Message Passing Least Squares Framework and Its Applications to Rotation Synchronization

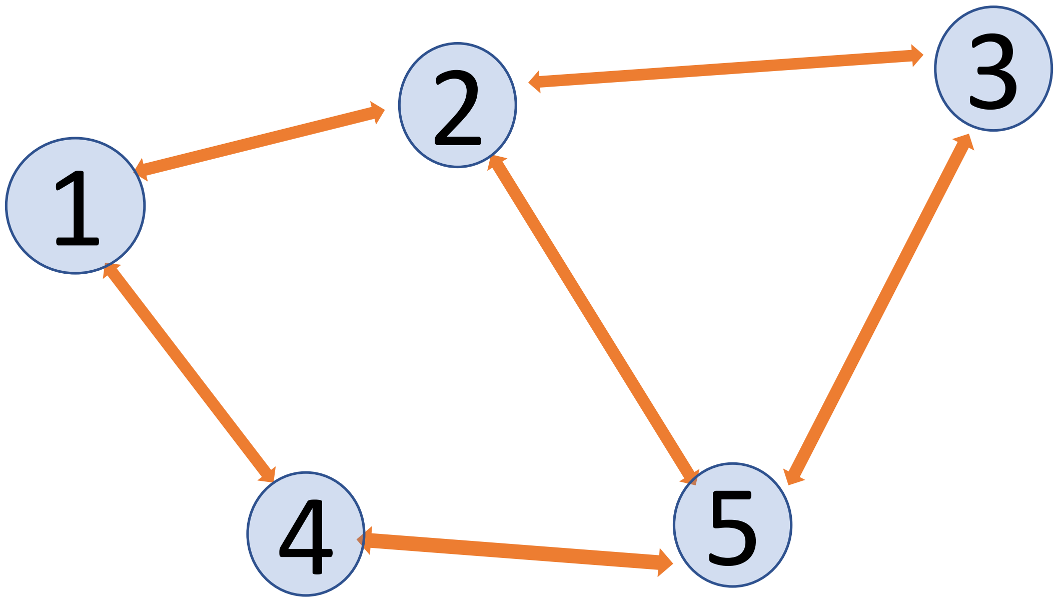
Yunpeng Shi

Joint work with Prof. Gilad Lerman

School of Mathematics

University of Minnesota

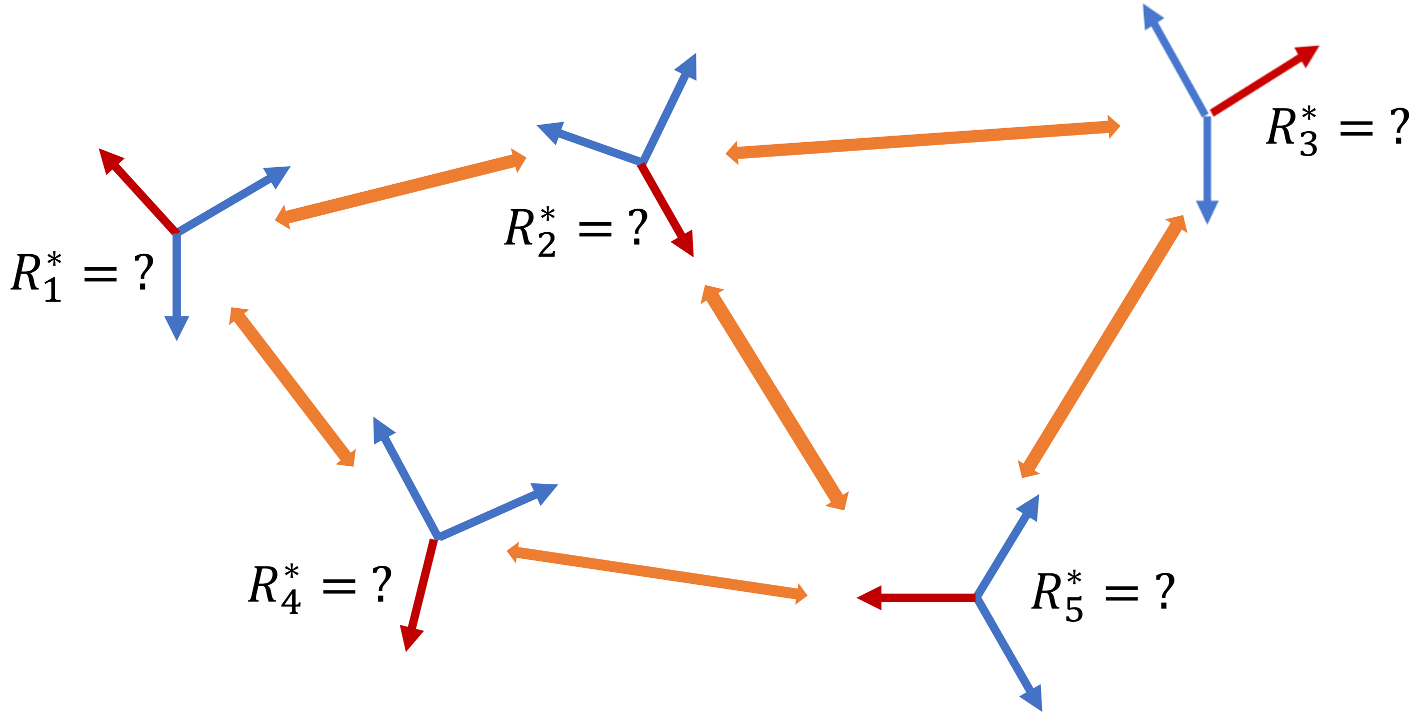
Rotation Synchronization



Given a graph $G([n], E)$

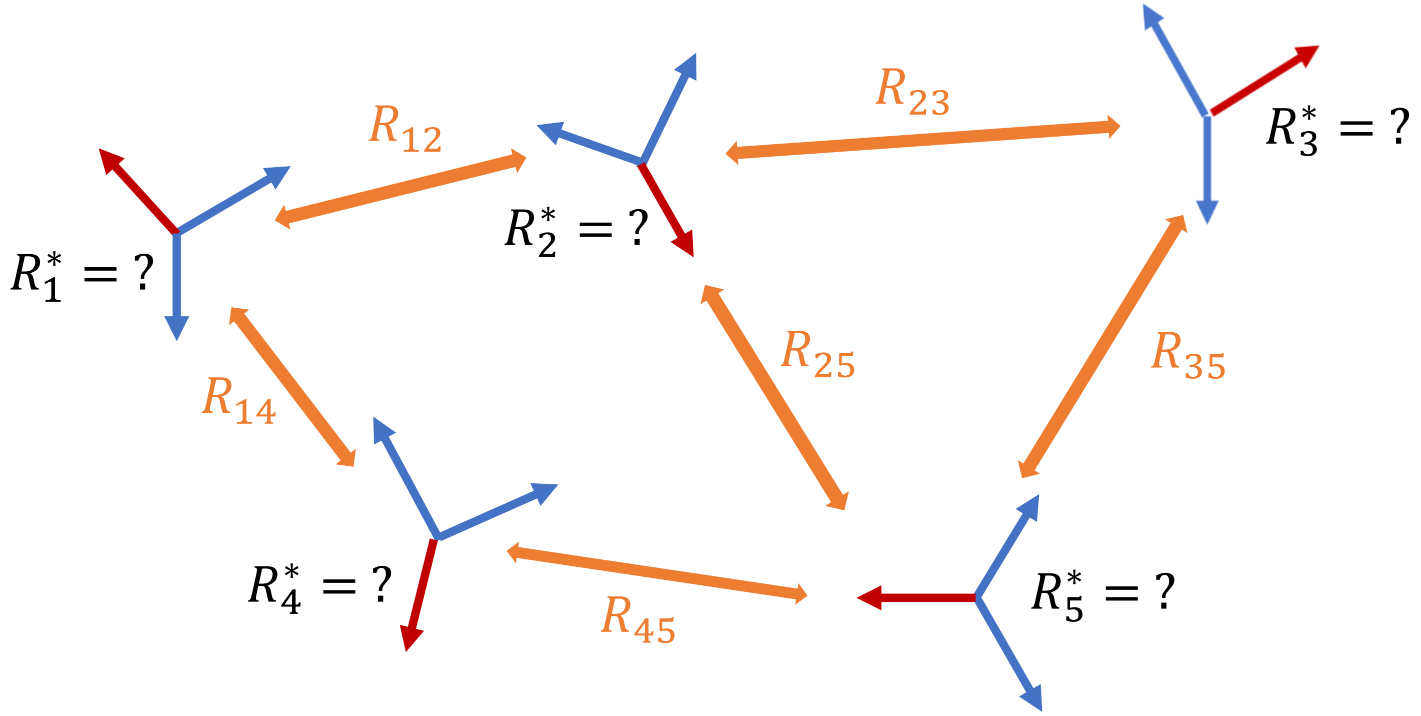
$[n] := \{1,2,3, \ldots, n\}$, E is the set of edges

Rotation Synchronization



Each node $i \in [n]$ is assigned an *unknown* ground truth rotation $R_i^* \in SO(3)$

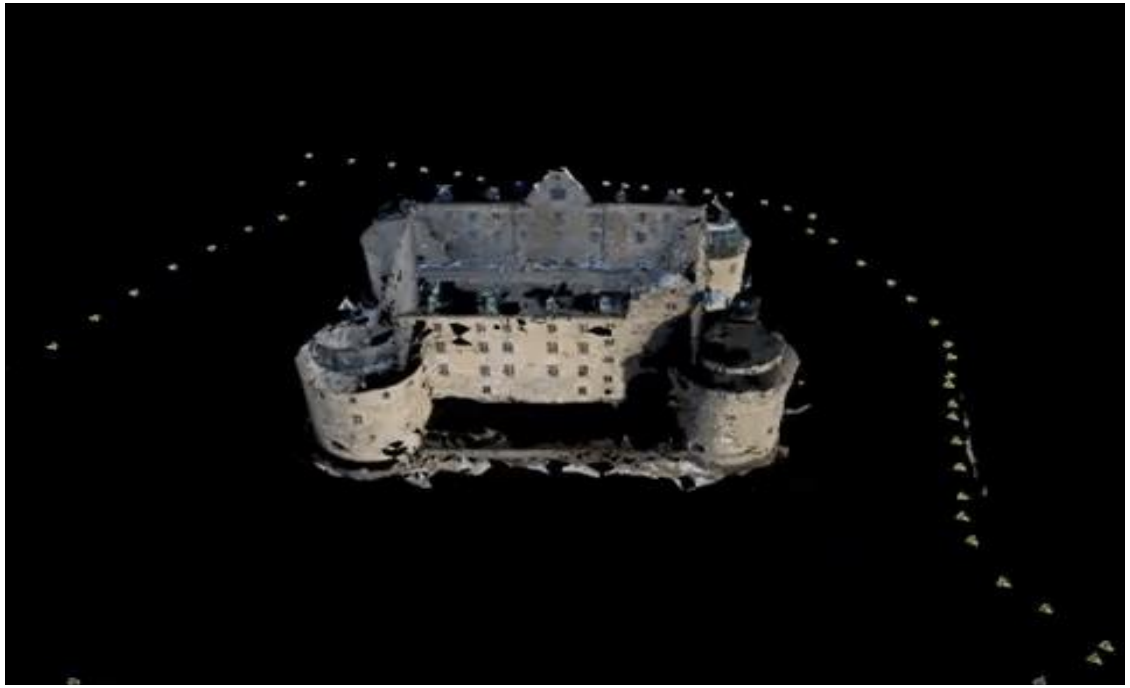
Rotation Synchronization



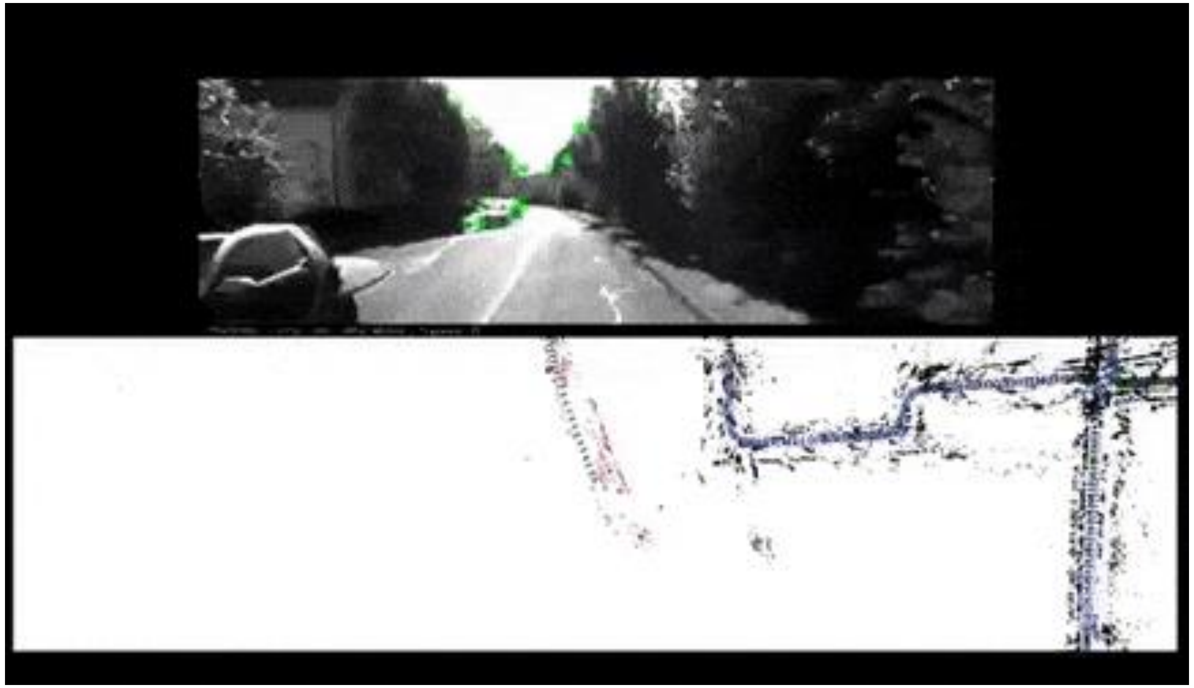
- Each edge $ij \in E$ is given a possibly *noisy* and *corrupted* relative rotation R_{ij}
- The *uncorrupted* relative rotation for $ij \in E$ is $R_{ij}^* = R_i^* R_j^{*-1}$
- **Rotation Synchronization**: Estimate $\{R_i^*\}_{i\in[n]}$ from $\{R_{ij}\}_{ij\in E}$
- $\{R_i^* R\}_{i\in[n]}$ for any rotation R is also a solution

Applications

Camera orientation estimation in 3D reconstruction tasks:



Structure from motion (SfM)
Demonstration by Carl Olsson



Simultaneous localization and mapping (SLAM)
Demonstration by Raúl Mur-Artal

Adversarial Corruption Model

$$R_{ij} = \begin{cases} R_{ij}^* := R_i^* R_j^{*-1}, & ij \in E_g \quad \text{(good edges)} \\ \tilde{R}_{ij}, & ij \in E_b \quad \text{(bad edges)} \end{cases}$$

Adversarial Corruption Model

$$R_{ij} = \begin{cases} R_{ij}^* := R_i^* R_j^{*-1}, & ij \in E_g \quad \text{(good edges)} \\ \tilde{R}_{ij}, & ij \in E_b \quad \text{(bad edges)} \end{cases}$$

Corruption Level $\quad s_{ij}^* \coloneqq d(R_{ij}, R_{ij}^*)$

Commonly, d is the geodesic distance on $SO(3)$

Least Squares Solvers

$$\underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\text{minimize}} \sum_{ij\in E} d^2(R_i R_j^{-1}, R_{ij})$$

The most common approximate solution is the Lie algebraic averaging

Robust Solvers: l_p minimization $(0 < p \leq 1)$

$$\underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\text{minimize}} \sum_{ij\in E} d^p(R_i R_j^{-1}, R_{ij})$$

How to minimize $\sum_{ij\in E} d^p(R_i R_j^{-1}, R_{ij})$ over $\{R_i\}_{i\in[n]}$ in SO(3)?

How to minimize $\sum_{ij\in E} d^p(R_i R_j^{-1}, R_{ij})$ over $\{R_i\}_{i\in[n]}$ in SO(3)?

Iteratively Reweighted Least Squares (IRLS):

$$\{R_{i,t}\}_{i\in[n]} = \underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t} d^2(R_i R_j^{-1}, R_{ij})$$

$$r_{ij,t} = d(R_{i,t} R_{j,t}^{-1}, R_{ij})$$

$$w_{ij,t+1} = r_{ij,t}^{p-2}$$

How to minimize $\sum_{ij\in E} d^p(R_i R_j^{-1}, R_{ij})$ over $\{R_i\}_{i\in[n]}$ in SO(3)?

Iteratively Reweighted Least Squares (IRLS):

$$\{R_{i,t}\}_{i\in[n]} = \underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t} d^2(R_i R_j^{-1}, R_{ij})$$

$$r_{ij,t} = d(R_{i,t} R_{j,t}^{-1}, R_{ij})$$

$$w_{ij,t+1} = r_{ij,t}^{p-2}$$

Ideally, $r_{ij,t} \approx s_{ij}^* := d(R_{ij}, R_{ij}^*)$ and $w_{ij,t+1} \approx \left(\frac{1}{s_{ij}^*}\right)^{2-p}$ concentrates on E_g

Issue 1: Over-Aggressive Reweighting

- Under severe corruption, $R_{i,t} \not\approx R_i^*$ and thus $r_{ij,t} \not\approx s_{ij}^*$
- In certain cases, $r_{ij,t} \approx 0$ for $ij \in E_b$, and thus

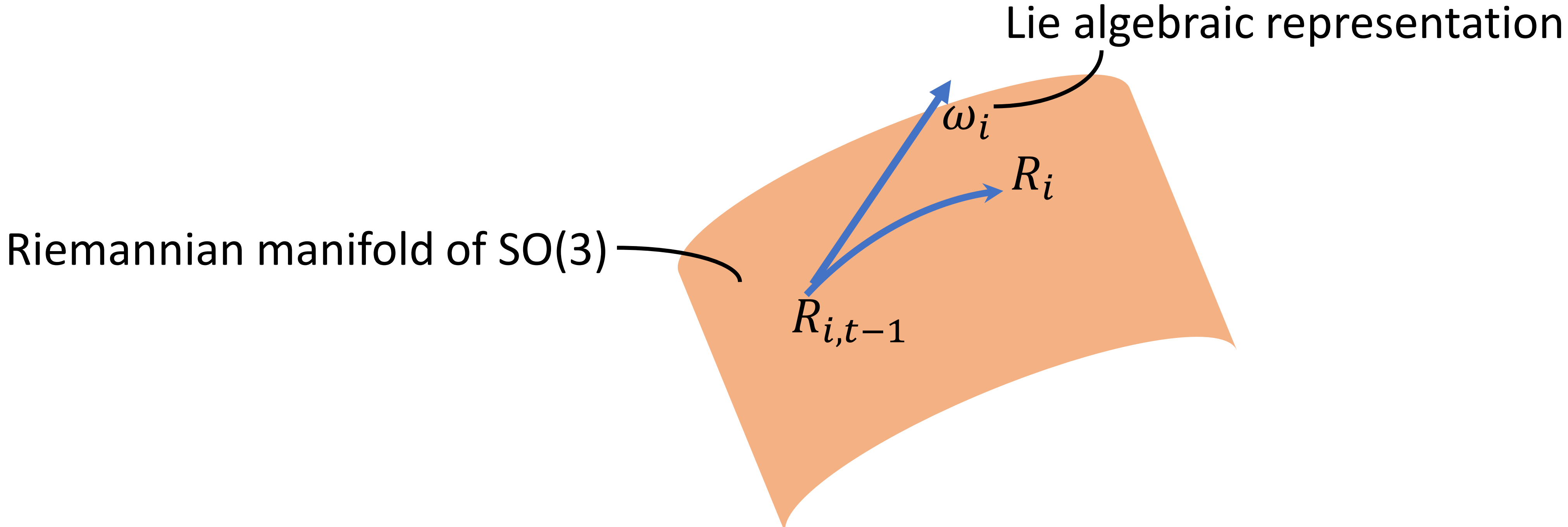
$w_{ij,t+1} = \left(\frac{1}{r_{ij,t}}\right)^{2-p}$ can be extremely high for $ij \in E_b$

Issue 2: Poor Least Squares Solution

$$\{R_{i,t}\}_{i\in[n]} = \underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t} d^2(R_i R_j^{-1}, R_{ij})$$

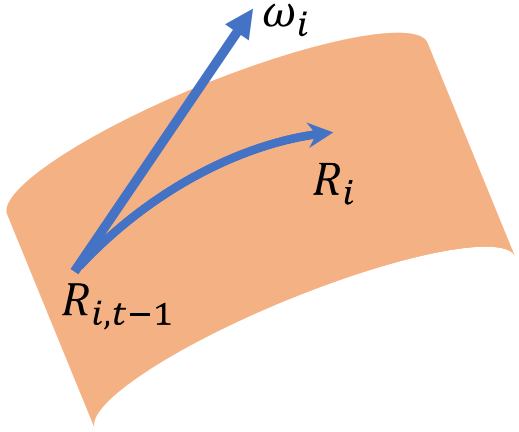
Issue 2: Poor Least Squares Solution

$$\{R_{i,t}\}_{i\in[n]} = \underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t} d^2(R_i R_j^{-1}, R_{ij})$$



Issue 2: Poor Least Squares Solution

$$\{R_{i,t}\}_{i\in[n]} = \underset{\{R_i\}_{i\in[n]}\subset SO(3)}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t} d^2(R_i R_j^{-1}, R_{ij})$$



The Lie Algebraic Averaging uses the approximation

$$d(R_i R_j^{-1}, R_{ij}) \approx \|\omega_i - \omega_j - \omega_{ij}\|_2$$

where $\omega_i = \log(R_{i,t-1}^{-1} R_i)$

and $\omega_{ij} = \log(R_{i,t-1}^{-1} R_{ij} R_{j,t-1})$

The approximation is valid only when $R_i \approx R_i^*$ and $R_{ij} \approx R_i^* R_j^{*-1}$

How to accurately estimate s_{ij}^* without knowing R_i^* and R_j^* ?

Cycle-Edge Message Passing (CEMP)

- **Goal:** Estimate corruption level

$$s_{ij}^{*} \coloneqq d(R_{ij}, R_{ij}^{*})$$

from 3-cycle inconsistency measure

$$d_{ijk} \coloneqq d(R_{ij}R_{jk}R_{ki}, I)$$

Cycle-Edge Message Passing (CEMP)

- **Goal:** Estimate corruption level

$$s_{ij}^* := d(R_{ij}, R_{ij}^*)$$

from 3-cycle inconsistency measure

$$d_{ijk} := d(R_{ij}R_{jk}R_{ki}, I)$$

- For each $ij \in E$, sample 50 3-cycles ijk and for each cycle compute d_{ijk}

Cycle-Edge Message Passing (CEMP)

- **Goal:** Estimate corruption level

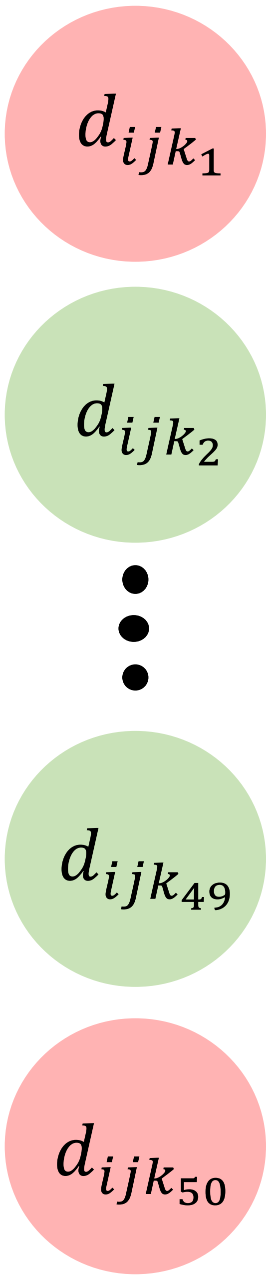
$$s_{ij}^* := d(R_{ij}, R_{ij}^*)$$

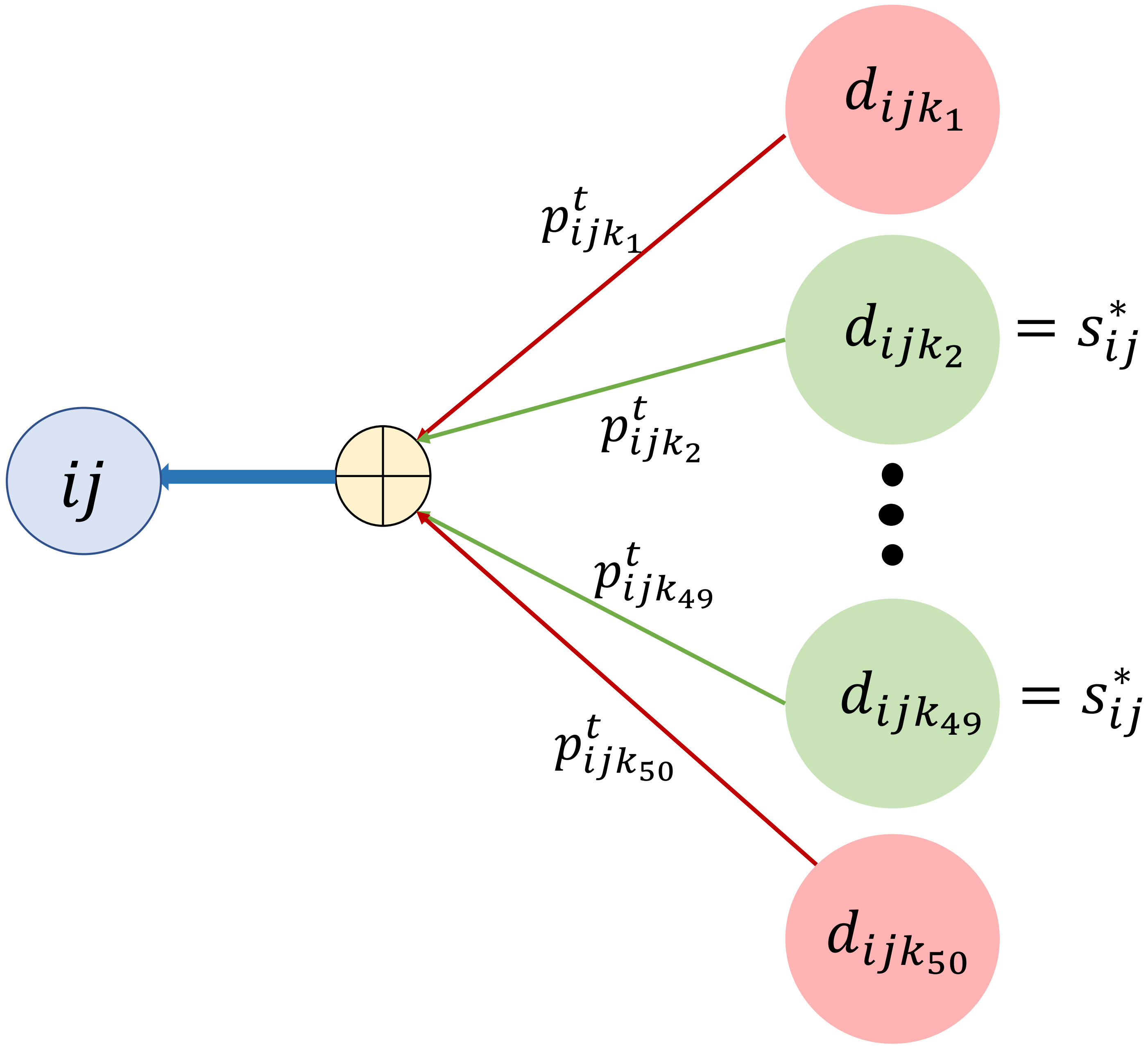
from 3-cycle inconsistency measure

$$d_{ijk} := d(R_{ij}R_{jk}R_{ki}, I)$$

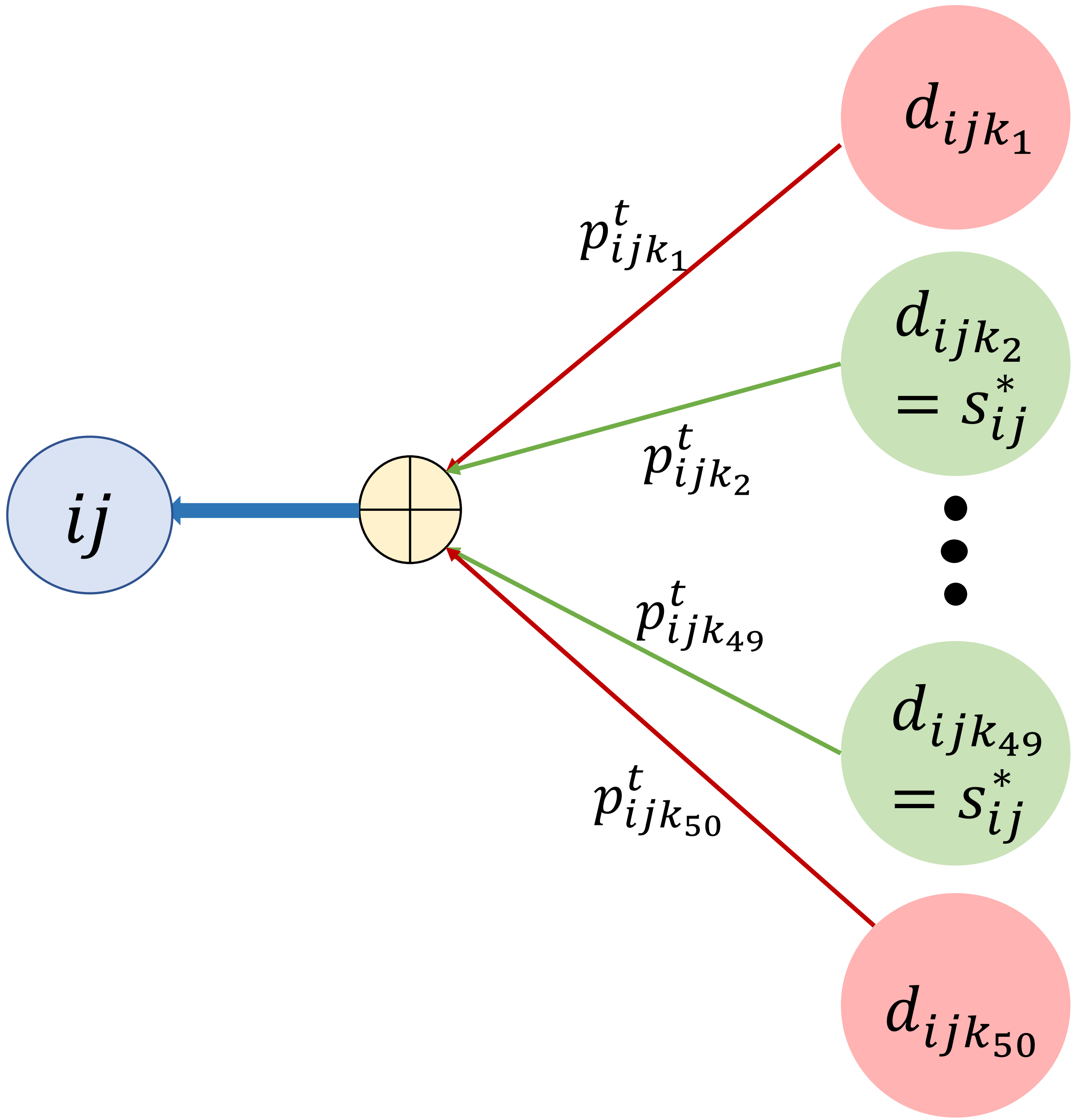
- For each $ij \in E$, sample 50 3-cycles ijk and for each cycle compute d_{ijk}
- For fixed ij and good 3-cycles w.r.t. ij (That is, $ik, jk \in E_g$)

$$d_{ijk} = d(R_{ij}R_{jk}^*R_{ki}^*, I) = d(R_{ij}\underbrace{R_{jk}^*R_{ki}^*R_{ij}^*}_{= I \text{ by cycle consistency}}, R_{ij}^*) = d(R_{ij}, R_{ij}^*) = s_{ij}^*$$


d_{ijk_1}
d_{ijk_2}
$d_{ijk_{49}}$
$d_{ijk_{50}}$



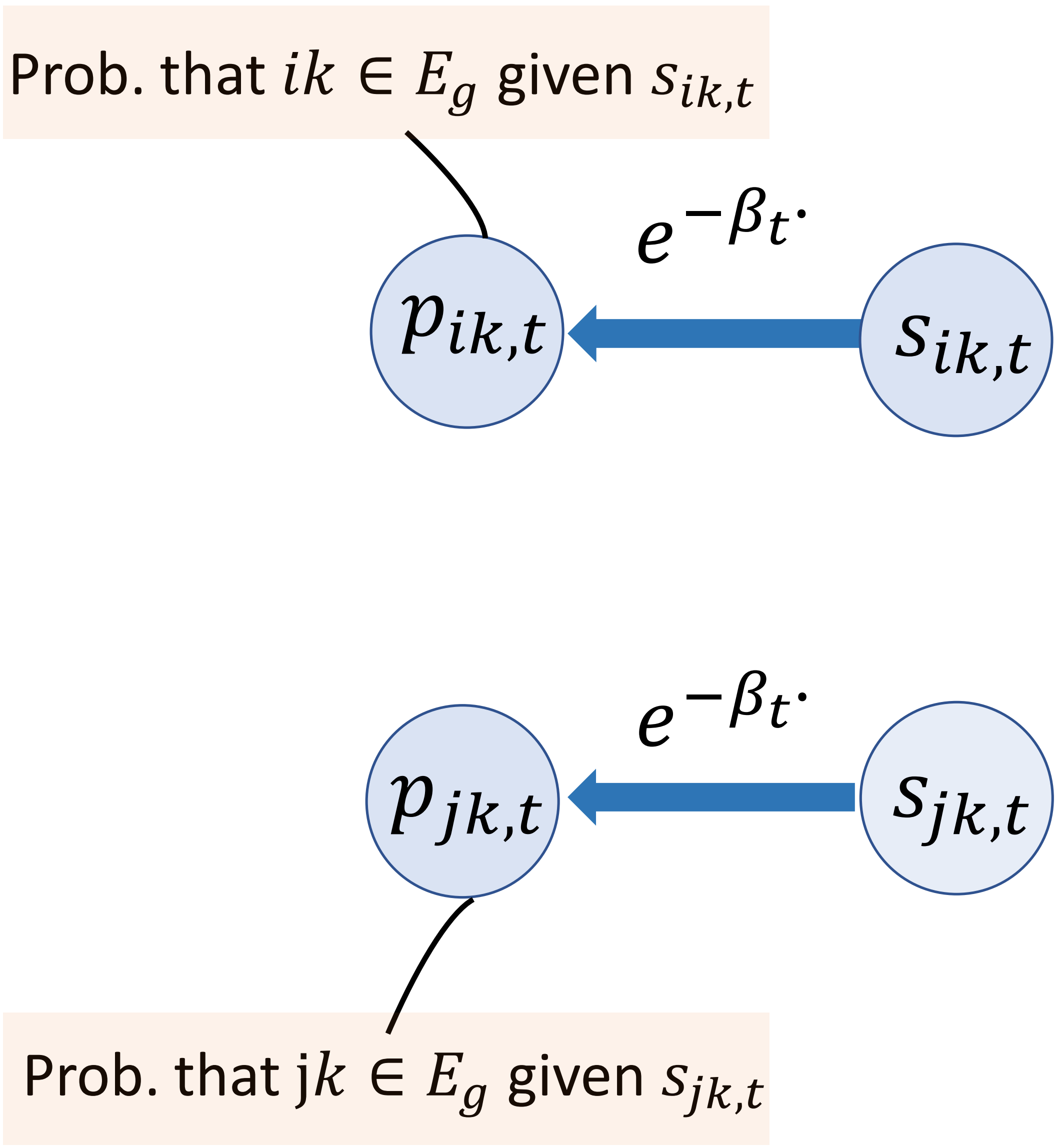
d_{ijk_1}
$p^t_{ijk_1}$
$d_{ijk_2} = s^*_{ij}$
$p^t_{ijk_2}$
ij
$p^t_{ijk_{49}}$
$d_{ijk_{49}} = s^*_{ij}$
$p^t_{ijk_{50}}$
$d_{ijk_{50}}$

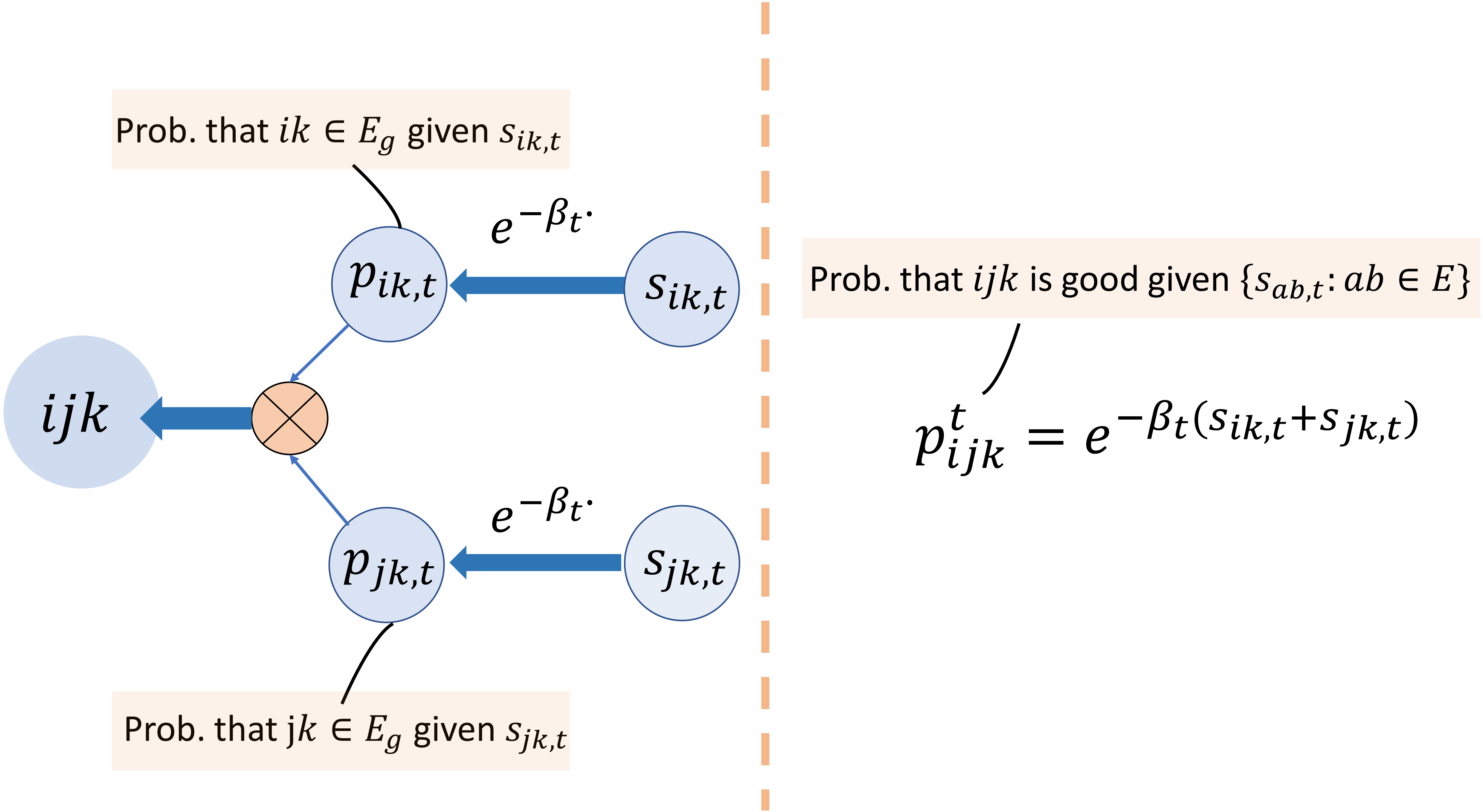


$$s^*_{ij} \approx s_{ij,t+1} := \frac{1}{Z^t_{ij}} \sum_k p^t_{ijk} d_{ijk}$$

$$Z^t_{ij} = \sum_k p^t_{ijk} d_{ijk}$$


$s_{ik,t}$
$s_{jk,t}$


Prob. that $ik \in E_g$ given $s_{ik,t}$
$e^{-\beta_t \cdot}$
$p_{ik,t}$
$s_{ik,t}$
$e^{-\beta_t \cdot}$
$p_{jk,t}$
$s_{jk,t}$
Prob. that j$k \in E_g$ given $s_{jk,t}$


Prob. that $ik \in E_g$ given $s_{ik,t}$
$e^{-\beta_t \cdot}$
$p_{ik,t}$
$s_{ik,t}$
ijk
$e^{-\beta_t \cdot}$
$p_{jk,t}$
$s_{jk,t}$
Prob. that j$k \in E_g$ given $s_{jk,t}$
Prob. that ijk is good given $\{s_{ab,t} : ab \in E\}$
$p_{ijk}^t = e^{-\beta_t(s_{ik,t}+s_{jk,t})}$

Cycle-Edge Message Passing (CEMP)

The conditional probability that ijk is good ($d_{ijk} = s_{ij}^*$):

$$p_{ijk}^t = e^{-\beta_t (s_{ik,t} + s_{jk,t})} = Pr(d_{ijk} = s_{ij}^* | \{s_{ab,t} : ab \in E\})$$

The estimate of the corruption level:

$$s_{ij,t+1} := \frac{1}{Z_{ij}^t} \sum_k p_{ijk}^t d_{ijk} = \mathbb{E}(s_{ij}^* | \{s_{ab,t} : ab \in E\})$$

Theory

If

- the maximal ratio of corrupted cycles per edge $< \frac{1}{5}$
- β_t increases exponentially with a sufficiently small rate,

then

- for all $ij \in E$, $s_{ij,t}$ computed by CEMP linearly and uniformly converges to s_{ij}^*.

Message Passing Least Squares (MPLS)

Initialization

Run CEMP for T iterations

Build a weighted graph with edge weights $\{s_{ij,T}\}_{ij\in E}$

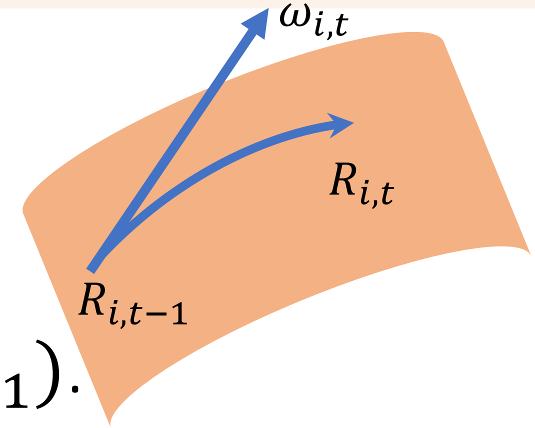
Find the minimal spanning tree using Prim's Algorithm

Initialize $R_{i,0}$ by fixing $R_{1,0} = I$ and $R_{i,0} = R_{ij}\, R_{j,0}$

Initialize weights $w_{ij,0} = F(s_{ij,T})$

For l_p minimization, $F(x) = x^{p-2}$

Message Passing Least Squares (MPLS)



$$\{\omega_{i,t}\}_{i\in[n]} = \underset{\omega_i\in\mathbb{R}^3}{\operatorname{argmin}} \sum_{ij\in E} w_{ij,t}\|\omega_i - \omega_j - \omega_{ij}\|_2^2$$

where $\omega_i = \log(R_{i,t-1}^{-1} R_i)$ and $\omega_{ij} = \log(R_{i,t-1}^{-1} R_{ij} R_{j,t-1})$.

After solving $\omega_{i,t}$, update $R_{i,t} = R_{i,t-1}\exp(\omega_{i,t})$.

Residual $r_{ij,t} := \|\omega_{i,t} - \omega_{j,t} - \omega_{ij}\|_2 \approx d\left(R_{i,t}R_{j,t}^{-1}, R_{ij}\right) \approx s_{ij}^*$

Reweighting

- Ideally, $w_{ij,t} = F\left(s_{ij}^*\right)$, where $F(x) = x^{p-2}$
- We set $w_{ij,t} = F\left(a_{ij,t}\right)$, where $a_{ij,t}$ is a better approximation of s_{ij}^* than $s_{ij,t}$

IRLS: $s_{ij}^{*} \approx r_{ij,t} \approx d\left(R_{i,t}R_{j,t}^{-1}, R_{ij}\right)$ and $w_{ij,t+1} = F\left(r_{ij,t}\right)$

CEMP: $s_{ij}^{*} \approx s_{ij,t} := \frac{1}{Z_{ij}^{t}} \sum_k p_{ijk}^{t-1} d_{ijk}$ and $p_{ijk}^{t-1} = e^{-\beta_t (s_{ik,t-1} + s_{jk,t-1})}$

MPLS: $s_{ij}^{*} \approx a_{ij,t} := \alpha_t\, h_{ij,t} + (1-\alpha_t)\, r_{ij,t}$ and $w_{ij,t+1} = F\left(a_{ij,t}\right)$

where $h_{ij,t} := \frac{1}{Z_{ij}^{t}} \sum_k q_{ijk}^{t} d_{ijk}$ and $q_{ijk}^{t} = e^{-\beta_T (r_{ik,t} + r_{jk,t})}$

IRLS: $s_{ij}^* \approx r_{ij,t} \approx d\left(R_{i,t}R_{j,t}^{-1}, R_{ij}\right)$ and $w_{ij,t+1} = F\left(r_{ij,t}\right)$

CEMP: $s_{ij}^* \approx s_{ij,t} := \frac{1}{Z_{ij}^t} \sum_k p_{ijk}^{t-1} d_{ijk}$ and $p_{ijk}^{t-1} = e^{-\beta_t (s_{ik,t-1} + s_{jk,t-1})}$

MPLS: $s_{ij}^* \approx a_{ij,t} := \alpha_t\, h_{ij,t} + (1 - \alpha_t)\, r_{ij,t}$ and $w_{ij,t+1} = F\left(a_{ij,t}\right)$

where $h_{ij,t} := \frac{1}{Z_{ij}^t} \sum_k q_{ijk}^t d_{ijk}$ and $q_{ijk}^t = e^{-\beta_T (r_{ik,t} + r_{jk,t})}$

$$h_{ij,t} = \mathbb{E}(s_{ij}^* | \{r_{ab,t} : ab \in E\})$$

MPLS: $a_{ij,t-1} \to w_{ij,t}$

$w_{ij,t}$: weights

MPLS: $a_{ij,t-1} \to w_{ij,t} \to R_{ij,t}$

$w_{ij,t}$: weights

$R_{ij,t}$: estimated relative rotations $R_{i,t}R_{j,t}^{-1}$

MPLS: $a_{ij,t-1} \rightarrow w_{ij,t} \rightarrow R_{ij,t} \rightarrow r_{ij,t}$

$w_{ij,t}$: weights

$R_{ij,t}$: estimated relative rotations $R_{i,t}R_{j,t}^{-1}$

$r_{ij,t}$: residuals

MPLS: $a_{ij,t-1} \to w_{ij,t} \to R_{ij,t} \to r_{ij,t} \to h_{ij,t}$

$w_{ij,t}$: weights

$R_{ij,t}$: estimated relative rotations $R_{i,t}R_{j,t}^{-1}$

$r_{ij,t}$: residuals

$h_{ij,t}$: analogue of $s_{ij,t}$ (Note that $s_{ij,t}$ =CEMP($s_{ij,t-1}$) and $h_{ij,t}$ =CEMP($r_{ij,t}$))

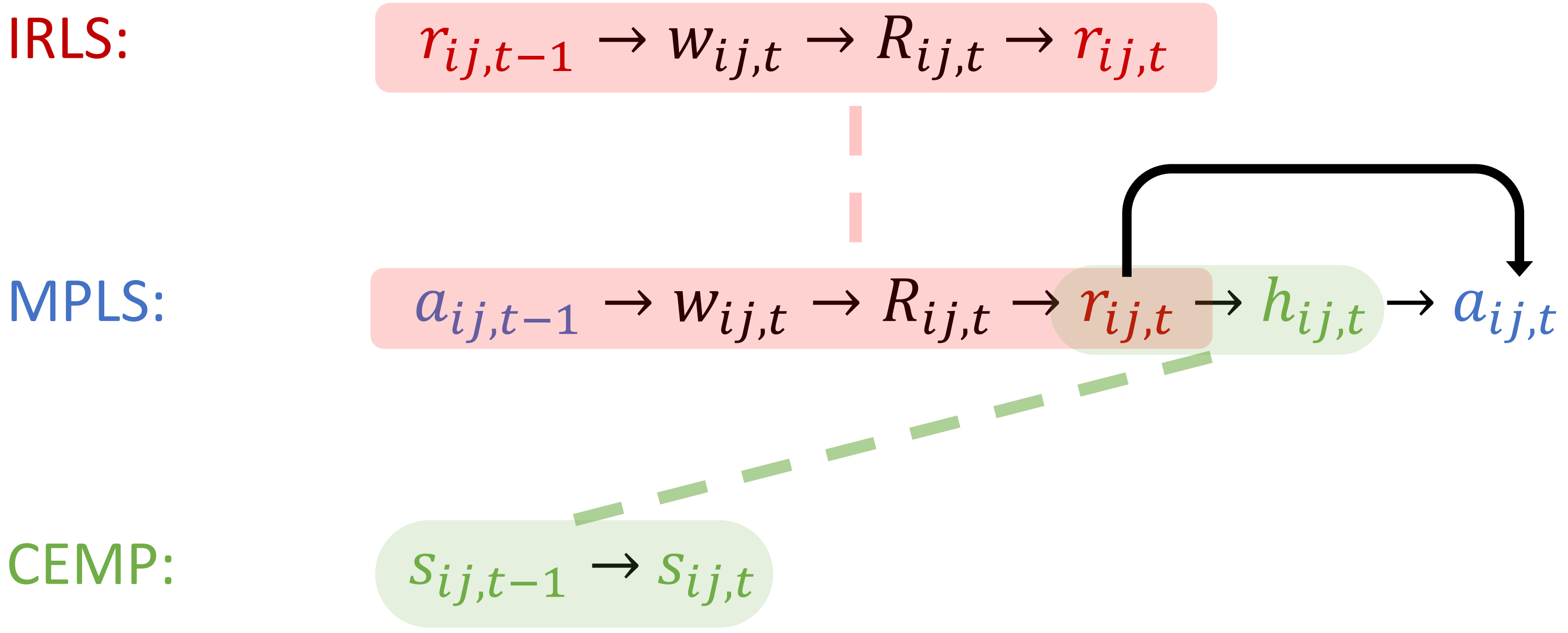
MPLS: $a_{ij,t-1} \rightarrow w_{ij,t} \rightarrow R_{ij,t} \rightarrow r_{ij,t} \rightarrow h_{ij,t} \rightarrow a_{ij,t}$

$w_{ij,t}$: weights

$R_{ij,t}$: estimated relative rotations $R_{i,t}R_{j,t}^{-1}$

$r_{ij,t}$: residuals

$h_{ij,t}$: analogue of $s_{ij,t}$ (Note that $s_{ij,t}$ =CEMP($s_{ij,t-1}$) and $h_{ij,t}$ =CEMP($r_{ij,t}$))



$w_{ij,t}$: weights

$R_{ij,t}$: estimated relative rotations $R_{i,t}R_{j,t}^{-1}$

$r_{ij,t}$: residuals

$h_{ij,t}$: analogue of $s_{ij,t}$ (Note that $s_{ij,t}$ =CEMP($s_{ij,t-1}$) and $h_{ij,t}$ =CEMP($r_{ij,t}$))

Experiments

$T = 5,\ \beta_t = 2^t$ for $t = 0, \ldots, 5.$

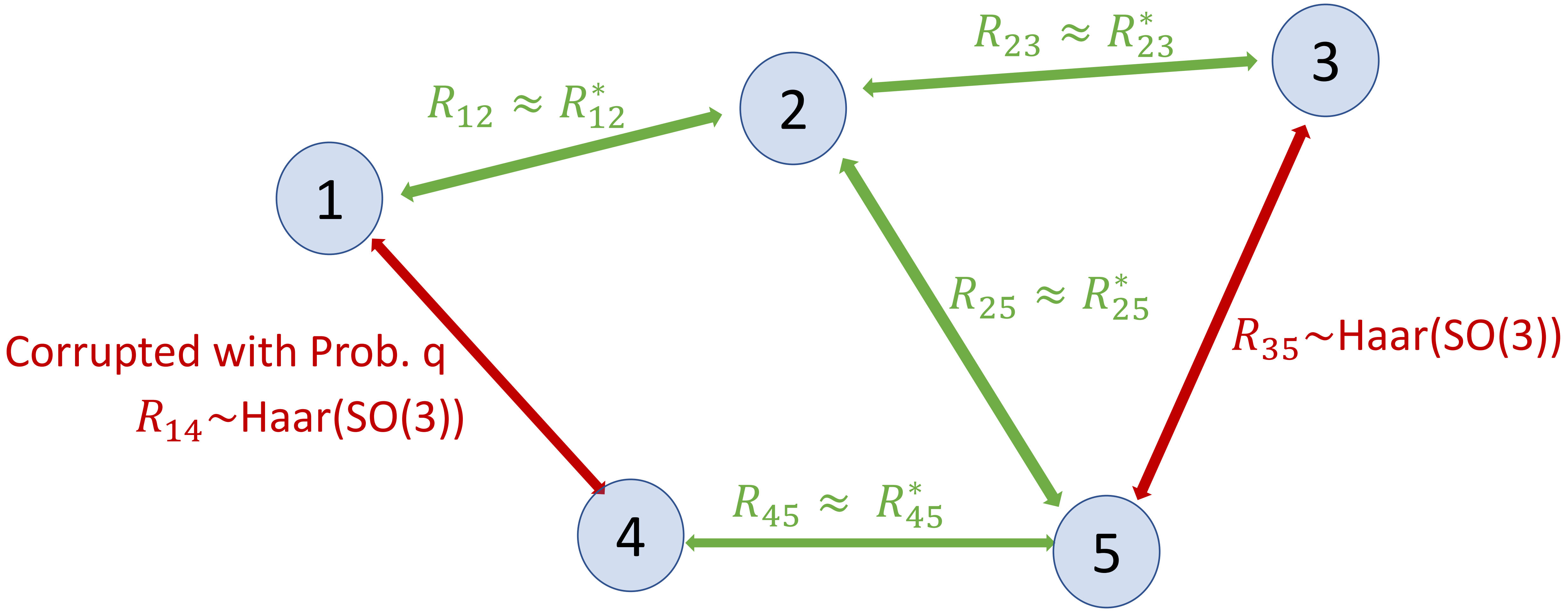
For $t > 0$

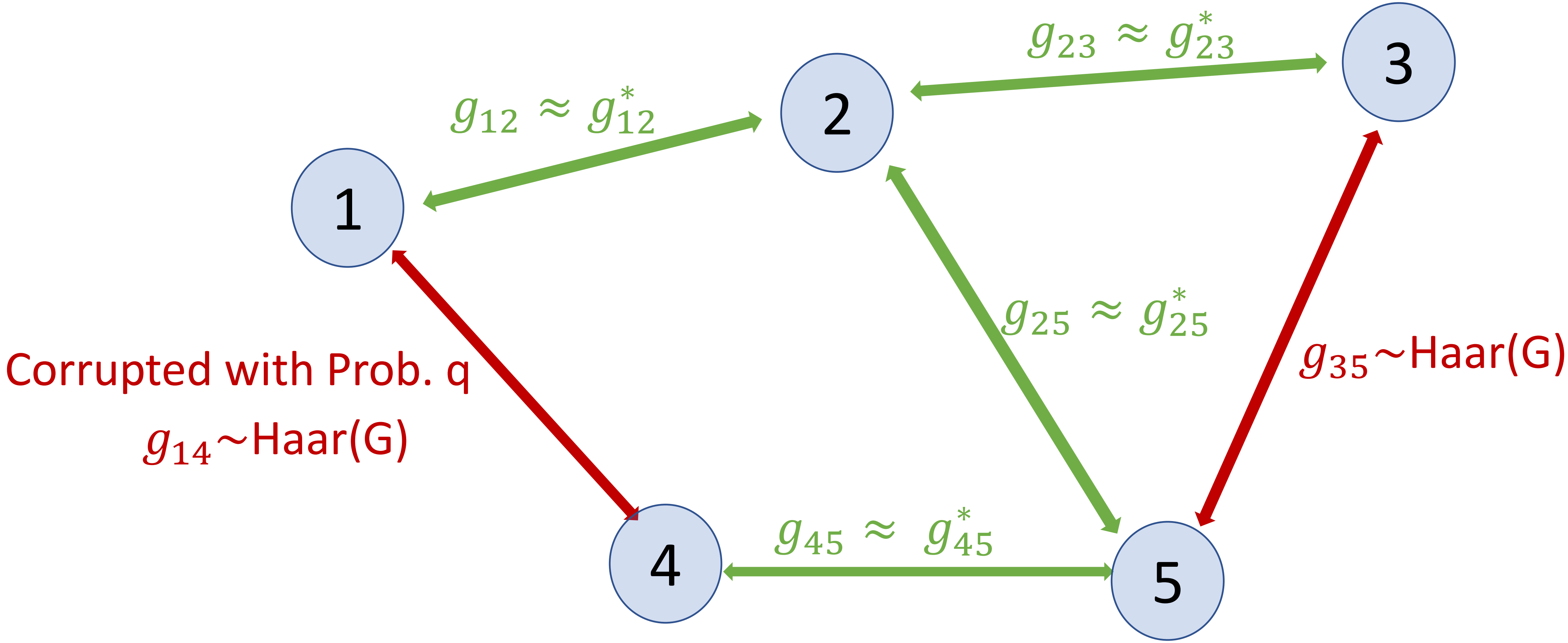
$$\alpha_t = \frac{1}{t+1}, \qquad F(x) = x^{-\frac{3}{2}} \cdot 1(x < \tau_t)$$

$l_{1/2}$ minimization

Additional Thresholding

Uniform Corruption Model

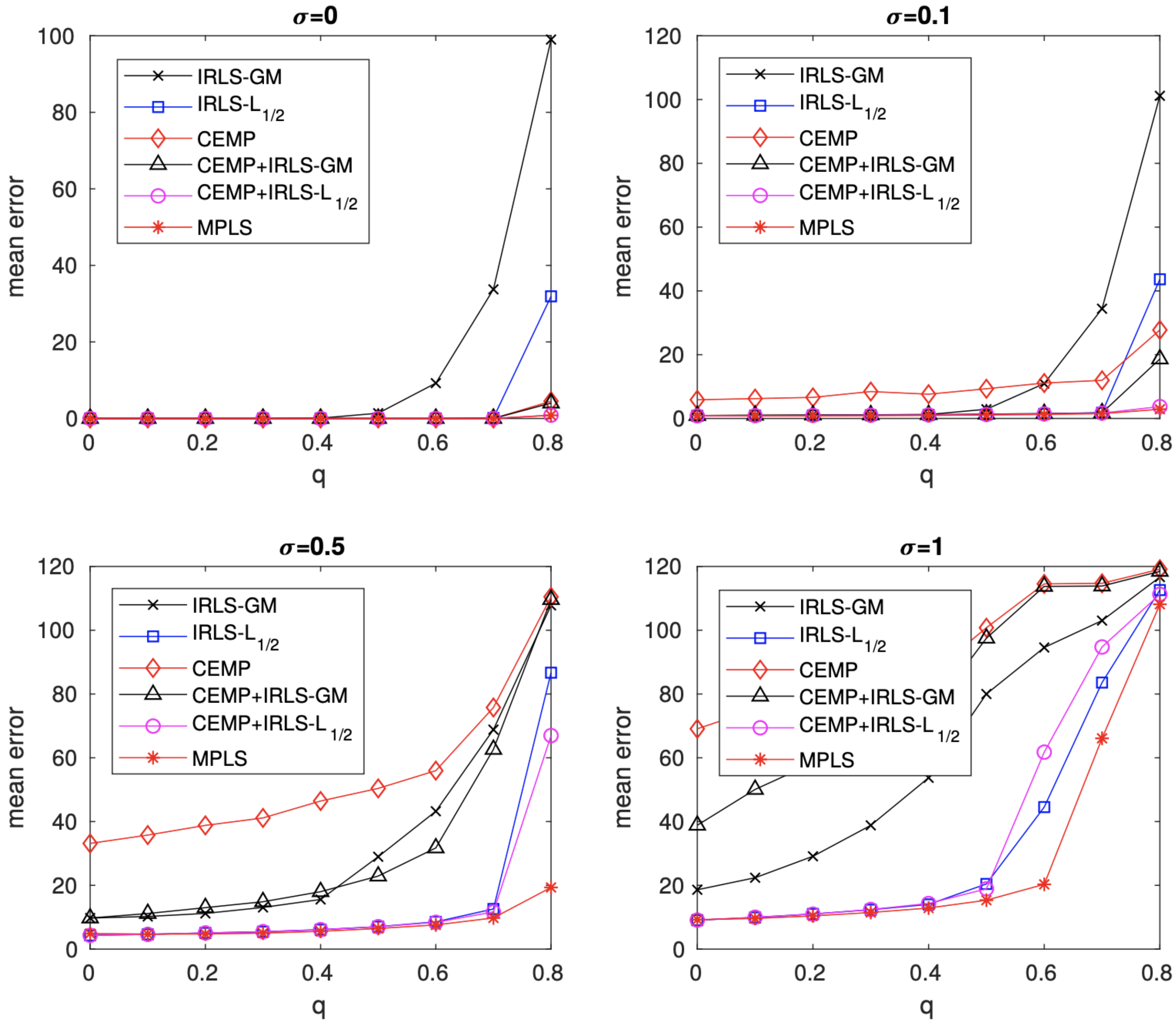




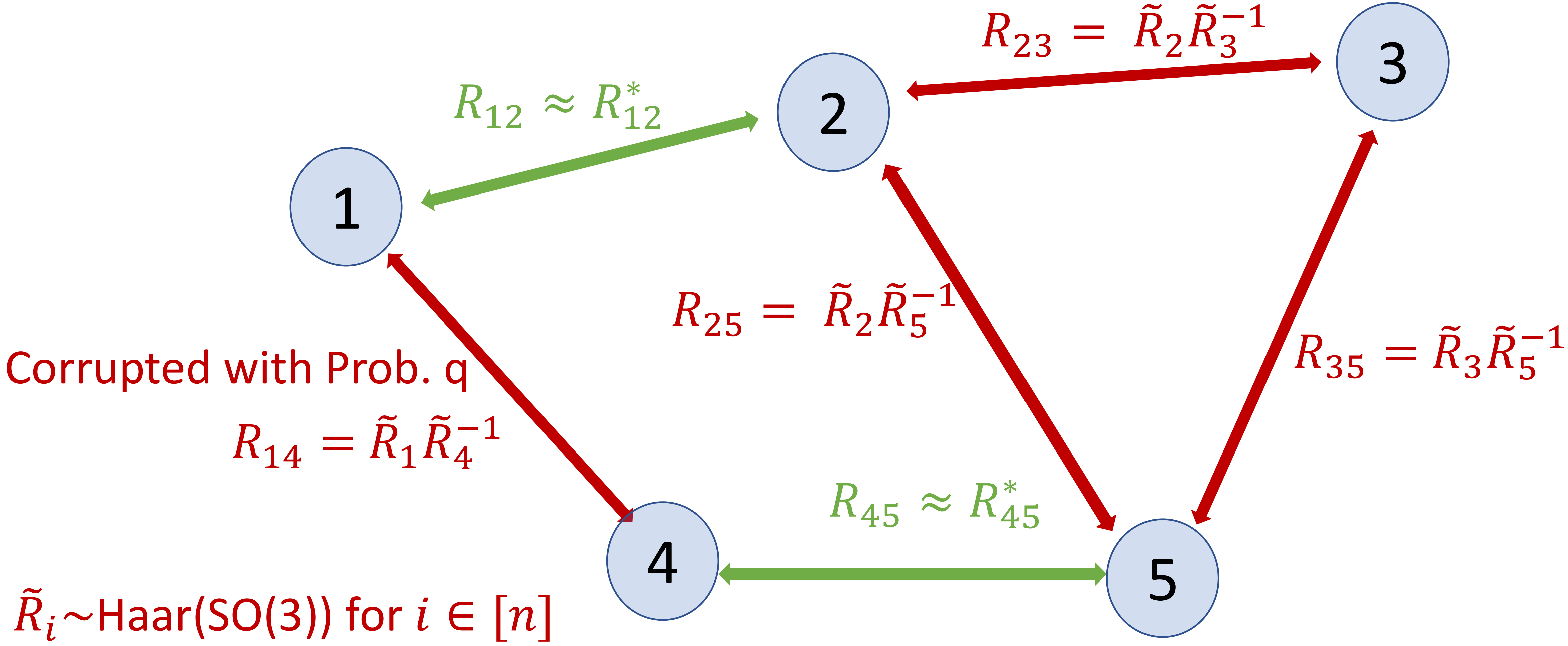
$g_{12} \approx g_{12}^*$
$g_{23} \approx g_{23}^*$
1
2
3
Corrupted with Prob. q
$g_{14} \sim \text{Haar}(G)$
$g_{25} \approx g_{25}^*$
$g_{35} \sim \text{Haar}(G)$
$g_{45} \approx g_{45}^*$
4
5

Uniform Corruption

q: prob. of corruption
σ: noise level

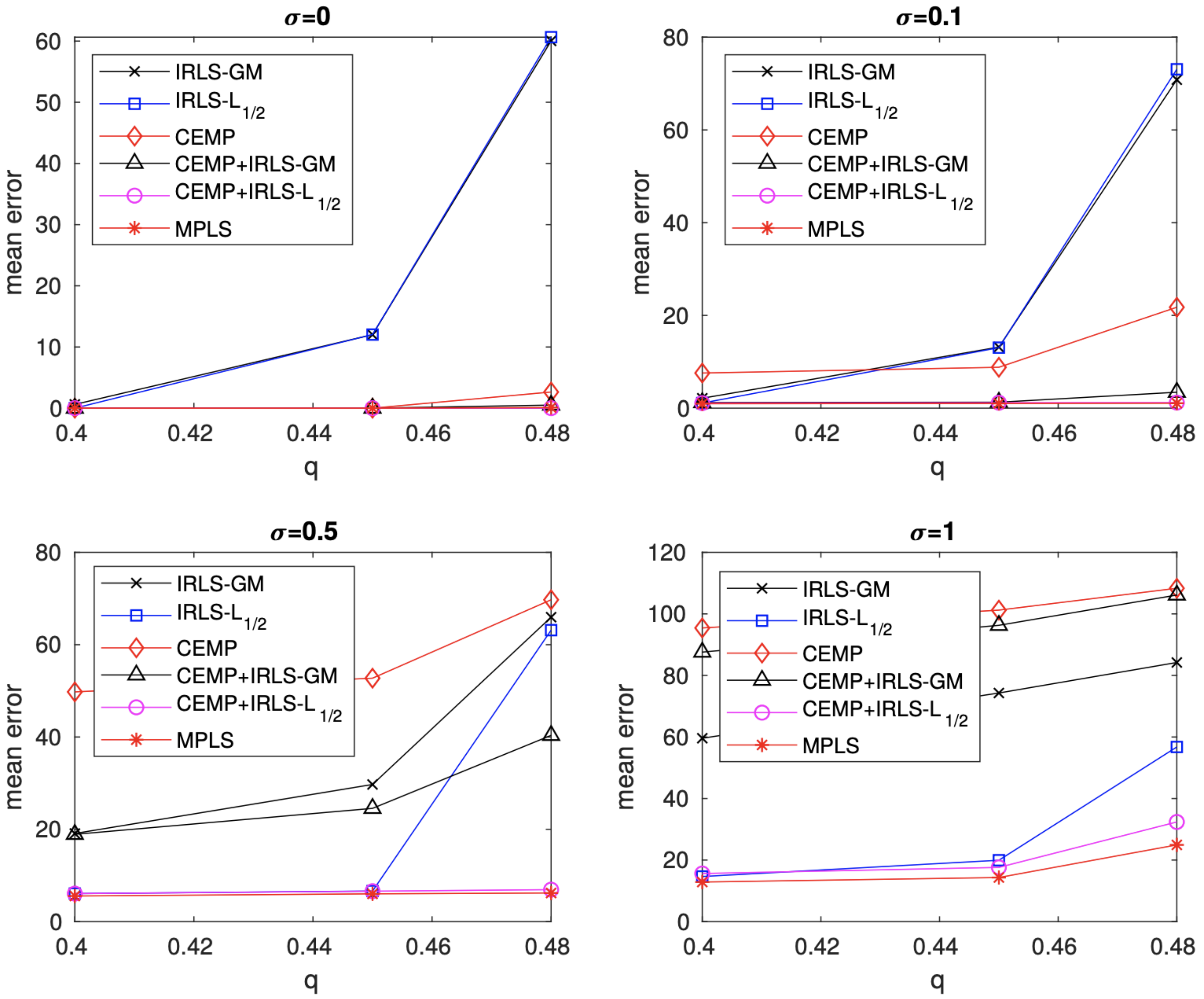


Self-consistent Corruption Model



Self-Consistent Corruption

q: prob. of corruption
σ: noise level



Algorithms	IRLS-GM		IRLS-$\ell_{\frac{1}{2}}$		CEMP (Ours)		MPLS (Ours)	
Dataset	$\tilde{e}$	runtime	$\tilde{e}$	runtime	$\tilde{e}$	runtime	$\tilde{e}$	runtime
Alamo	3.64	14.2	3.67	15.5	4.05	**10.38**	**3.44**	20.6
Ellis Island	3.04	3.2	2.71	2.8	2.94	**2.4**	**2.61**	4.0
Gendarmenmarkt	**39.24**	6.5	39.41	7.3	45.33	**4.7**	44.94	17.8
Madrid Metropolis	5.30	3.8	4.88	2.7	5.10	**2.1**	**4.65**	5.2
Montreal N.D.	1.25	6.5	1.22	7.3	1.33	**6.3**	**1.04**	9.3
Notre Dame	2.63	17.2	2.26	22.5	2.35	**13.2**	**2.06**	31.5
NYC Library	2.71	2.5	2.66	2.6	3.00	**1.9**	**2.63**	4.5
Piazza Del Popolo	4.10	2.8	3.99	3.1	**3.44**	**2.6**	3.73	3.5
Piccadilly	5.12	153.5	5.19	170.2	4.66	**45.8**	**3.93**	191.9
Roman Forum	2.66	8.6	2.69	11.4	2.80	**6.1**	**2.62**	8.8
Tower of London	3.42	2.6	3.41	2.4	**2.84**	**2.2**	3.16	2.7
Union Square	6.77	5.0	6.77	5.6	7.47	**2.5**	**6.54**	5.7
Vienna Cathedral	8.13	28.3	8.07	45.4	**6.91**	**13.1**	7.21	42.6
Yorkminster	2.60	**2.4**	**2.45**	3.3	2.49	2.8	2.47	3.9

Algorithms	IRLS-GM		IRLS-$\ell_{\frac{1}{2}}$		CEMP (Ours)		MPLS (Ours)	
Dataset	$\tilde{e}$	runtime	$\tilde{e}$	runtime	$\tilde{e}$	runtime	$\tilde{e}$	runtime
Alamo	3.64	14.2	3.67	15.5	4.05	**10.38**	**3.44**	20.6
Ellis Island	3.04	3.2	2.71	2.8	2.94	**2.4**	**2.61**	4.0
Gendarmenmarkt	**39.24**	6.5	39.41	7.3	45.33	**4.7**	44.94	17.8
Madrid Metropolis	5.30	3.8	4.88	2.7	5.10	**2.1**	**4.65**	5.2
Montreal N.D.	1.25	6.5	1.22	7.3	1.33	**6.3**	**1.04**	9.3
Notre Dame	2.63	17.2	2.26	22.5	2.35	**13.2**	**2.06**	31.5
NYC Library	2.71	2.5	2.66	2.6	3.00	**1.9**	**2.63**	4.5
Piazza Del Popolo	4.10	2.8	3.99	3.1	**3.44**	**2.6**	3.73	3.5
Piccadilly	5.12	153.5	5.19	170.2	4.66	**45.8**	**3.93**	191.9
Roman Forum	2.66	8.6	2.69	11.4	2.80	**6.1**	**2.62**	8.8
Tower of London	3.42	2.6	3.41	2.4	**2.84**	**2.2**	3.16	2.7
Union Square	6.77	5.0	6.77	5.6	7.47	**2.5**	**6.54**	5.7
Vienna Cathedral	8.13	28.3	8.07	45.4	**6.91**	**13.1**	7.21	42.6
Yorkminster	2.60	**2.4**	**2.45**	3.3	2.49	2.8	2.47	3.9

Conclusion

- We proposed the MPLS framework for robustly solving rotation synchronization
- Our reweighting strategy is more reliable under high corruption and noise
- Future directions:

  theory for MPLS (exact recovery and convergence); more applications;

  adaptive reweighting parameters/optimal reweighting functions

Thank you!