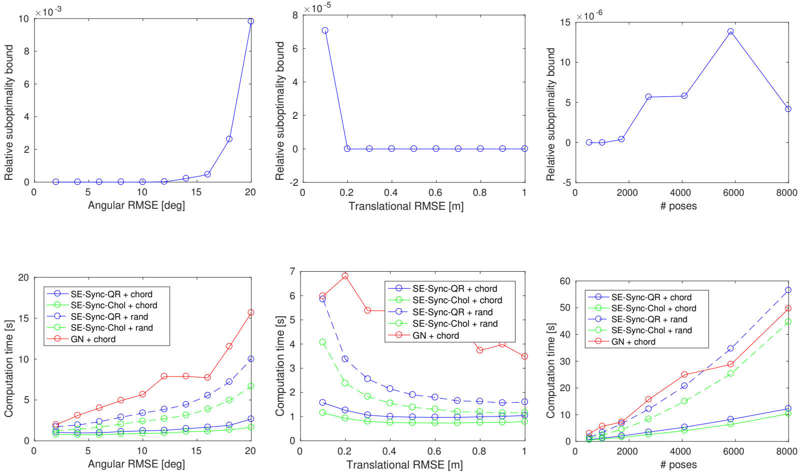
$$\bar{R}_{ij} = R_i^T R_j \cdot \delta R_{ij}, \qquad \delta R_{ij} \sim \text{Langevin}(I_d, \kappa)$$

Baseline: Gauss-Newton using chordal initialization

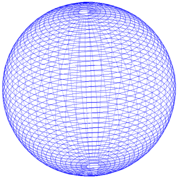
Experimental results I: Cube datasets



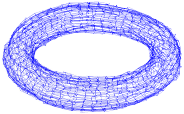
Experimental results I: Cube datasets



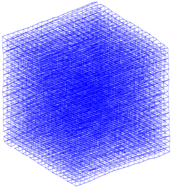
	# Poses	# Edges	Gauss-Newton Objective value	Gauss-Newton Time [s]	SE-Sync Objective value	SE-Sync Time [s]	SE-Sync Max. suboptimality
sphere	2500	4949	1.687×10^3	14.98	1.687×10^3	2.81	1.410×10^{-11}
torus	5000	9048	2.423×10^4	31.94	2.423×10^4	5.67	7.276×10^{-12}
grid	8000	22236	8.432×10^4	130.35	8.432×10^4	22.37	4.366×10^{-11}
garage	1661	6275	1.263×10^0	17.81	1.263×10^0	5.33	2.097×10^{-11}
cubicle	5750	16869	7.171×10^2	136.86	7.171×10^2	13.08	1.603×10^{-11}
rim	10195	29743	5.461×10^3	575.42	5.461×10^3	36.66	5.639×10^{-11}



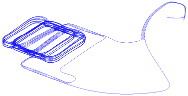
sphere



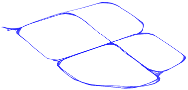
torus



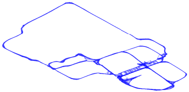
grid



garage



cubicle



rim

Conclusion

Contributions:

SE-Sync: A *certifiably correct* algorithm for SE(d) synchronization

- Recovers *globally optimal* estimates in a non-adversarial noise regime (up to 10x typical levels)
- Significantly faster than standard Gauss-Newton-based approaches (3x-15x in our experiments)

Code: `https://github.com/david-m-rosen/SE-Sync`

Future directions:

- Extension to robust estimation w/ outliers
- Generalization to *polynomial optimization problems* (e.g. monocular and range-only measurements)

Dirty laundry

Proposition

Let $\underline{Q}$ be the data matrix constructed using the true relative transforms $\underline{x}_{ij}$. There exists a constant $\beta \triangleq \beta(\underline{Q}) > 0$ such that, if $\|\tilde{Q} - \underline{Q}\|_2 < \beta$, then:

- *(i) The semidefinite relaxation has a unique solution Z^*, and*
- *(ii) $Z^* = R^{*\mathsf{T}} R^*$, where $R^* \in \mathrm{SO}(d)^n$ is a minimizer of MLE.*

But: How large is β?

- Empirically: Pretty large! (≈10x typical for laser + camera)
- Theoretically: No sharp results yet
  - Propagation of distributions difficult to control
  - Appears to be closely related to $\lambda_2(\underline{Q})$

Conclusion

Contributions:

SE-Sync: A *certifiably correct* algorithm for SE(d) synchronization

- Recovers *globally optimal* estimates in a non-adversarial noise regime (up to 10x typical levels)
- Significantly faster than standard Gauss-Newton-based approaches (3x-15x in our experiments)

Code: `https://github.com/david-m-rosen/SE-Sync`

Future directions:

- Extension to robust estimation w/ outliers
- Generalization to *polynomial optimization problems* (e.g. monocular and range-only measurements)
- *A priori bounds* on admissible noise level