ECCV'20
ONLINE
23-28 AUGUST 2020

16TH EUROPEAN CONFERENCE ON

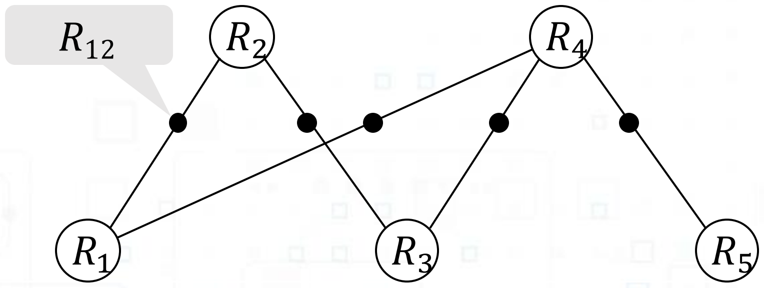
COMPUTER VISION

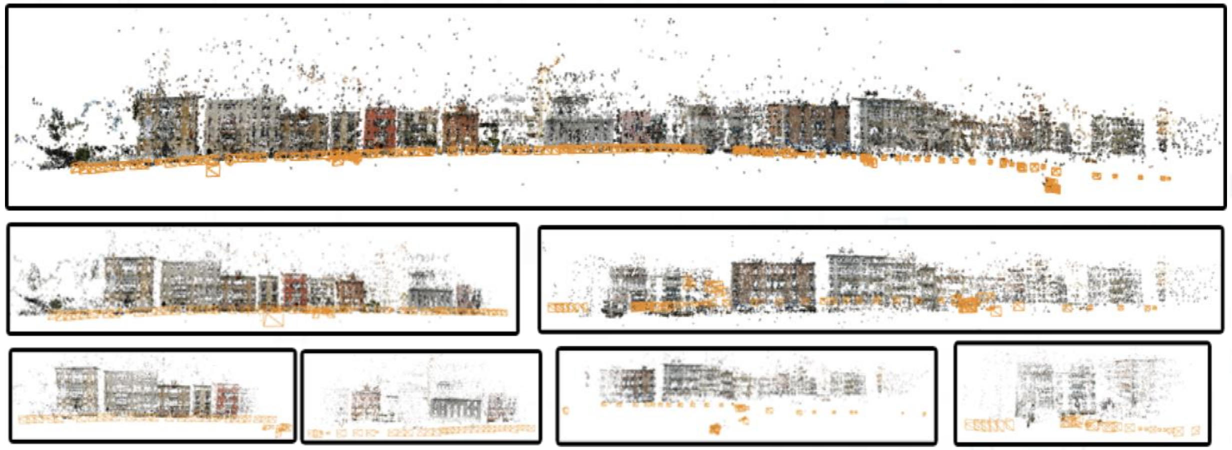
WWW.ECCV2020.EU

Shonan* Rotation Averaging: Global Optimality by Surfing $SO(p)^n$

Frank Dellaert*, David Rosen*, Jing Wu, Robert Mahony, and Luca Carlone

- Rotation Averaging:
  - Given relative rotations
  - find absolute rotations
- Of great practical interest in
  - 3D reconstruction
  - Multi-camera rig calibration
  - Sensor network Localization

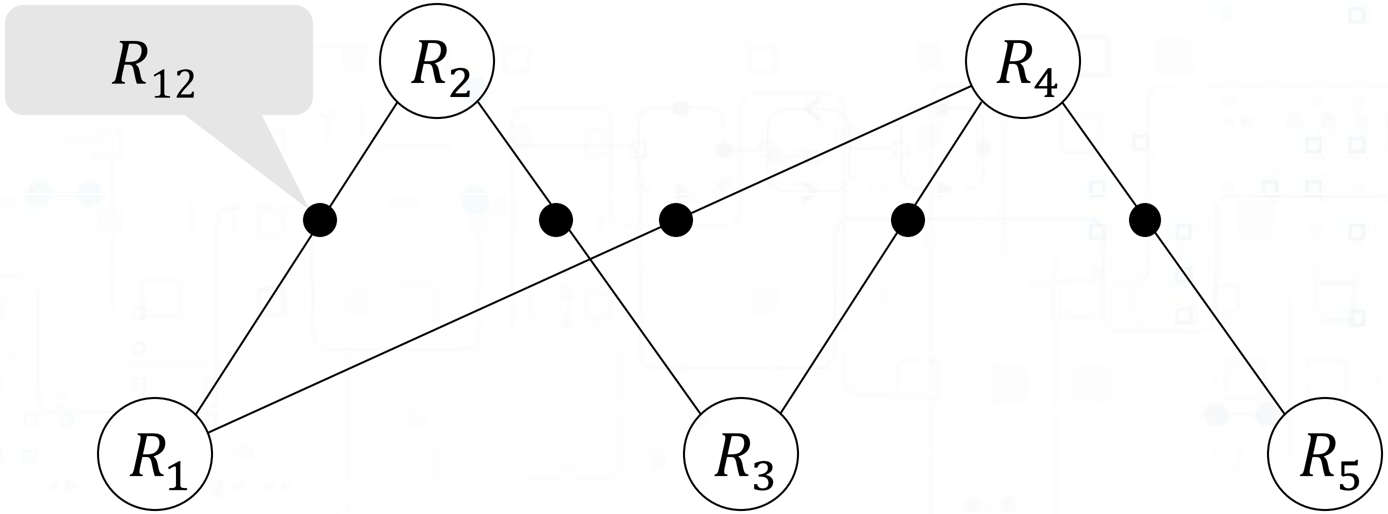




* Equal Contribution

* After Shonan (湘南) peninsula, Japan, where authors conceived of main idea.

Rotation averaging is simply minimizing the sum of Frobenius norms between predicted and observed relative rotations, but this is a high-dimensional and non-convex problem.



$$\min_{\boldsymbol{R} \in \mathrm{SO}(d)^n} \sum_{(i,j) \in \mathcal{E}} \kappa_{ij} \left\| \boldsymbol{R}_j - \boldsymbol{R}_i \bar{\boldsymbol{R}}_{ij} \right\|_{\mathrm{F}}^2$$

ECCV'20 ONLINE 23-28 AUGUST 2020

Shonan Rotation averaging is a very simple algorithm, that terminates at the global optimum, guaranteed.

- Try non-linear optimizer over $R \in SO(3)^n$
- If you fail, for p=4,5,... :
  - Try again optimizing over $Q \in SO(p)^n$
- Repeat until you terminate

Probably terminates at a low p, e.g. 4 or 5!

Many practical problems terminate at SO(3), so no performance penalty there.

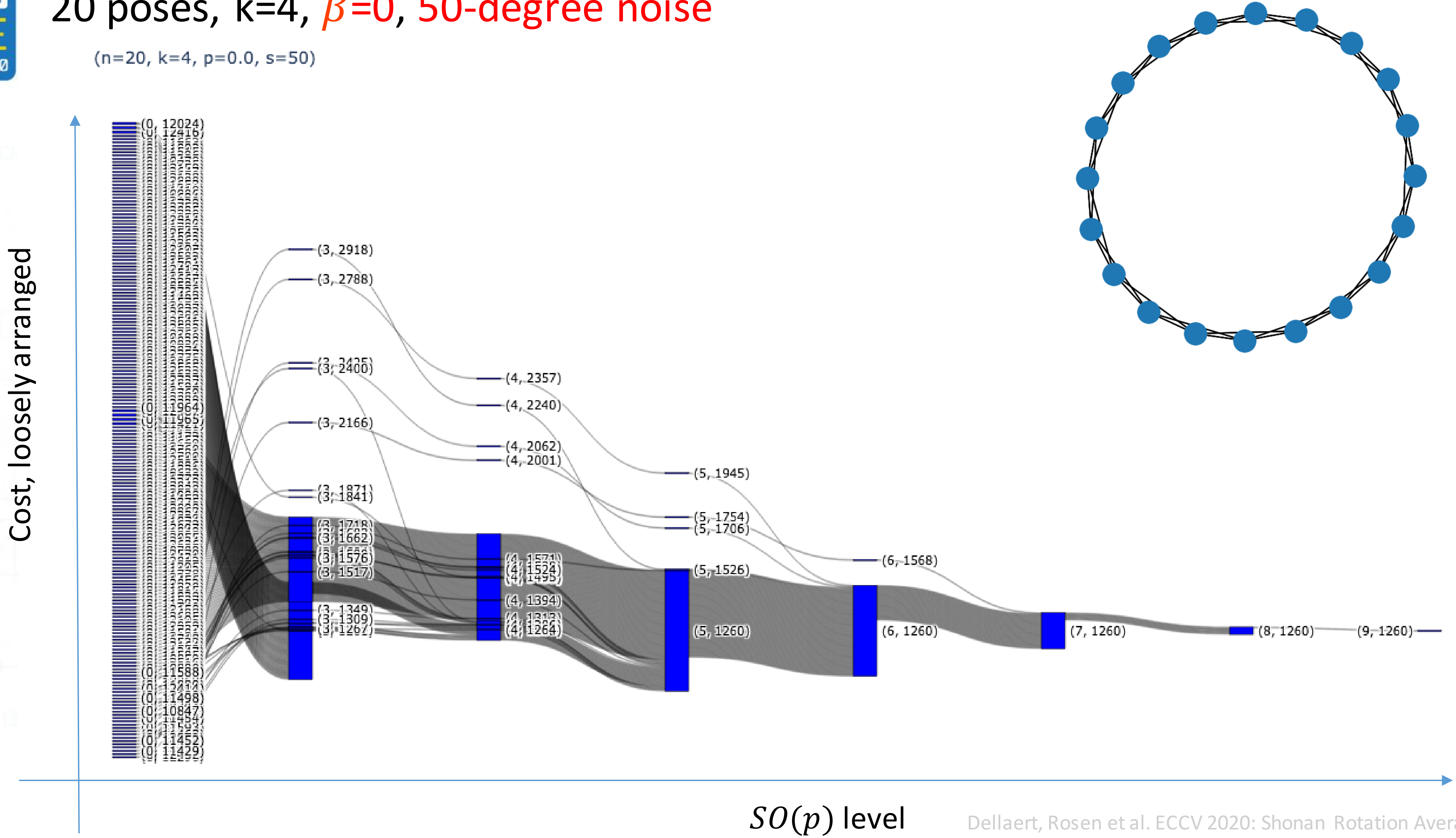
There is a (loose) assumption on the noise magnitude*, and you need to do a trivial projection to make sure the Frobenius norm is over 3×3 matrices, but that's it.

Need to project Q back to a $p \times 3$ error matrix using a p $\times$ 3 projection matrix $P = [I_3; 0]$:

$$\min_{\boldsymbol{Q} \in \mathrm{SO}(p)^n} \sum_{(i,j) \in \mathcal{E}} \kappa_{ij} \left\| \mathrm{vec}(\boldsymbol{Q}_j \boldsymbol{P} - \boldsymbol{Q}_i \boldsymbol{P} \bar{\boldsymbol{R}}_{ij}) \right\|_2^2$$

* A. Eriksson, C. Olsson, F. Kahl, and T.-J. Chin. Rotation averaging and strong duality, CVPR 2018

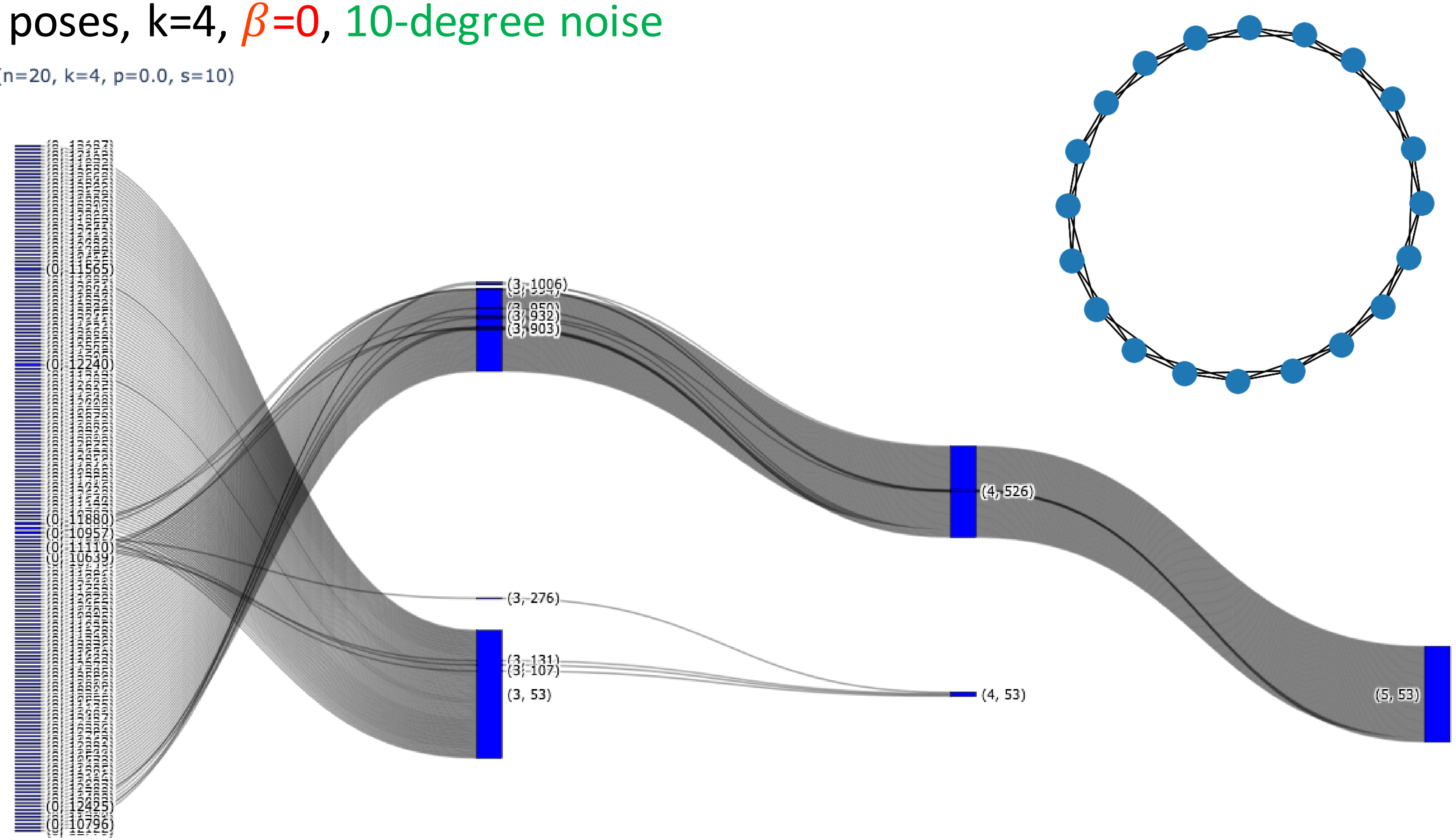
Using “small-world” graphs, used in CVPR by Wilson & Bindel, we can examine the behavior of Shonan averaging. Here we sample a *single instance* with 20 poses, k=4, β=0, 50-degree noise



The fate of 200 random initial estimates shows that when the added noise is 10 degrees instead of 50 degrees, termination happens at earlier levels.

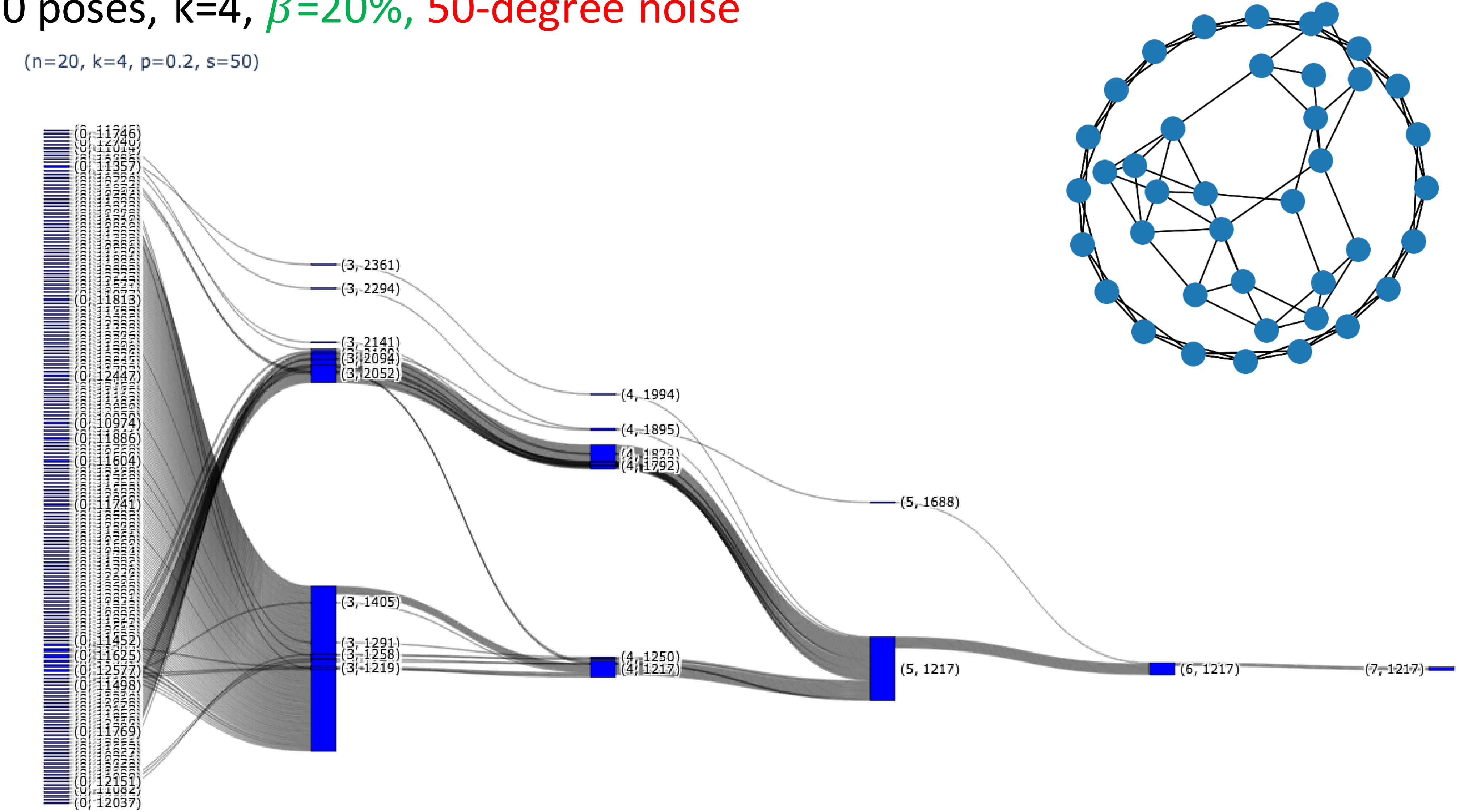
20 poses, k=4, β=0, 10-degree noise

(n=20, k=4, p=0.0, s=10)



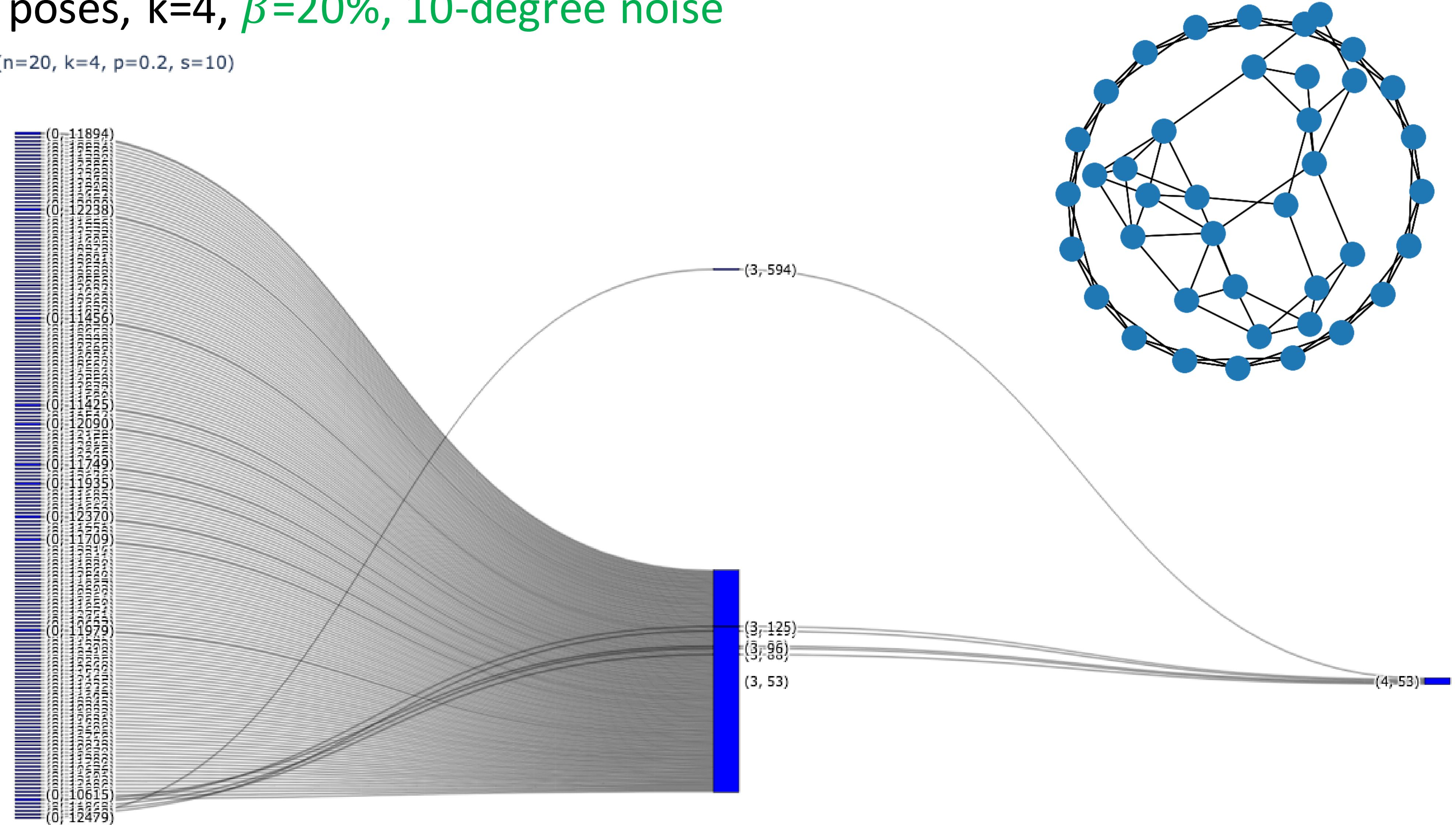
Early termination also happens if we rewire a fraction of the edges randomly (20% here), which increases the algebraic connectivity of the graph.

20 poses, k=4, β=20%, 50-degree noise



Finally, reducing the noise while simultaneously having more random edges is the best case, and we terminate at the $SO(3)$ level the majority of the time.
20 poses, k=4, β=20%, 10-degree noise

(n=20, k=4, p=0.2, s=10)



dataset	method	error	min	avg	max	success
statue_of_liberty_1	SA	0.000%	0.038	0.313	1.211	100%
(n=19, m=54)	SL	0.000%	0.010	0.219	0.901	100%
	S3	nan%	nan	nan	nan	0%
	S4	0.001%	0.009	0.016	0.019	30%
	S5	0.001%	0.010	0.018	0.022	40%
	SK	-0.000%	0.008	**0.108**	0.459	100%
natural_history_museum_london	SA	0.000%	0.019	0.036	0.049	100%
(n=30, m=274)	SL	0.000%	0.011	**0.021**	0.068	100%
	S3	0.000%	0.010	0.013	0.015	60%
	S4	0.000%	0.011	0.016	0.021	100%
	S5	0.000%	0.009	0.014	0.018	100%
	SK	-0.000%	0.021	**0.022**	0.024	100%
statue_of_liberty_2	SA	0.000%	0.030	0.063	0.094	100%
(n=39, m=156)	SL	0.001%	0.011	**0.034**	0.060	100%
	S3	0.000%	0.010	0.028	0.046	40%
	S4	0.000%	0.011	0.024	0.047	100%
	S5	0.000%	0.010	0.030	0.057	100%
	SK	0.000%	0.019	0.050	0.113	100%
taj_mahal_entrance	SA	0.000%	0.071	0.117	0.165	100%
(n=42, m=1272)	SL	0.000%	0.032	**0.043**	0.062	100%
	S3	0.000%	0.037	0.046	0.063	80%
	S4	0.000%	0.033	0.042	0.051	100%
	S5	0.000%	0.032	0.037	0.039	100%
	SK	-0.000%	0.070	0.081	0.092	100%
sistine_chapel_ceiling_1	SA	0.000%	0.102	0.173	0.246	100%
(n=49, m=1754)	SL	0.000%	0.064	**0.085**	0.108	100%
	S3	0.000%	0.057	0.073	0.087	60%
	S4	0.000%	0.072	0.121	0.293	100%
	S5	0.000%	0.064	0.083	0.095	100%
	SK	-0.000%	0.102	0.116	0.130	100%

SA=Shonan from p=5
SL =Shonan from p=3
S3,S4,S5 = single-level
SK = with inner constraints

Many real-world datasets have high algebraic connectivity (provided by the cross-cutting edges) and hence terminate quickly. On the left we show YFCC instance for a relatively small number of cameras.

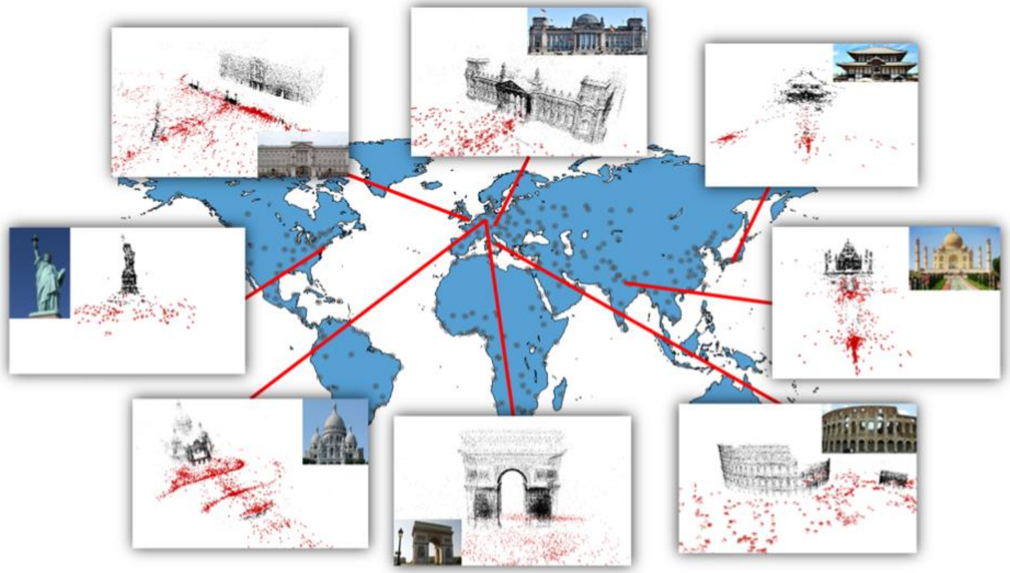


Image from Heinly et al, CVPR 2015

dataset	method	error	min	avg	max	success
st_peters_basilica_interior_2	SA	0.000%	0.948	1.165	1.738	100%
(n=173, m=11688)	SL	0.000%	0.557	**0.713**	0.897	100%
	S3	0.000%	0.531	0.587	0.715	80%
pantheon_interior	SA	0.000%	0.852	1.001	1.570	100%
(n=186, m=10000)	SL	0.000%	0.449	**0.509**	0.623	100%
	S3	0.000%	0.455	0.530	0.642	100%
florence_cathedral_dome_interior_1	SA	0.000%	2.435	2.848	3.226	100%
(n=213, m=31040)	SL	0.000%	1.869	**2.170**	2.499	100%
	S3	0.000%	1.856	2.209	2.814	100%
paris_opera_1	SA	0.000%	3.673	4.145	4.467	100%
(n=254, m=45754)	SL	0.000%	1.686	**2.024**	2.677	100%
	S3	0.000%	1.683	2.104	2.510	100%
pike_place_market	SA	0.000%	4.437	5.255	7.285	100%
(n=265, m=53242)	SL	0.000%	2.019	**2.322**	3.400	100%
	S3	0.000%	2.122	2.246	2.384	100%
blue_mosque_interior_1	SA	0.000%	3.581	4.153	4.964	100%
(n=272, m=40292)	SL	0.000%	2.268	**2.708**	2.991	100%
	S3	0.000%	2.511	2.997	3.645	100%
notre_dame_rosary_window	SA	0.000%	8.102	8.652	10.515	100%
(n=326, m=93104)	SL	0.000%	3.478	**4.074**	4.690	100%
	S3	0.000%	3.885	4.326	5.012	100%
british_museum	SA	0.000%	4.297	4.851	7.846	100%
(n=344, m=45450)	SL	0.000%	1.927	**2.558**	3.400	100%
	S3	0.000%	2.009	2.558	3.911	100%
palace_of_westminster	SA	0.000%	1.334	1.480	1.692	100%
(n=345, m=11522)	SL	0.000%	0.598	**0.837**	1.242	100%
	S3	0.000%	0.731	1.038	2.397	100%
louvre	SA	0.000%	2.872	3.141	3.709	100%
(n=367, m=26656)	SL	0.000%	1.606	**2.284**	3.265	100%
	S3	0.000%	1.648	2.058	2.495	100%
st_peters_basilica_interior_1	SA	0.000%	5.283	5.654	6.295	100%
(n=365, m=55024)	SL	0.000%	3.255	**4.030**	4.995	100%
	S3	0.000%	3.662	4.169	4.728	100%

SA=Shonan from p=5; SL =Shonan from p=3; S3 = single-level

Larger, well-connect datasets in YFCC almost always converge to the global optimum FAST. Minimal performance penalty is checking duality convergence proof, which is an eigenvalue problem.

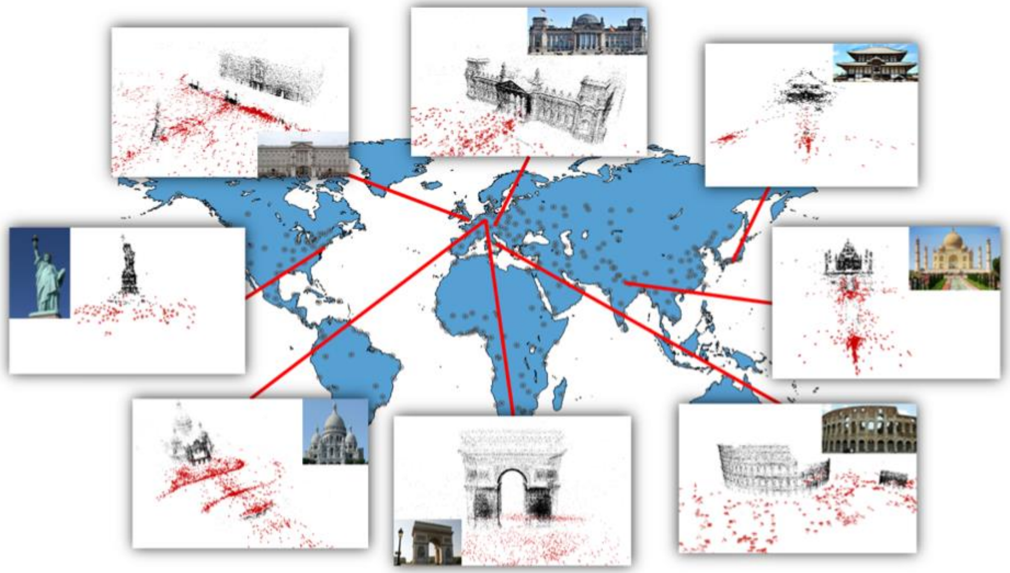
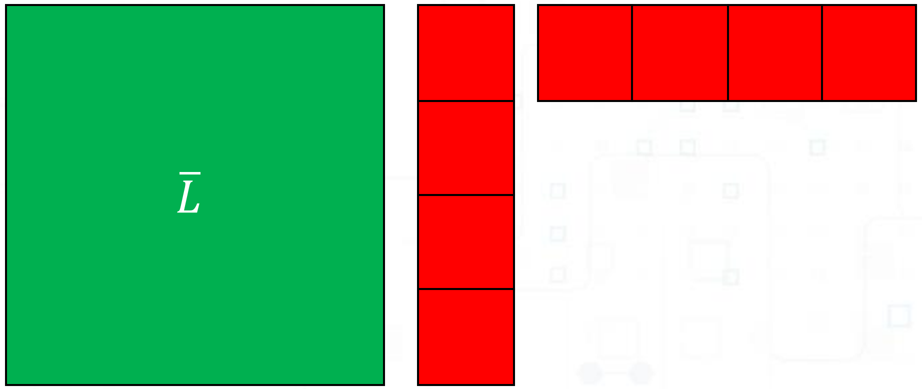


Image from Heinly et al, CVPR 2015

Why does it work? The proof is based on SE-Sync, a method for pose graph optimization by Rosen et al. To sketch the idea, we re-write objective using outer product:

- $R = 3 \times 3n$ $\begin{bmatrix} R_1 & R_2 & R_3 & R_4 \end{bmatrix}$
- $\bar{L} = 3n \times 3n$
- $R^T R = 3n \times 3n$
- Minimize $tr(\bar{L} R^T R) \rightarrow$ $tr($ 

$)$

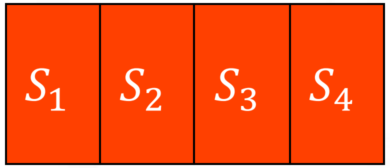
- Non-convex!
- But: terminates here in many many cases

While local optimization over the $SO(3)^n$ manifold is non-convex, there is beautiful yet expensive convex relaxation, in the form a semi-definite program (SDP).

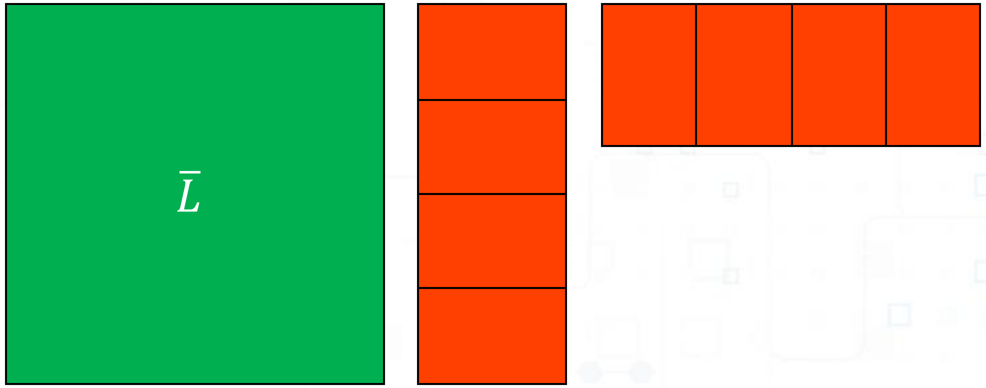
- $R = 3 \times 3n$ $\boxed{R_1 \mid R_2 \mid R_3 \mid R_4}$
- $\bar{L} = 3n \times 3n$
- $R^T R = 3n \times 3n$
- Minimize $tr(\bar{L} R^T R) \rightarrow tr(\boxed{\bar{L}}\ \boxed{Z})$
- Semidefinite program (SDP) = convex !!!
- Rosen '16 proved: *if noise is not too large*, Z^* *can* be factored as $R^T R$
- Hence, this is a convex relaxation, but expensive!

To find a global optimizer, Burer & Monteiro suggested to just try a low-rank approximation $S^T S$. Here S is a *Stiefel manifold*.

- $S = 4 \times 3n$



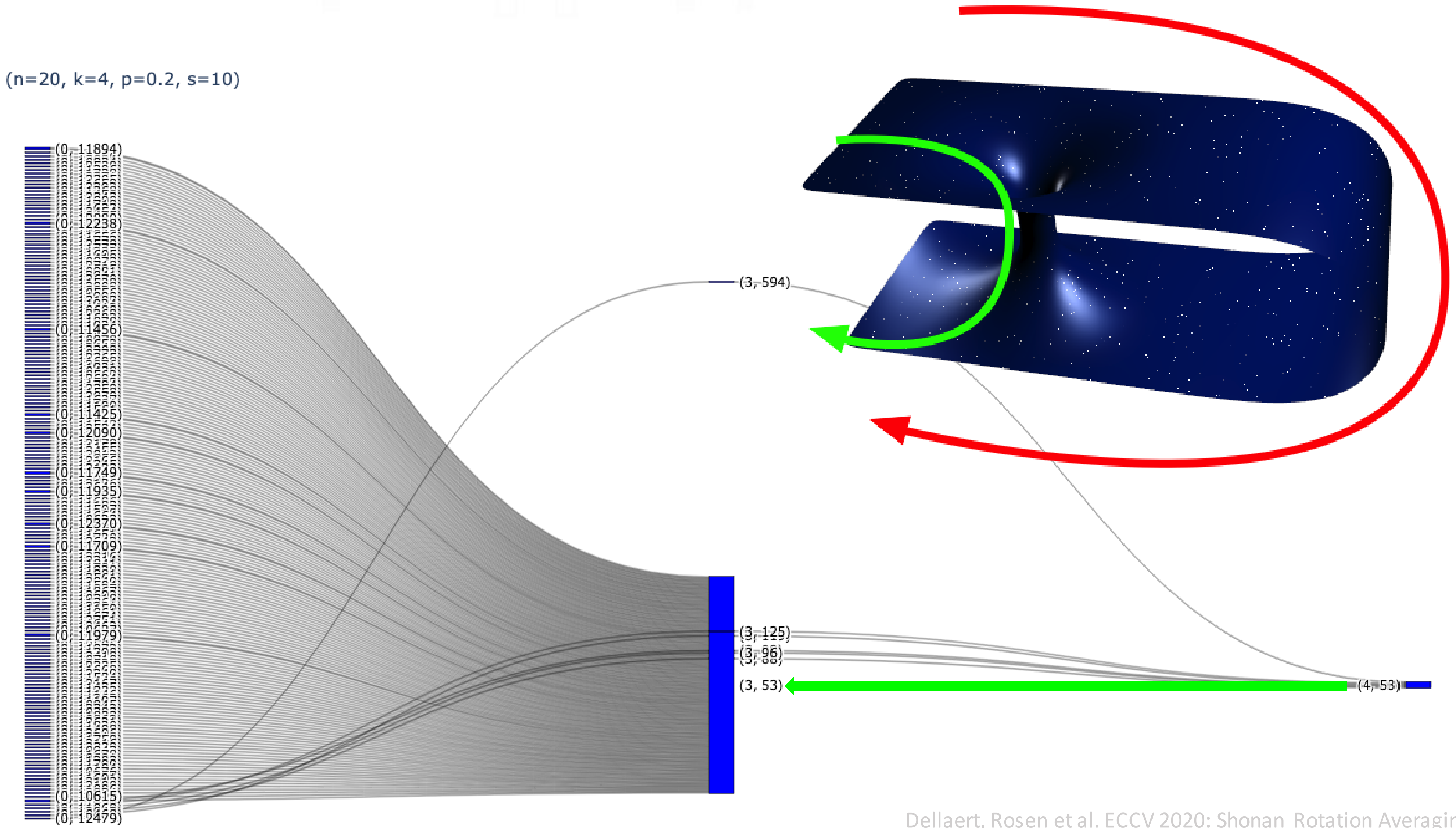
- $\bar{L} = 3n \times 3n$
- $S^T S = 3n \times 3n$
- Minimize $tr(\bar{L} S^T S) \rightarrow$ $tr($



$)$

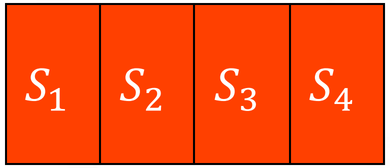
- Burer-Monteiro ’03: low-rank approximation
- In general, non-convex, rank 4 PSD
- Use fast local search, with a bit of extra room

Does not always work, but when it does, Stiefel manifold acts as a “wormhole”, from which we project back onto the original the $SO(3)^n$ manifold.
Here illustrated again with 20 poses, k=4, β=20%, 10-degree noise

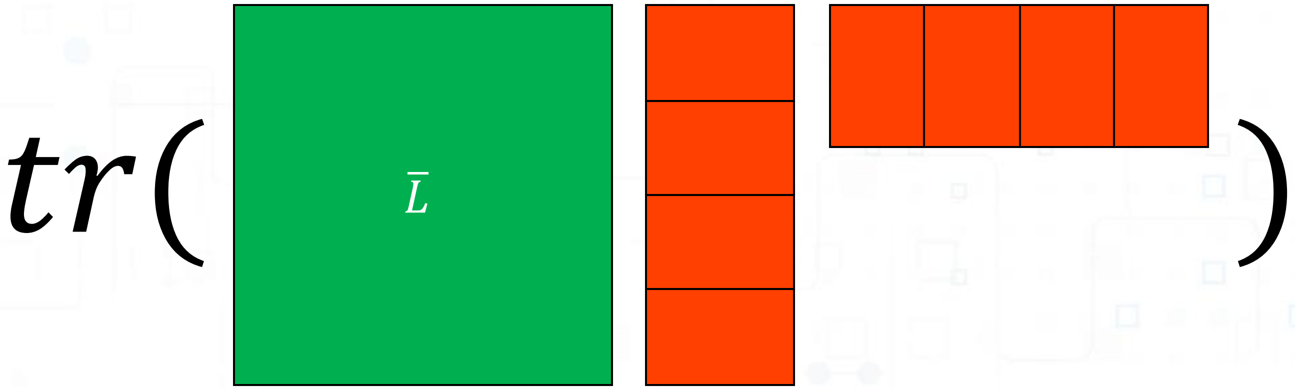


With some work we can optimize over this Stiefel manifold, but a cool fact is that the 4×3 Stiefel manifold is *isomorphic* to $SO(4)$

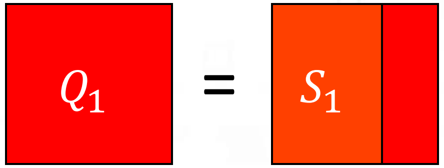
- $S = 4 \times 3n$



- $\bar{L} = 3n \times 3n$
- $S^T S = 3n \times 3n$
- Minimize $tr(\bar{L} S^T S) \rightarrow$



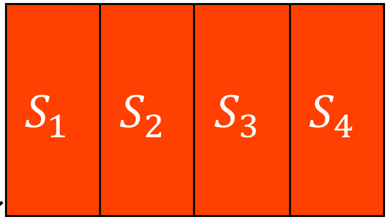
- Easy to see, last column must be orthogonal:



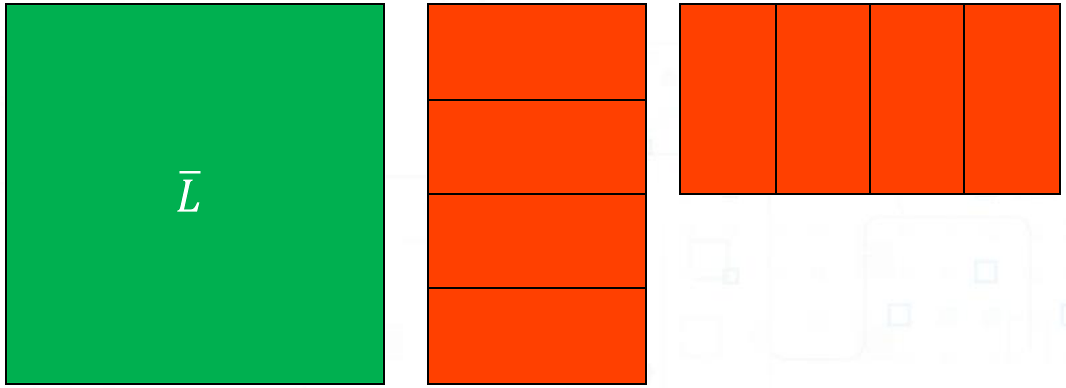
- ***So we can just re-use our usual machinery!***

What if it does not work at $SO(4)$?? Burer & Monteiro suggested just re-starting from a different seed, but Boumal invented the *Riemannian Staircase*: just switch to $SO(5)$, etc…

- $S = 5 \times 3n$

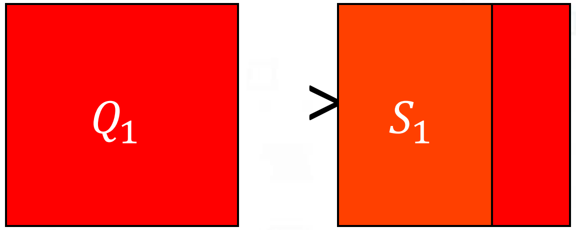


- $\bar{L} = 3n \times 3n$
- $S^T S = 3n \times 3n$
- Minimize $tr(\bar{L}S^T S) \rightarrow$ **$tr($**

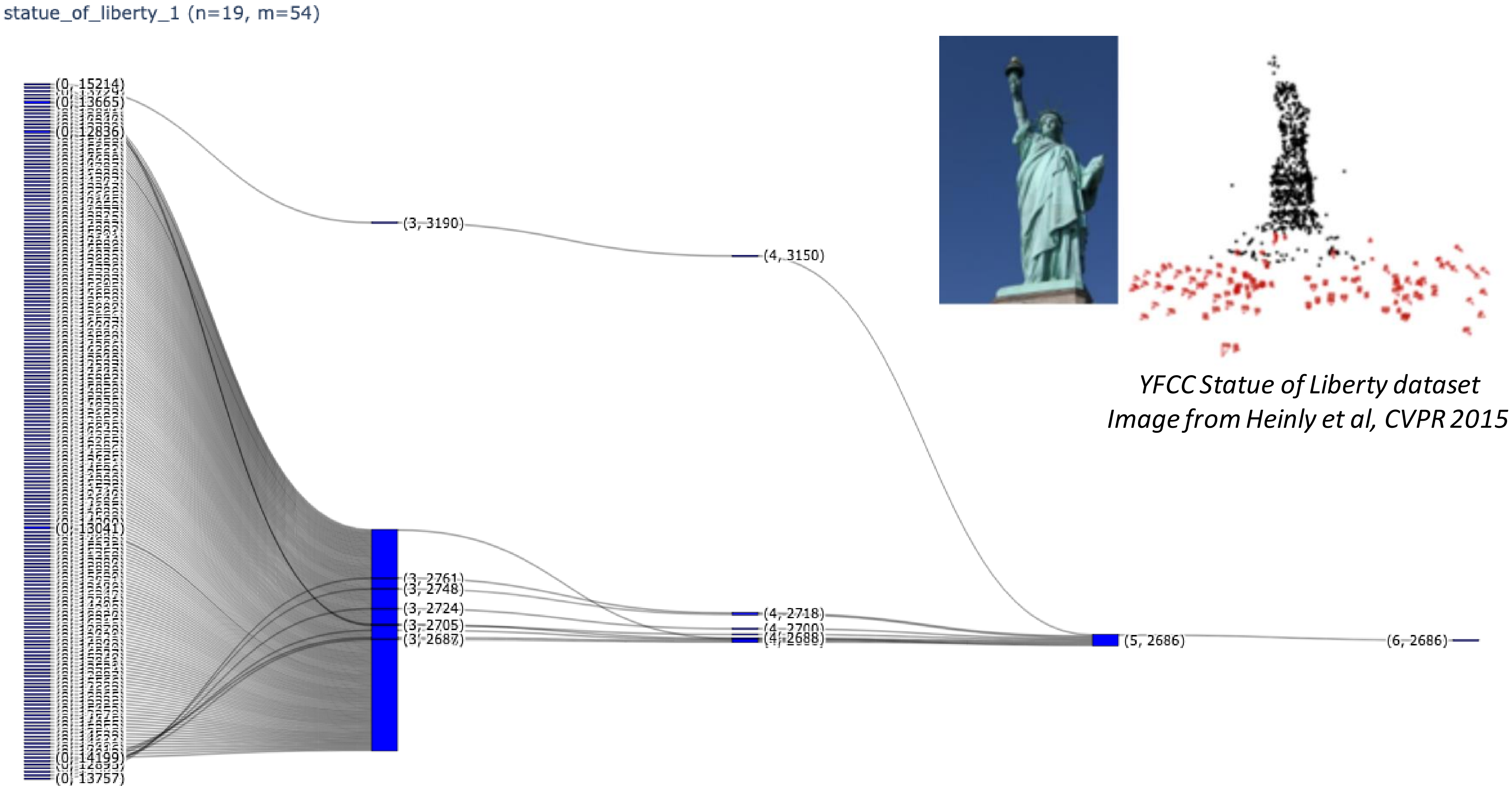


$)$

- Continue to use fast local search, on $SO(5)$!!
- $SO(5)$ is *not* isomorphic to 5×3 Stiefel:
- However, we just ignore the implied gauge



Empirically, the Riemannian staircase converges for a low value of p. p-distribution depends on algebraic connectivity (see Wilson& Bindel CVPR)



YFCC Statue of Liberty dataset
Image from Heinly et al, CVPR 2015

In summary:

Shonan Rotation averaging is a very simple algorithm, that terminates at the global optimum, guaranteed.

- Try non-linear optimizer over $R \in SO(3)^n$
- If you fail, for p=4,5,… :
    - Try again optimizing over $Q \in SO(p)^n$
- Repeat until you terminate

Probably terminates at a low p, e.g. 4 or 5!

Many practical problems terminate at SO(3), and only extra cost is verifying global optimality.