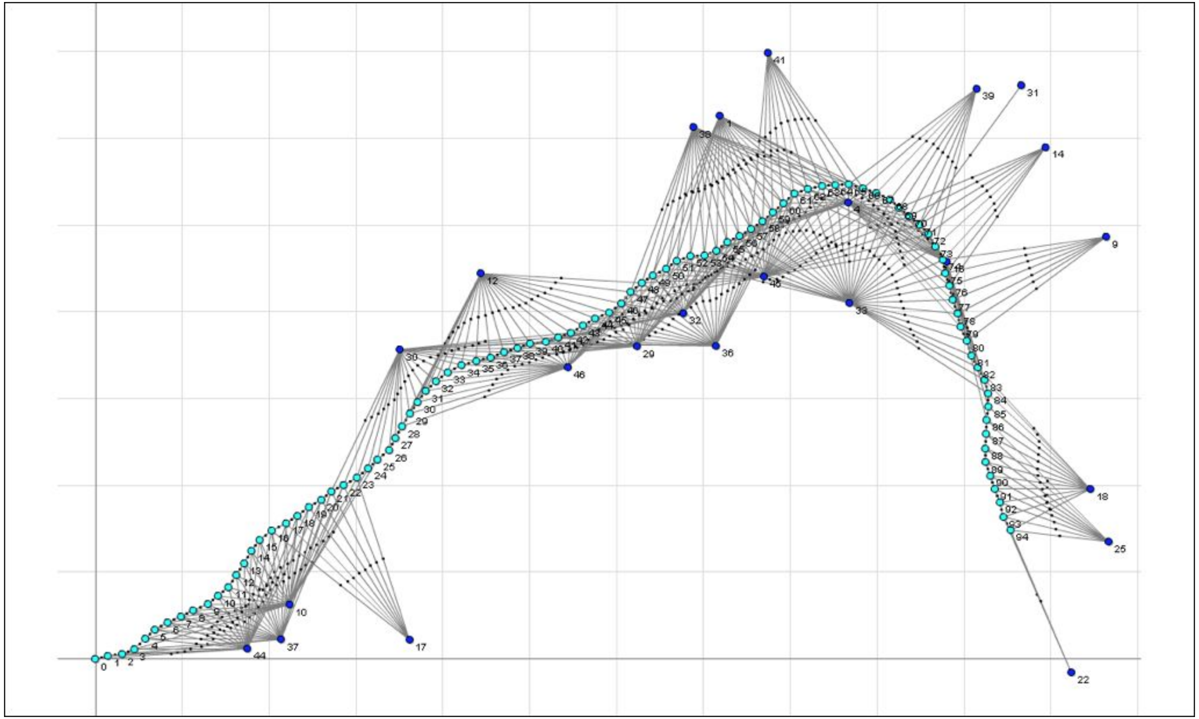
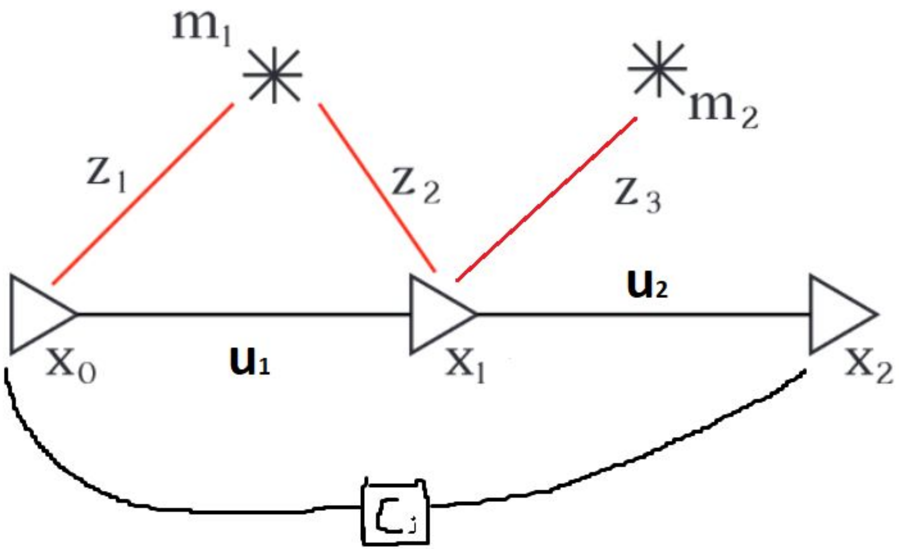
A globally consistent first estimation for them SLAM problem



Mark Griguletskii, Ivan Kudryakov. Skoltech, NLA, 2022

Problem statement

P(x_0) - priori
X_i - robot`s pose
U_i - odometry measurement
M_k - landmark
Z_k - landmark measurement
C - loop closure measurement

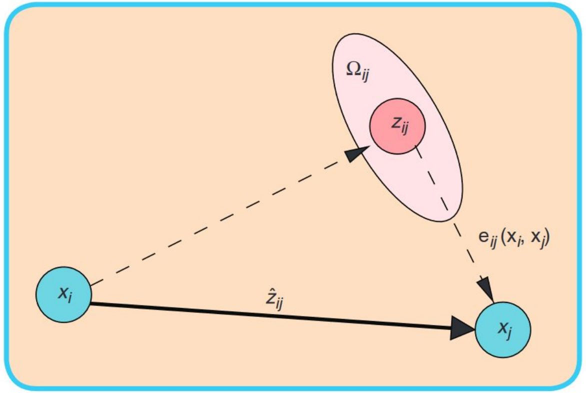


[Perception in Robotics course. Skoltech, 2019, Gonzalo Ferrer.]

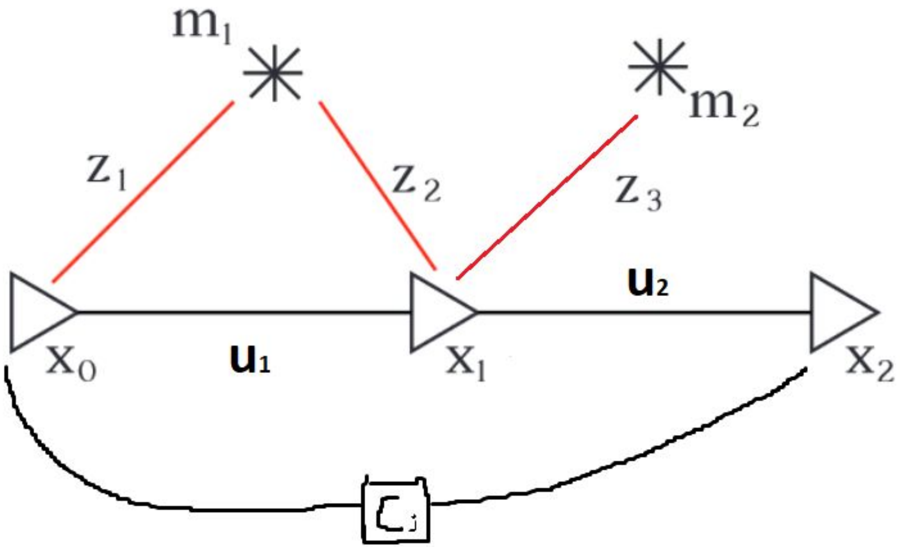
$$\mathcal{P}(\mathcal{X}, \mathcal{M}, \mathcal{Z}, \mathcal{U}) = p(x_0) \prod_{i=1}^{M} p(x_i | x_{i-1}, \mu_i) \cdot \prod^{k} p(z_k | x_{i_k}, m_{j_k})$$

$$\begin{aligned}\theta^* &= \arg\max_\theta \mathcal{P}(\mathcal{X}, \mathcal{M} | \mathcal{Z}, \mathcal{U}) = \arg\max_\theta \mathcal{P}(\mathcal{X}, \mathcal{M}, \mathcal{Z}, \mathcal{U}) \\ &= \arg\min_\theta \{-\log \mathcal{P}(\mathcal{X}, \mathcal{M}, \mathcal{Z}, \mathcal{U})\} = \\ &= \arg\min_\theta \left\{ \sum_i^M \|g_i(x_{i-1}, u_i) - x_i\|^2_{\Sigma_i} + \sum_{k=1}^{K} \|h_k(x_{i_k}, m_{j_k}) - z_k\|^2_{\Sigma_k} \right\}\end{aligned}$$

Error function linearization



A Tutorial on Graph-Based SLAM. Giorgio Grisetti; Rainer Kümmerle; Cyrill Stachniss; Wolfram Burgard



$$\mathbf{e}_{ij}(\breve{\mathbf{x}}_i + \Delta\mathbf{x}_i, \breve{\mathbf{x}}_j + \Delta\mathbf{x}_j) = \mathbf{e}_{ij}(\breve{\mathbf{x}} + \Delta\mathbf{x})$$

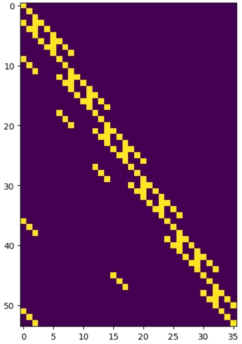
$$\simeq \mathbf{e}_{ij} + \mathbf{J}_{ij}\Delta\mathbf{x}.$$

$$g_i\left(x_{i-1}, u_i\right) - x_i \cong \left[g_i\left(x_{i-1}^0, u_i\right) + G_i^{i-1}\delta x_{i-1}\right] - \left[x_i^0 + \delta x_i\right] = \left(G_i^{i-1}\delta x_{i-1} - \delta x_i\right) - a_i$$

$$G_i^{i-1} = \frac{\partial g_i(x_{i-1}^0, u_i)}{\partial x_{i-1}} \quad |x_{i-1}$$

Adjacency matrix and solution

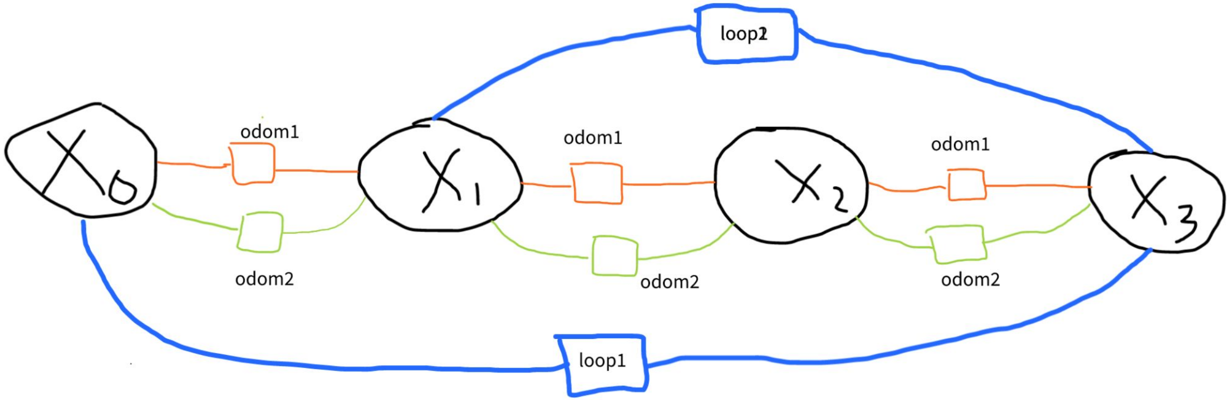
A =



	X0	X1	X2	X3
init	-I			
odom1	G	-I		
odom2	G	-I		
loop1	Hi			-Hj
odom1		G	-I	
odom2		G	-I	
odom1			G	-I
odom2			G	-I
loop2		Hi		-Hj

$$\lambda = argmin_{\delta} \sum_{i=1}^{N} \left|\left| E_{odom}^{-T/2}(G_i^{i-1}\delta x_{i-1} - I\delta x_i) - E_{odom}^{-T/2} a_i \right|\right|_2^2 +$$

$$\sum_{k=1}^{K} \left|\left| E_{loop}^{-T/2}(H_i^{i-1}\delta x_{i-1} - I\delta x_i) - E_{loop}^{-T/2} a_k \right|\right|_2^2$$

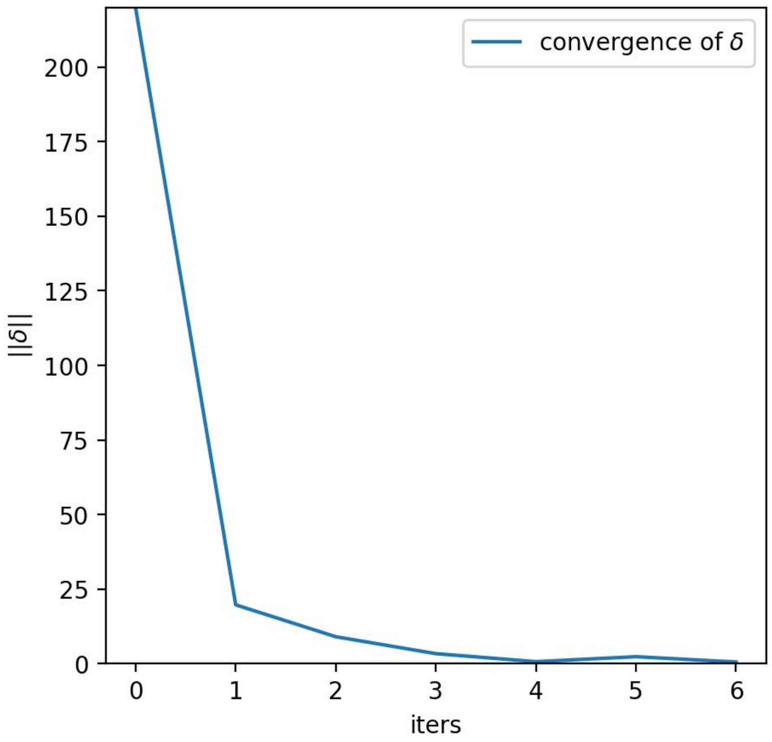


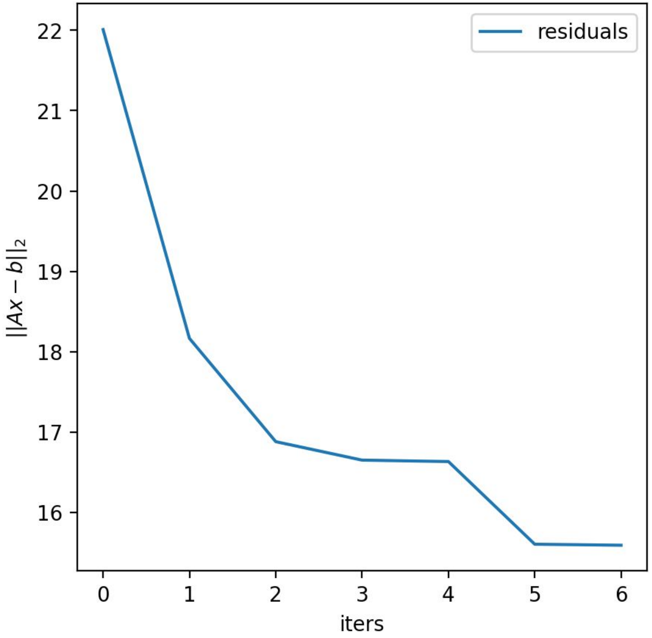
We solve linear system via QR decomposition, as it has exact solution

while $||\delta||_2 >= \epsilon$:

1) calculate A, b around X, and $A = QR$
2) solve $\delta = R^{-1}Q^T b$
3) update $X := X + \delta$
4) residual $= ||Ax - b||_2$

How does it converge ?

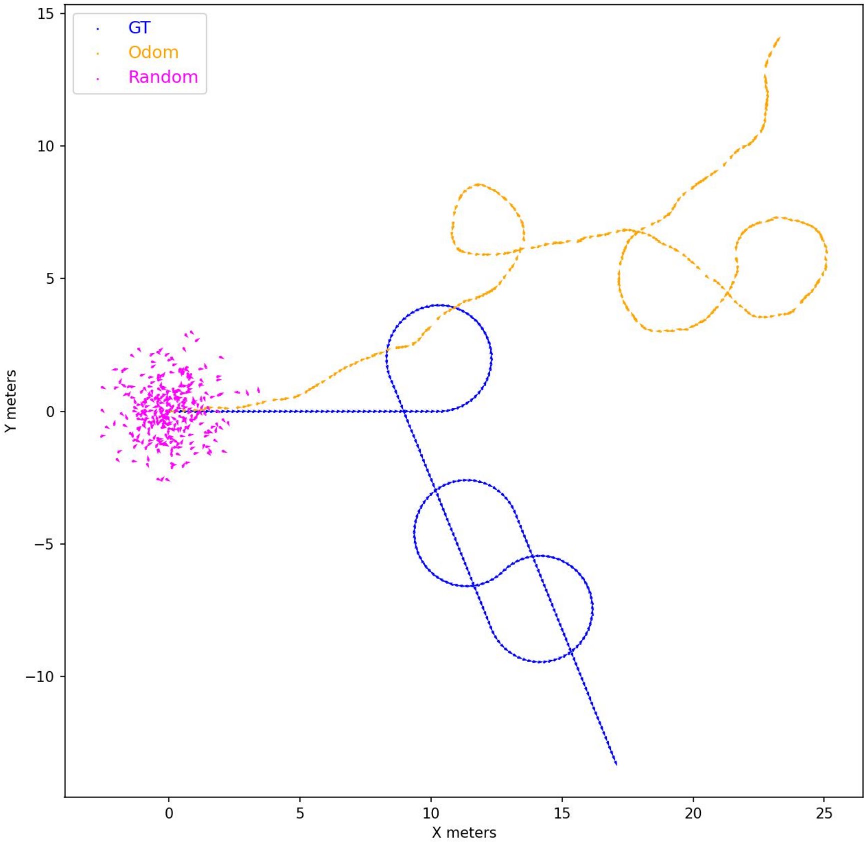


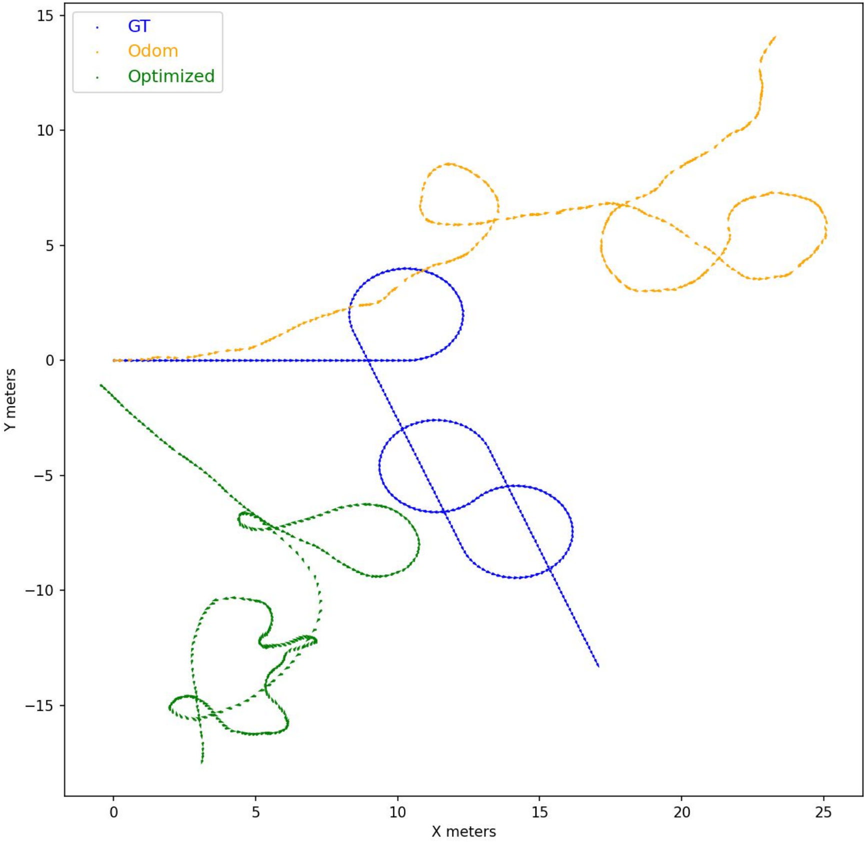


while $||\delta||_2 >= \epsilon$:

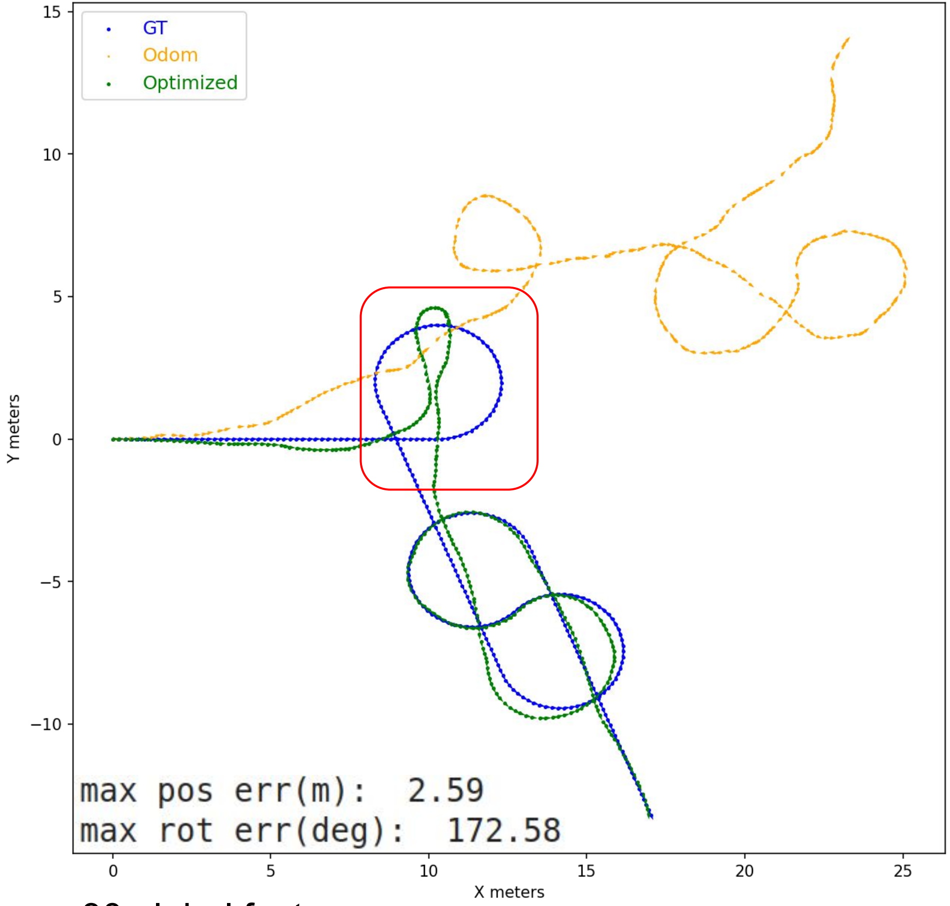
1) calculate A, b around X, and $A = QR$
2) solve $\delta = R^{-1}Q^T b$
3) update $X := X + \delta$
4) residual $= ||Ax - b||_2$

What happens if an initial estimate is random?

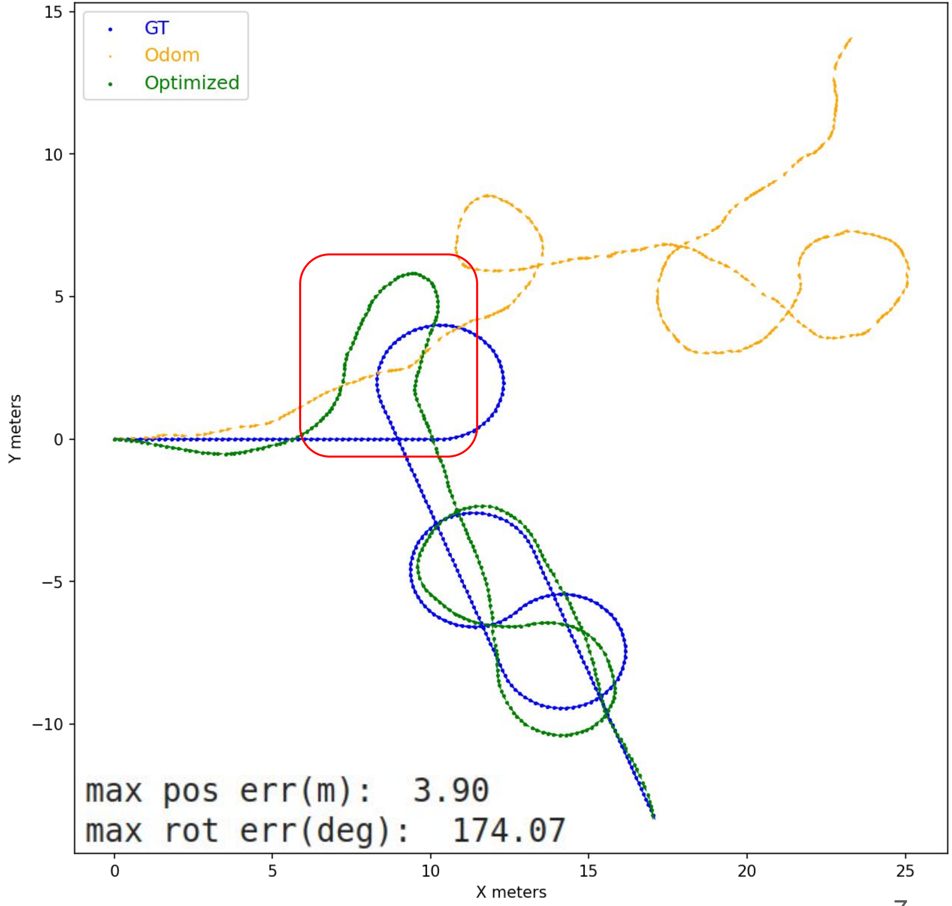




What happens if an initial estimate is zeros?

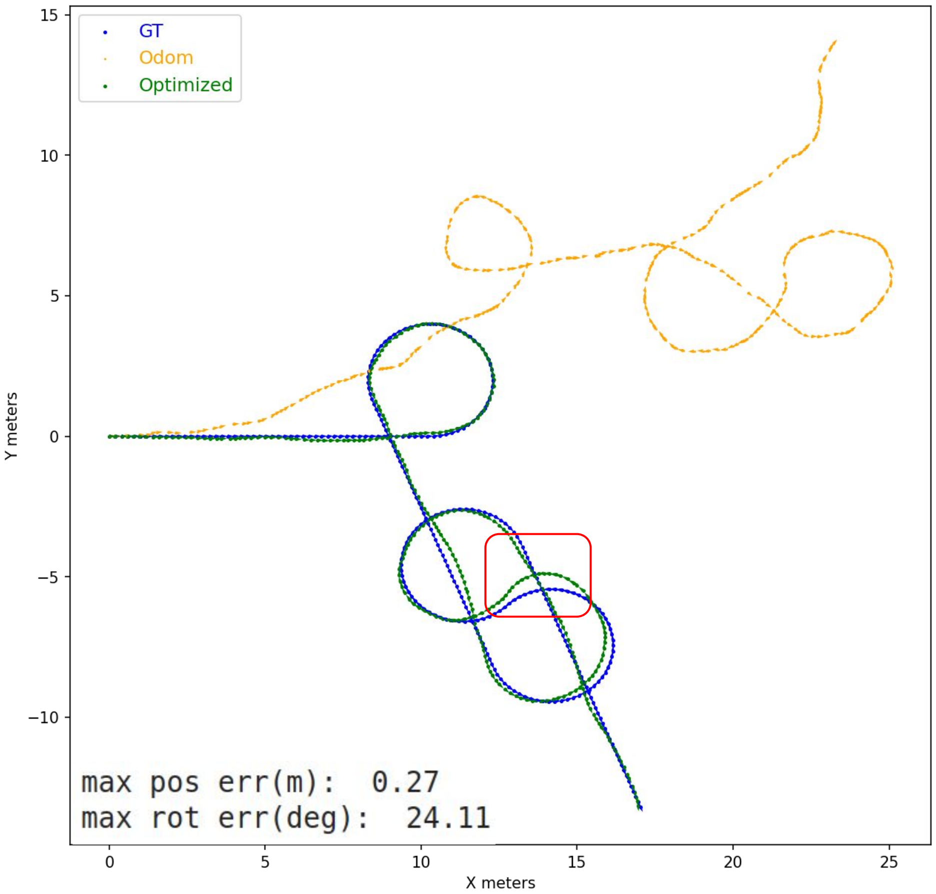


30 global factors

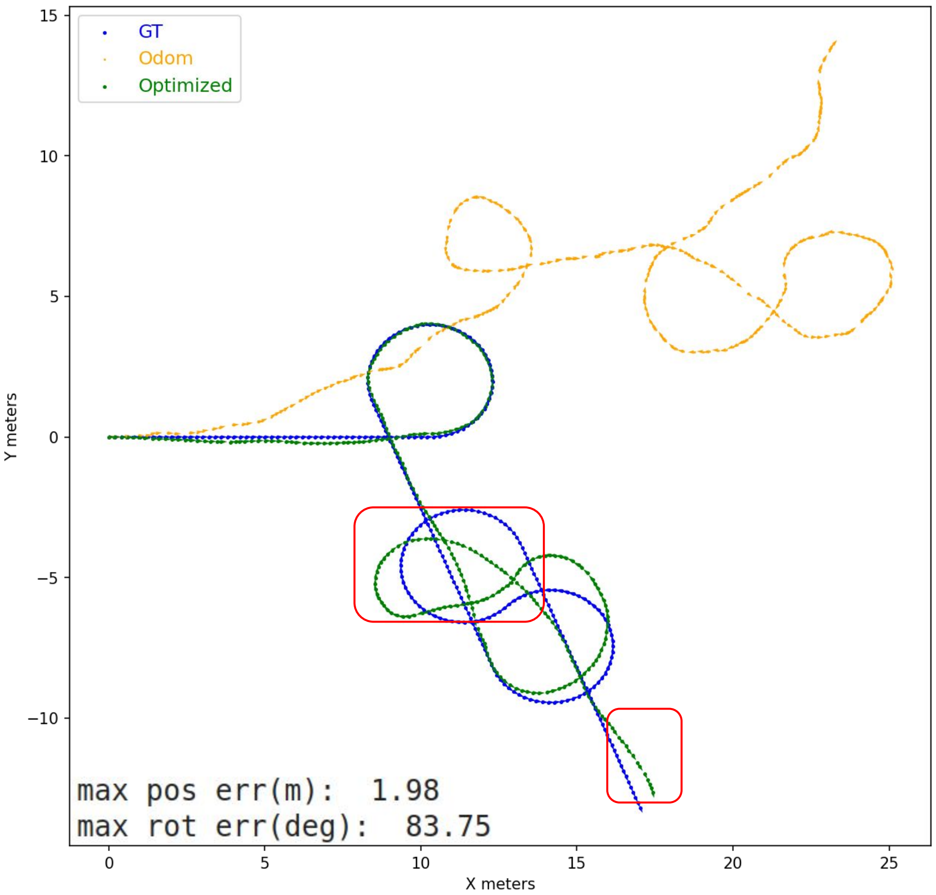


15 global factors

What happens if an initial estimate is better - noisy integration of odometry ?

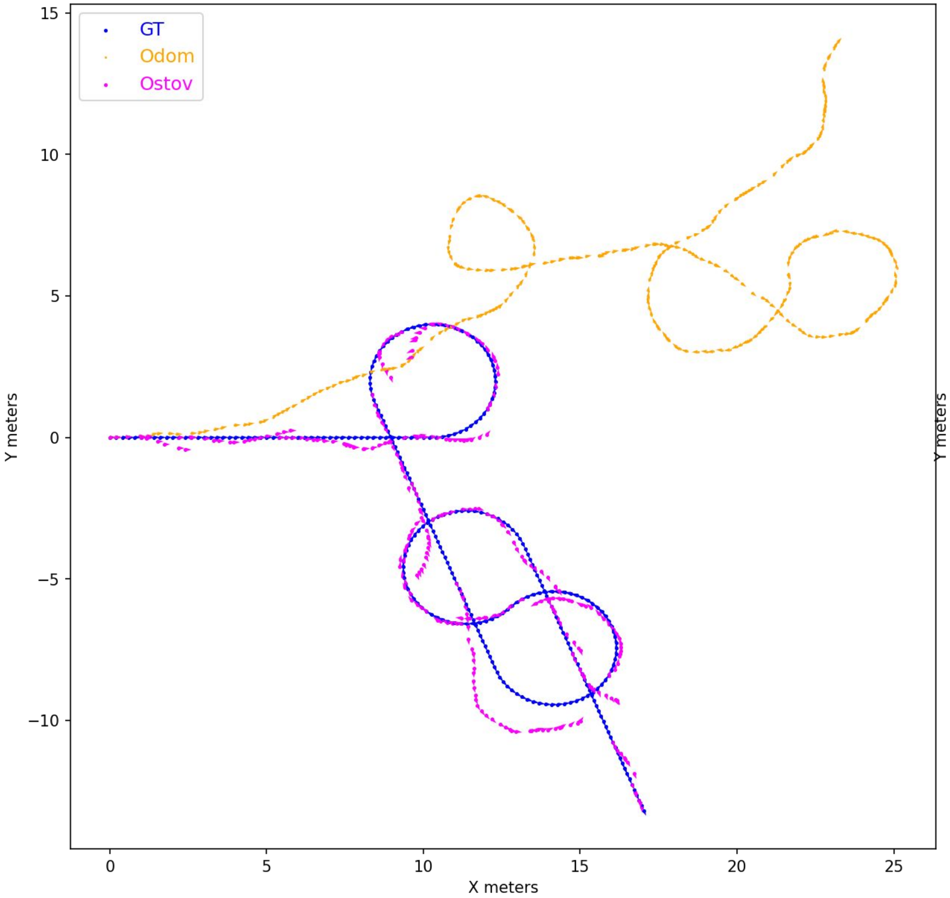


30 global factors

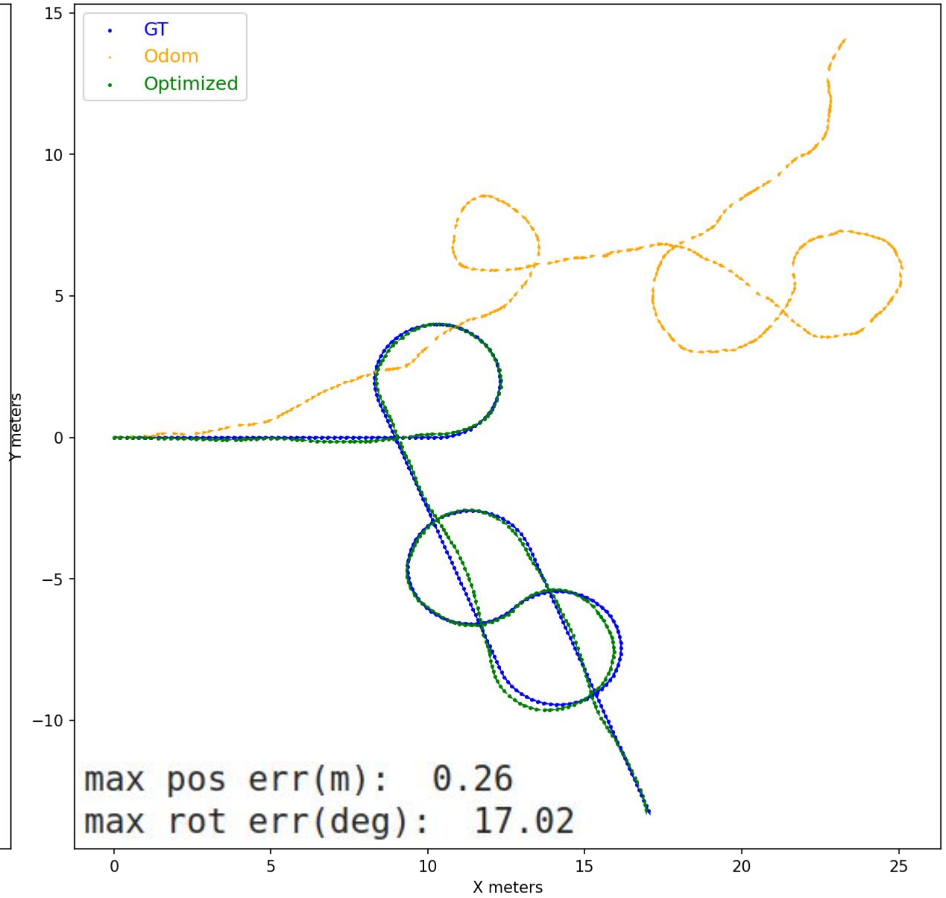


15 global factors

What happens if an initial estimate is Ostov-based solution?

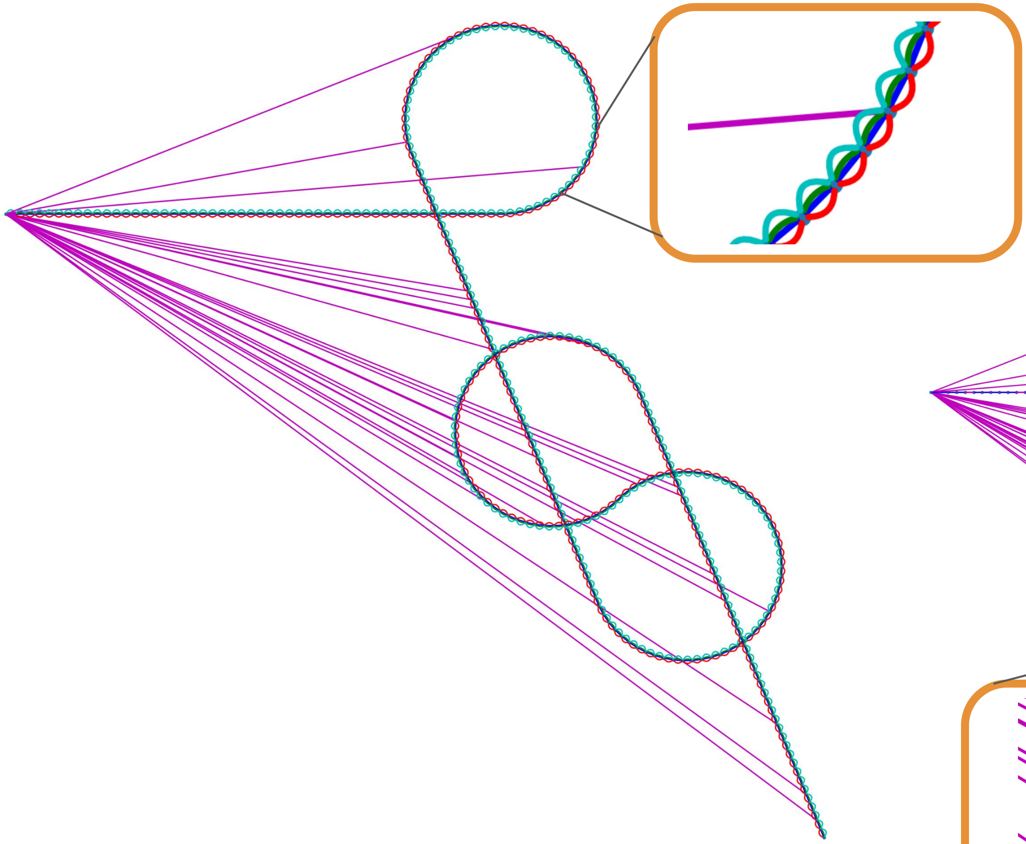


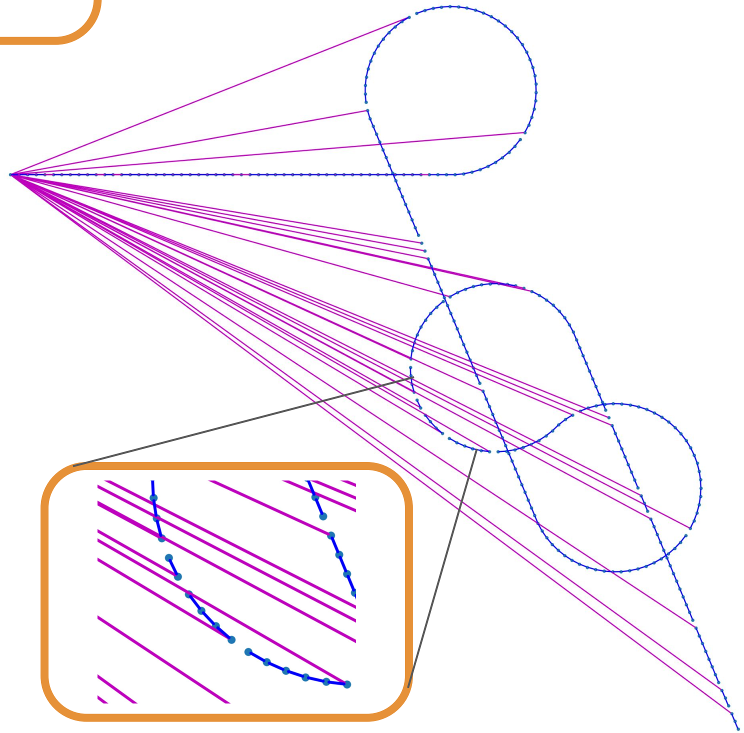
30 global factors



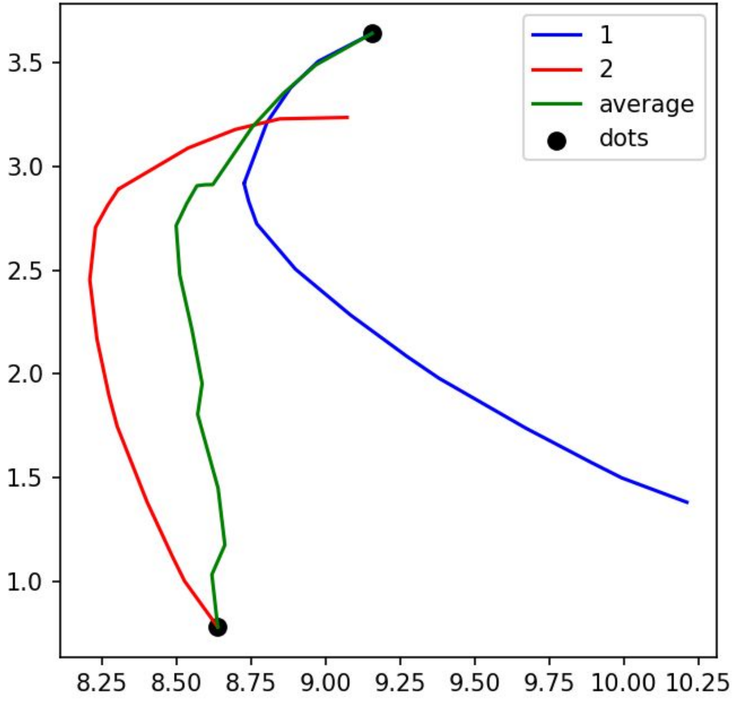
Ostov-based trajectory as init guess

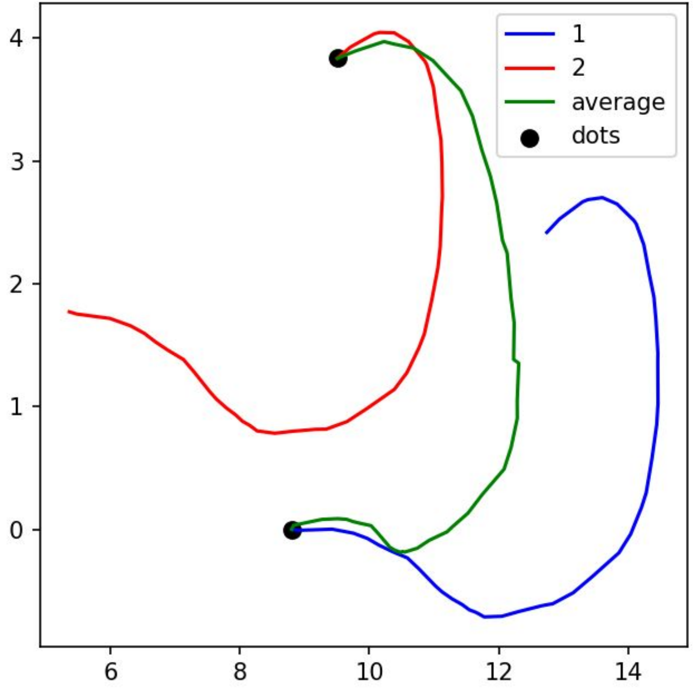
From all-data graph to minimum spanning tree (ostov) with Prim`s algorithm



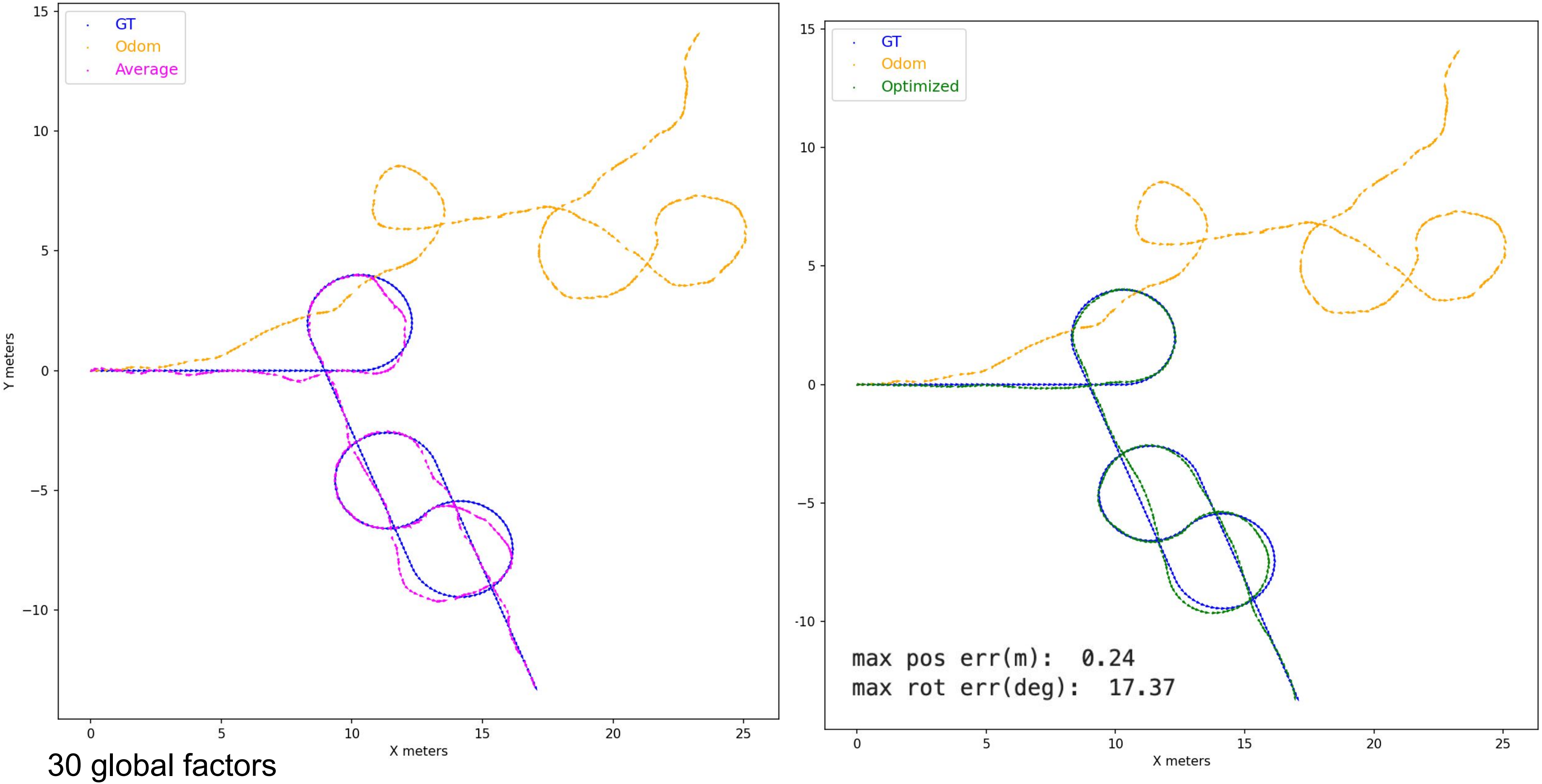


What happens if an initial estimate is Forward-Backward weighted average-based solution?





Forward-Backward weighted average results



30 global factors

From complete graph to minimum spanning tree (ostov) with Prim`s algorithm

while $||\delta||_2 >= \epsilon$:

1) calculate A, b around X, and $A = QR$

2) solve $\delta = R^{-1}Q^T b$ and update $X := X + \delta$

1) calculate Ostov / Forward-Backward average

2) compute initial estimation X_0

3) calculate A, b around X_0, and $A = QR$

4) solve $\delta = R^{-1}Q^T b$ **once** and update $X := X + \delta$

$$\boxed{m}[O_{A,b}(3k^2N) + O_{QR}(k^2N^3) + O_{solve}(k^2N^2)]$$

$$O_{A,b}(3k^2N) + \boxed{O_{ostov}(N + M\log M)} + O_{QR}(k^2N^3) + O_{solve}(k^2N^2)$$

$$O_{A,b}(3k^2N) + \boxed{O_{ostov}(N)} + O_{QR}(k^2N^3) + O_{solve}(k^2N^2)$$

N - number of measurements (edges)

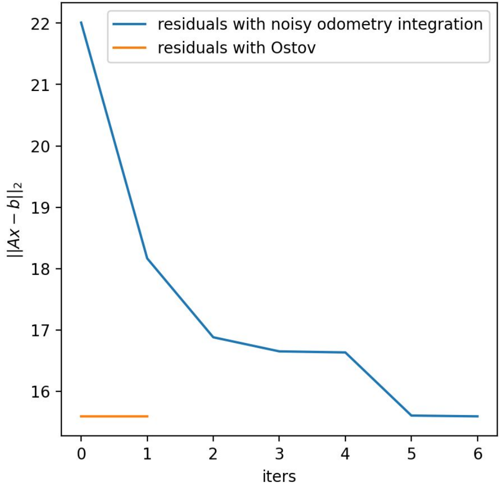
M - number of states (verices)

k - dim of each state (vertex)

Main results

Initial estimation	Max transition error (m)	Max rotation error (deg)	Complexity
Zeros	2.59	172.58	$O(m(N^2 + N^3))$
noisy odometry integration	0.27	24.11	$O(m(N^2 + N^3))$
Ostov-based solution	0.26	17.1	$O(N^2 + N^3)$
Forward-Backward Weighted Average	0.24	17.37	$O(N^2 + N^3)$

With the simplified problem and less amount of operations we have achieved the same accuracy



Literature

1)David P. Woodruff. 2014. Sketching as a Tool for Numerical Linear Algebra. Found. Trends Theor. Comput. Sci. 10, 1–2 (October 2014), 1–157. https://doi.org/10.1561/0400000060

2) R. Kümmerle, G. Grisetti, H. Strasdat, K. Konolige and W. Burgard, "G2o: A general framework for graph optimization," *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 3607-3613, doi: 10.1109/ICRA.2011.5979949.

3) M. Kaess, A. Ranganathan and F. Dellaert, "iSAM: Incremental Smoothing and Mapping," in IEEE Transactions on Robotics, vol. 24, no. 6, pp. 1365-1378, Dec. 2008, doi: 10.1109/TRO.2008.2006706.

3) Different sketching techniques https://github.com/JinChengneng/MatrixSketching/tree/master/doc