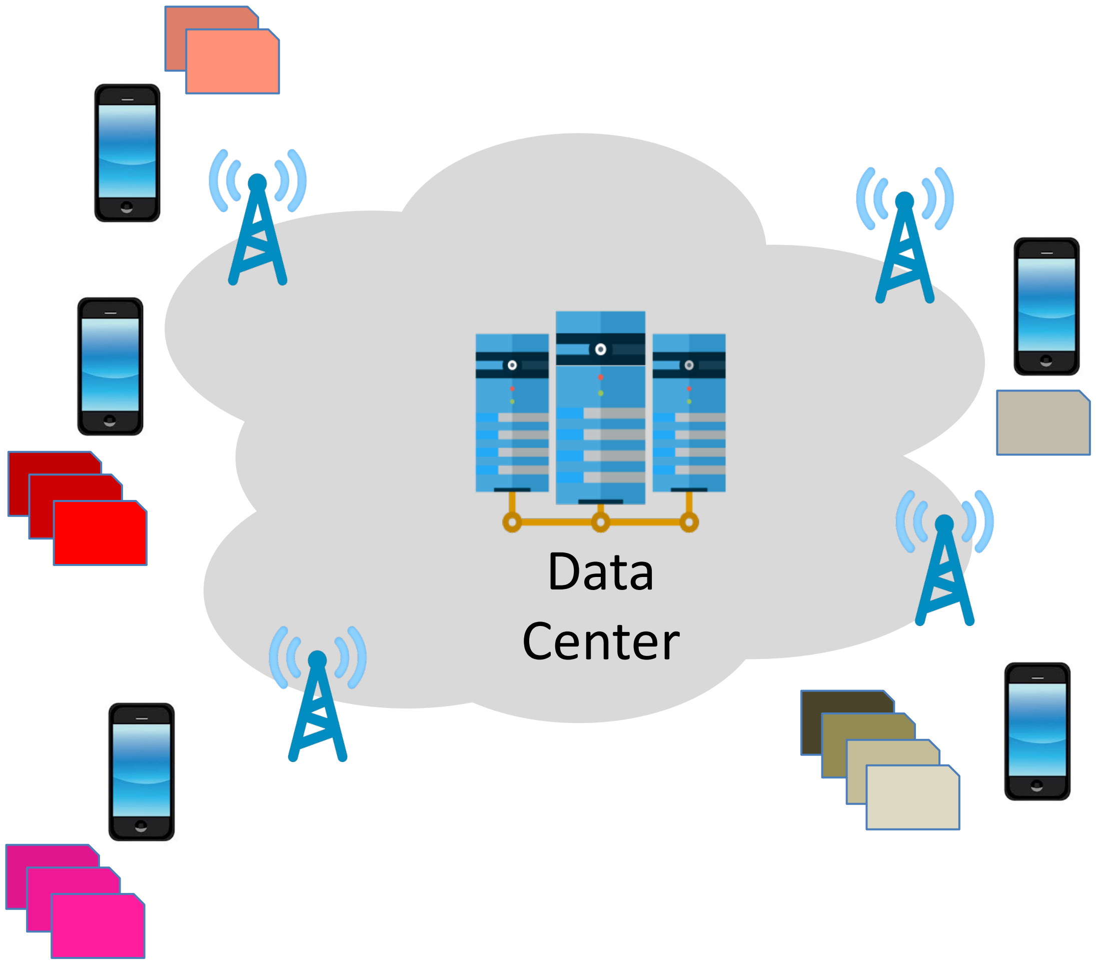
Scalable Decentralized Algorithms for Online Personalized Mean Estimation

Franco Galante, Giovanni Neglia, Emilio Leonardi

Atélier en Évaluation des Performances, Toulouse, 2-4/12/2024

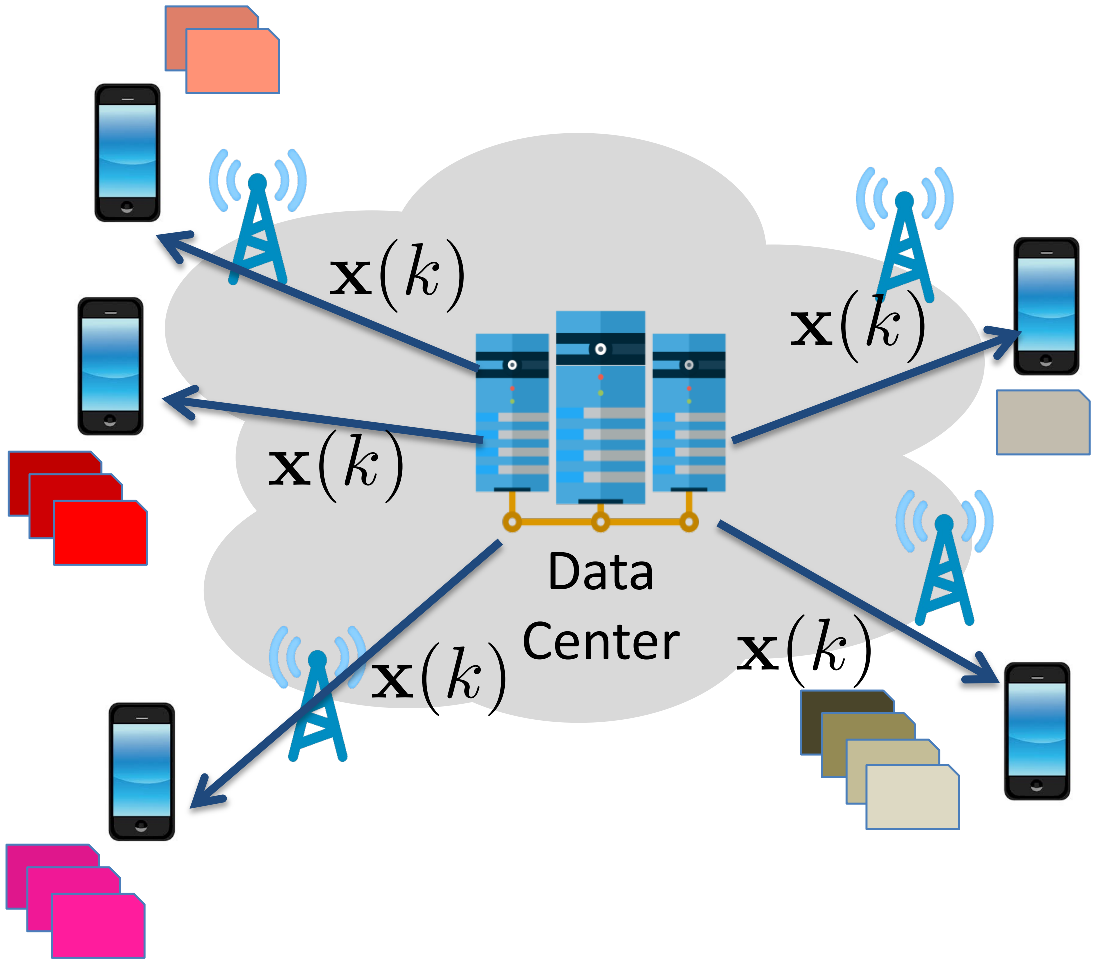
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

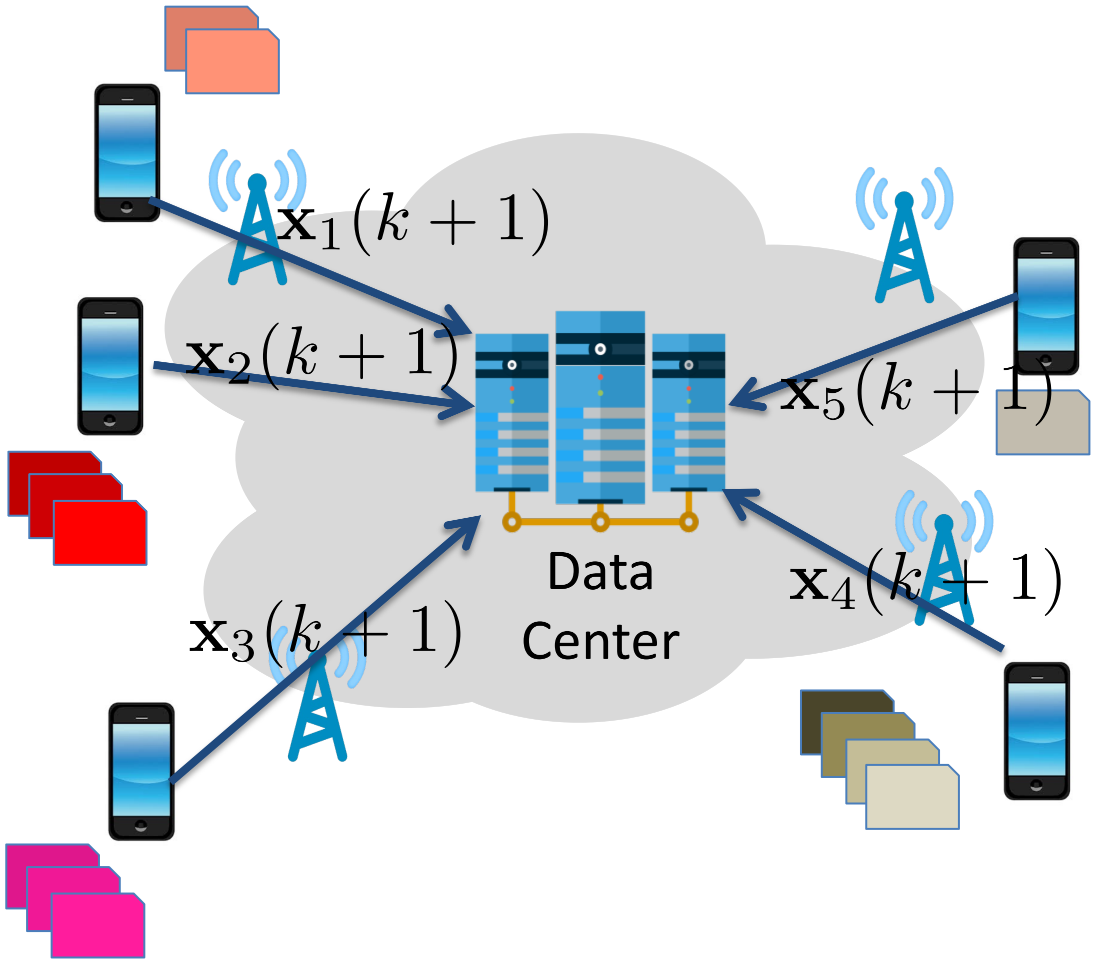
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

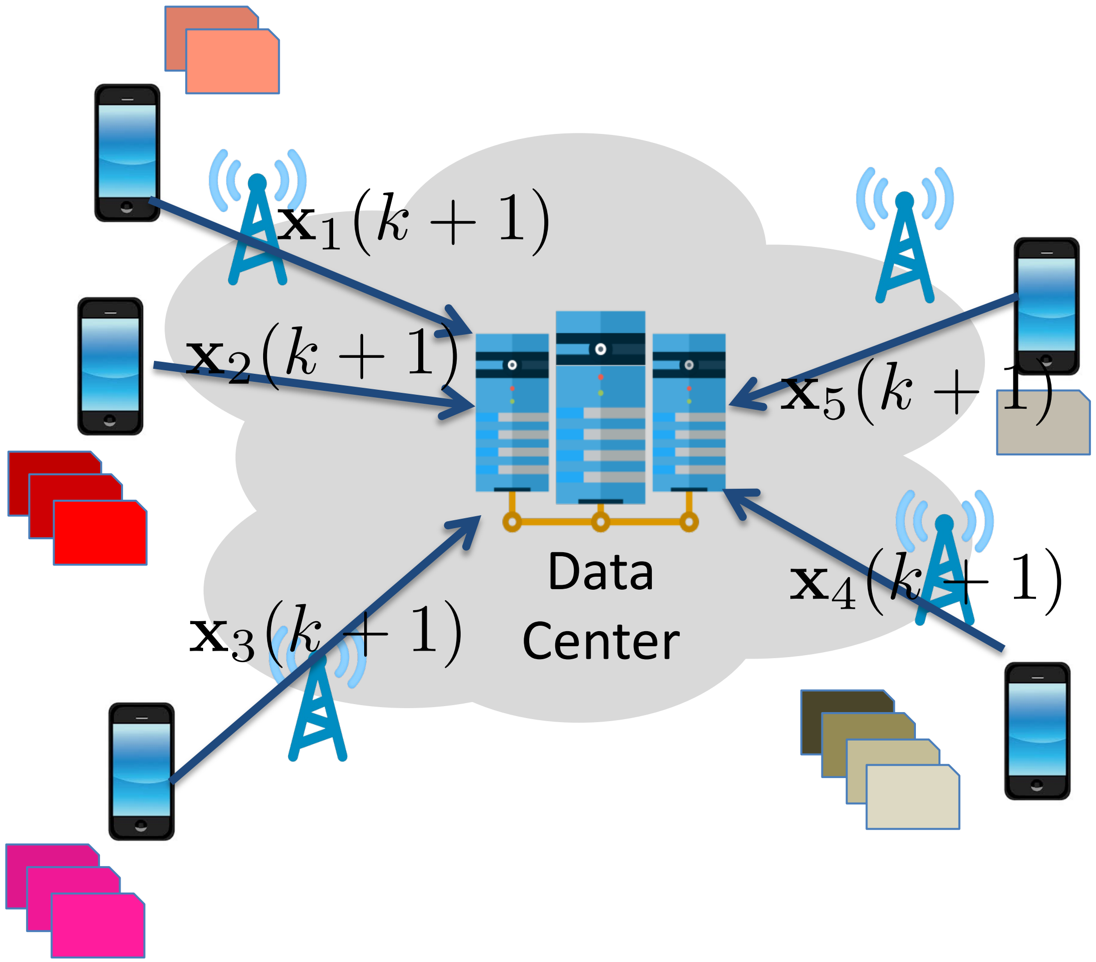
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

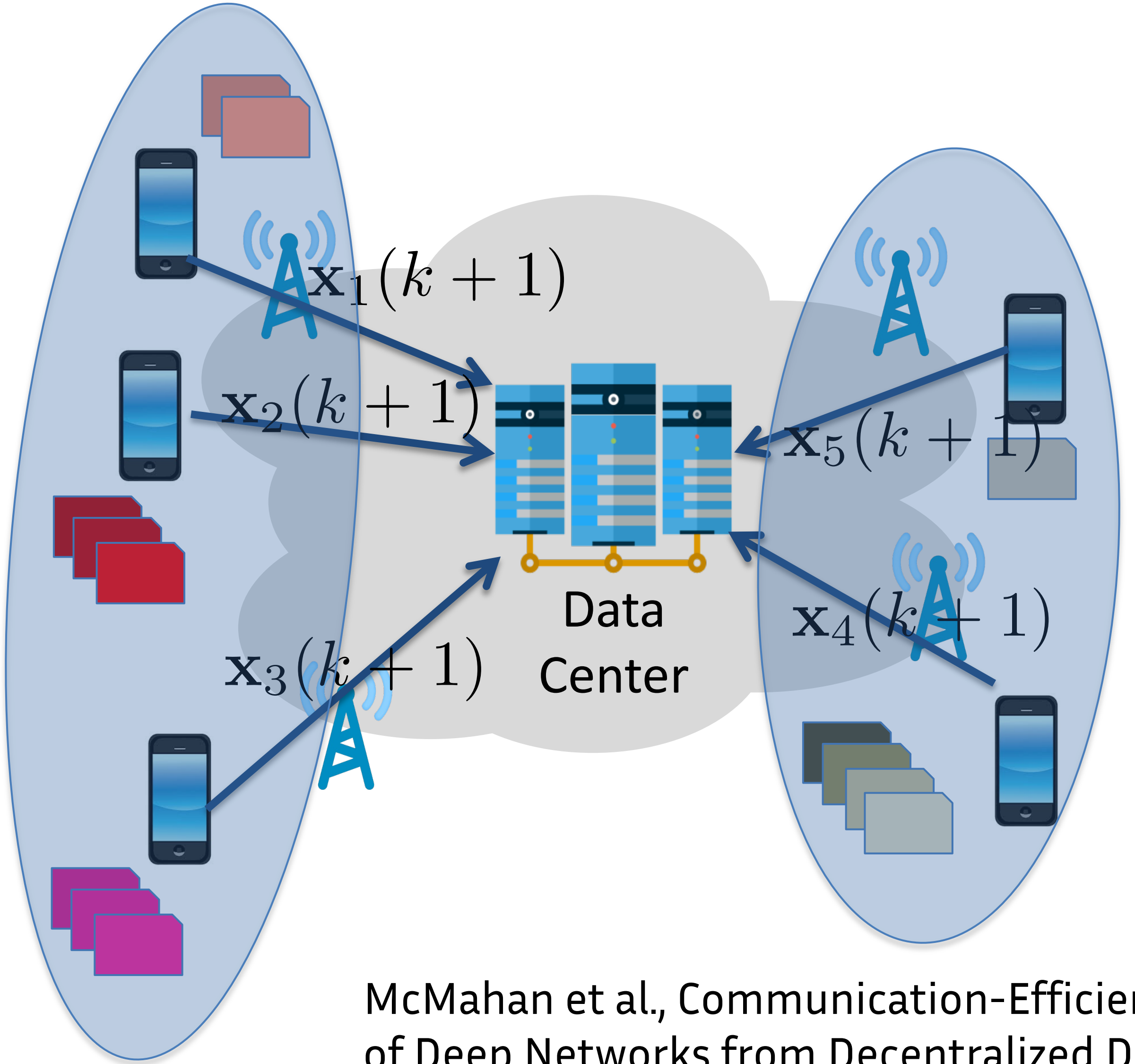
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy
- Bias-variance tradeoff

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

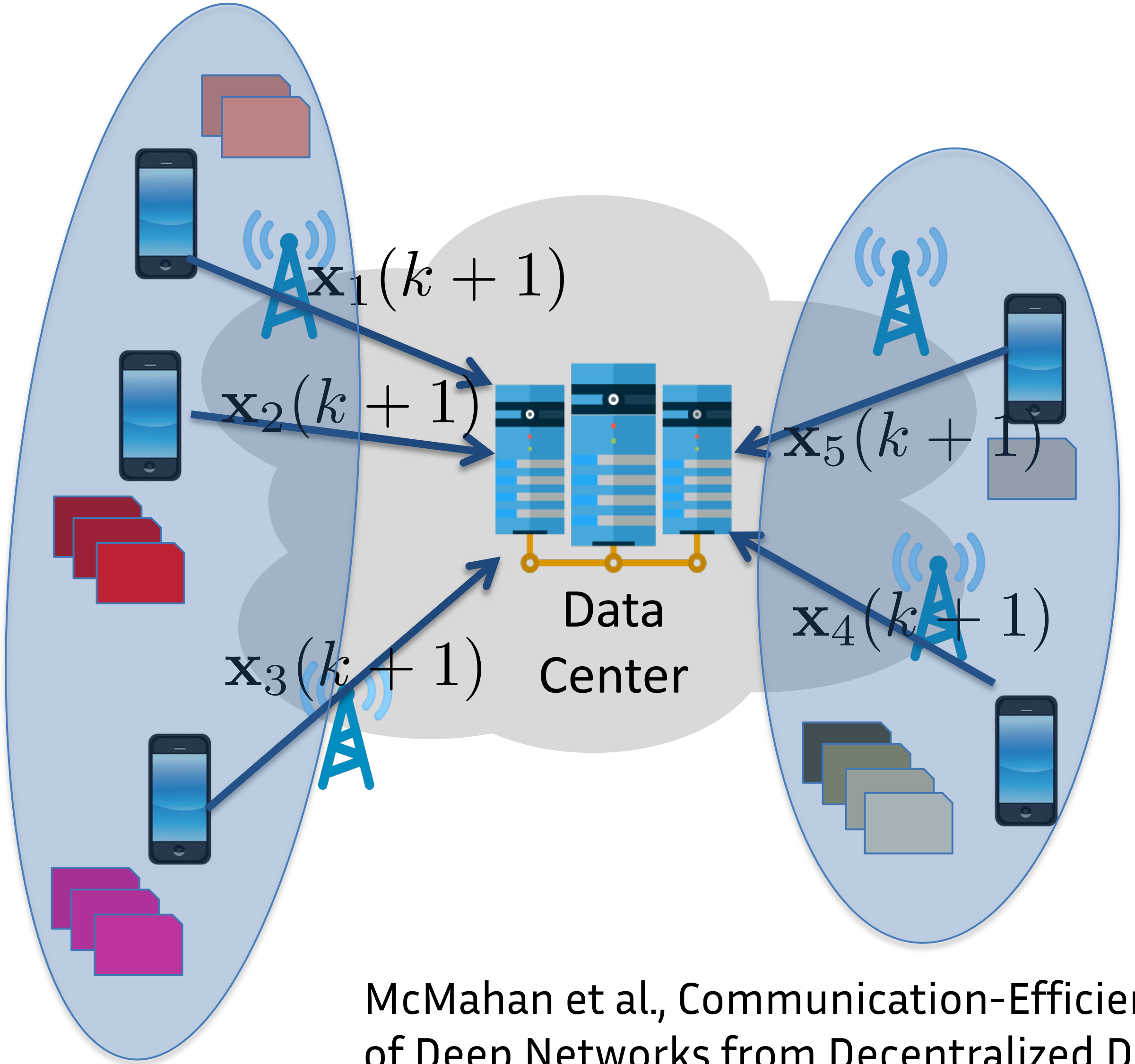
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy
- Bias-variance tradeoff

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

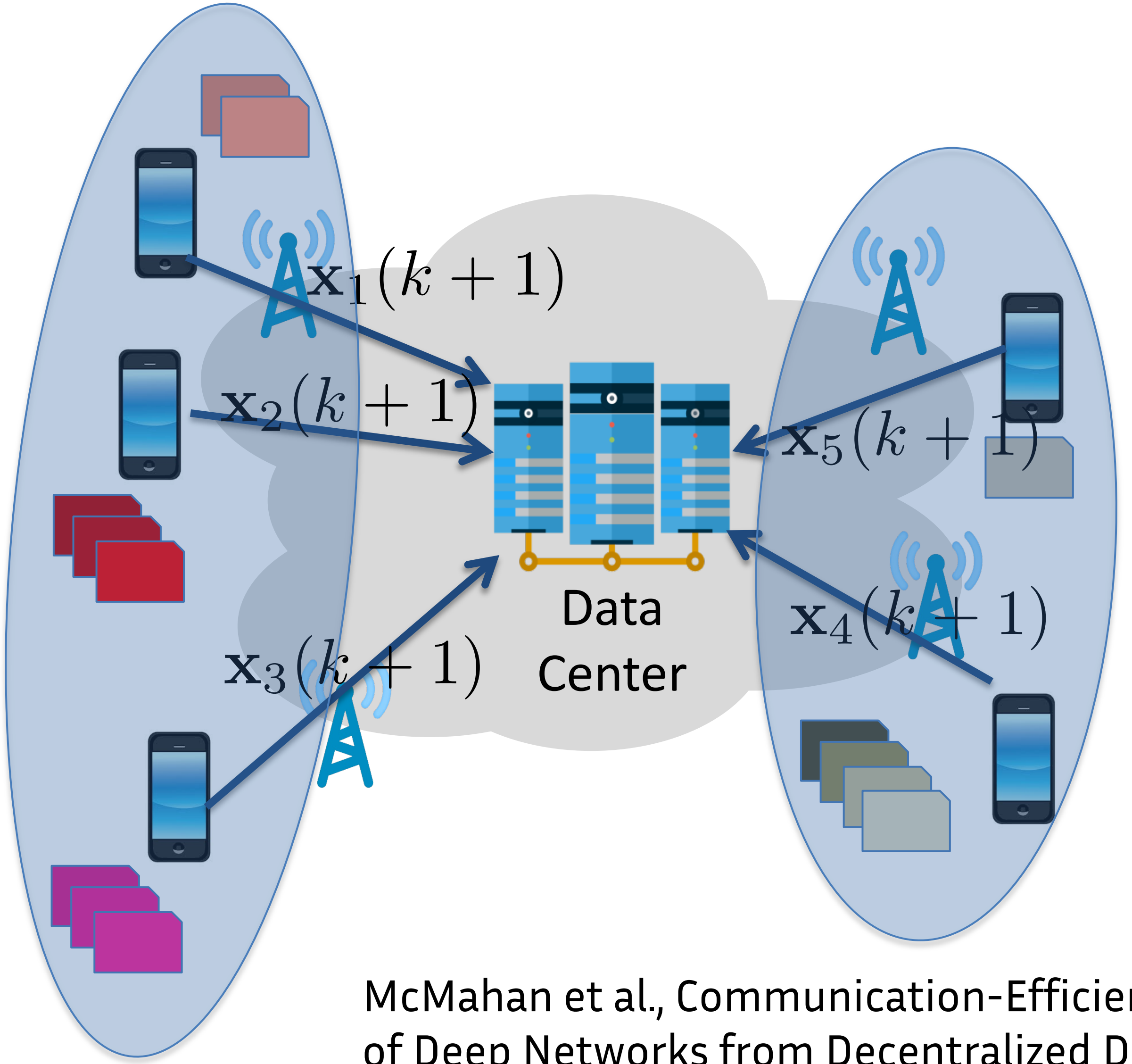
Federated Learning (Google&Apple)



- Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy
- Bias-variance tradeoff
- Learn a different model for each cluster of similar clients

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

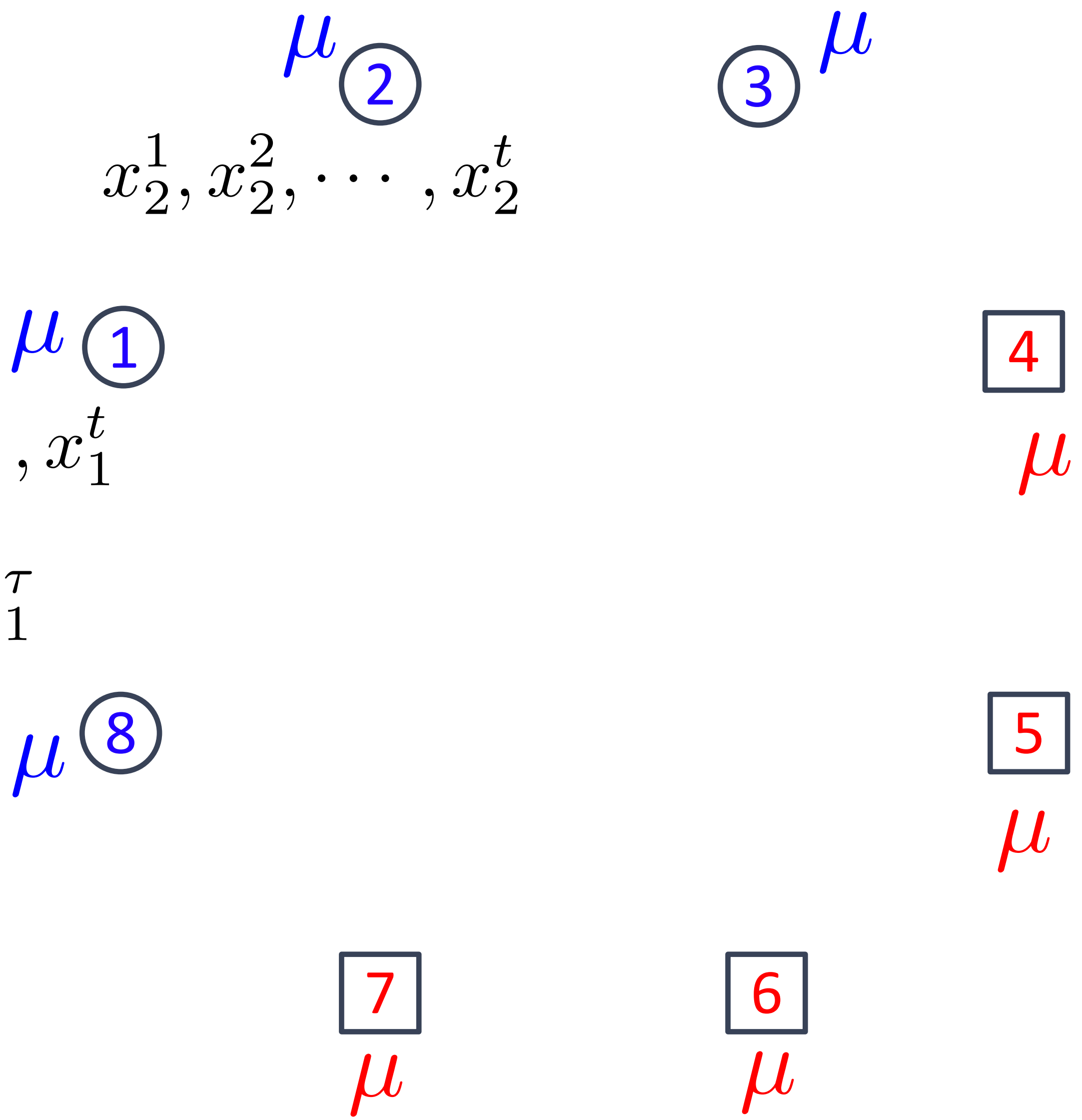
Federated Learning (Google&Apple)



- ➢ Train ML models keeping data local
  - transfer costs and privacy concerns…
  - but also energy
- ➢ Bias-variance tradeoff
- ➢ Learn a different model for each cluster of similar clients
  - similarity needs to be learned in parallel

McMahan et al., Communication-Efficient Learning of Deep Networks from Decentralized Data, 2017

Toy Model



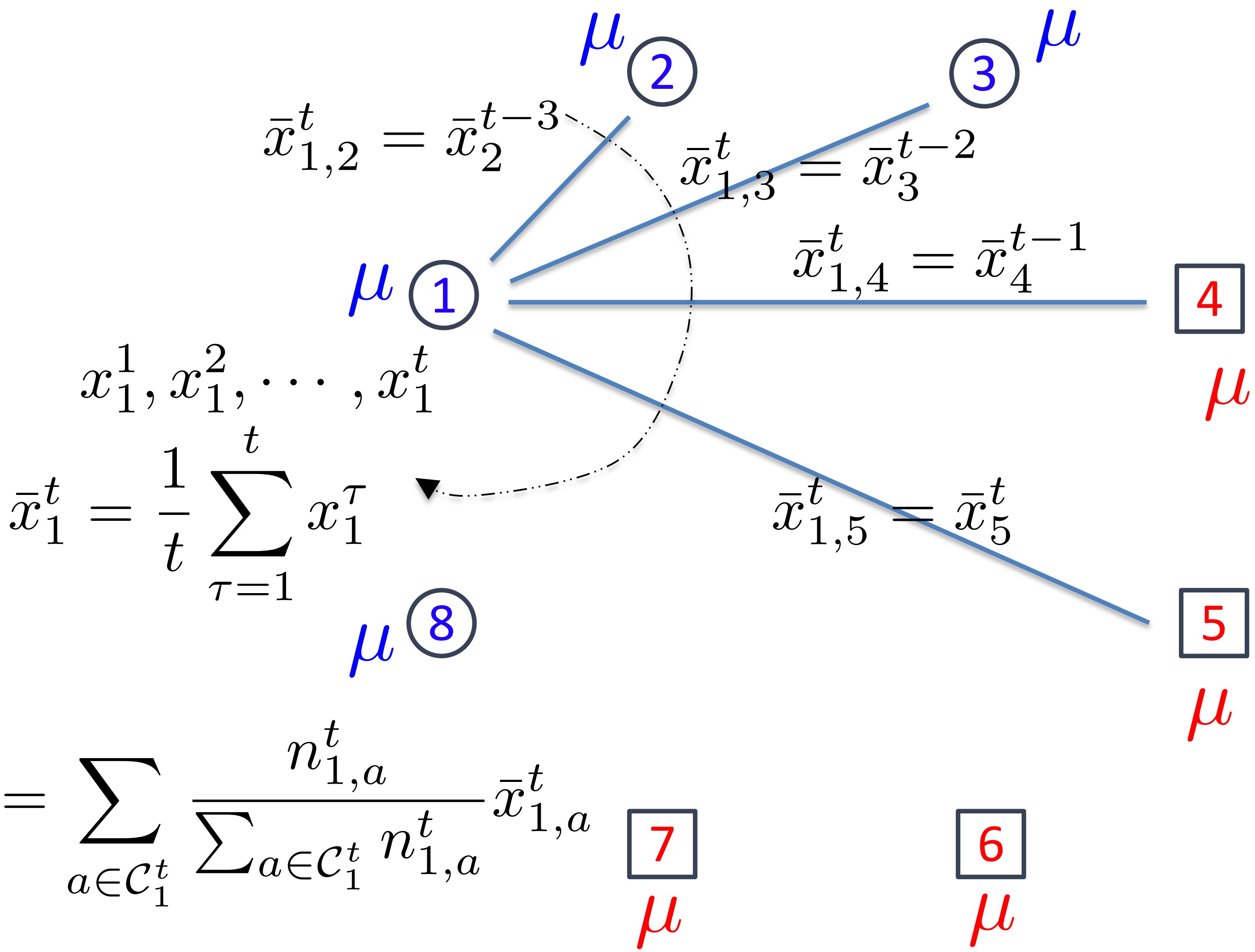
$$x_2^1, x_2^2, \cdots, x_2^t$$

$$x_1^1, x_1^2, \cdots, x_1^t$$

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$

- Each agent receives one sample per slot drawn from a (potentially) different distribution
- Each agent a wants to estimate its true mean μ_a

Collaborative Mean Estimation (Asadi et al, 2023)



$$\hat{\mu}_1^t = \sum_{a \in \mathcal{C}_1^t} \frac{n_{1,a}^t}{\sum_{a \in \mathcal{C}_1^t} n_{1,a}^t} \bar{x}_{1,a}^t$$

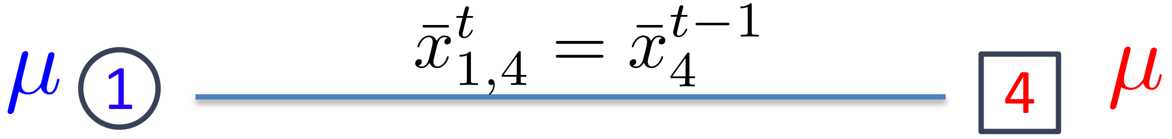
- Each agent receives one sample per slot drawn from a (potentially) different distribution
- Each agent a wants to estimate its true mean μ_a

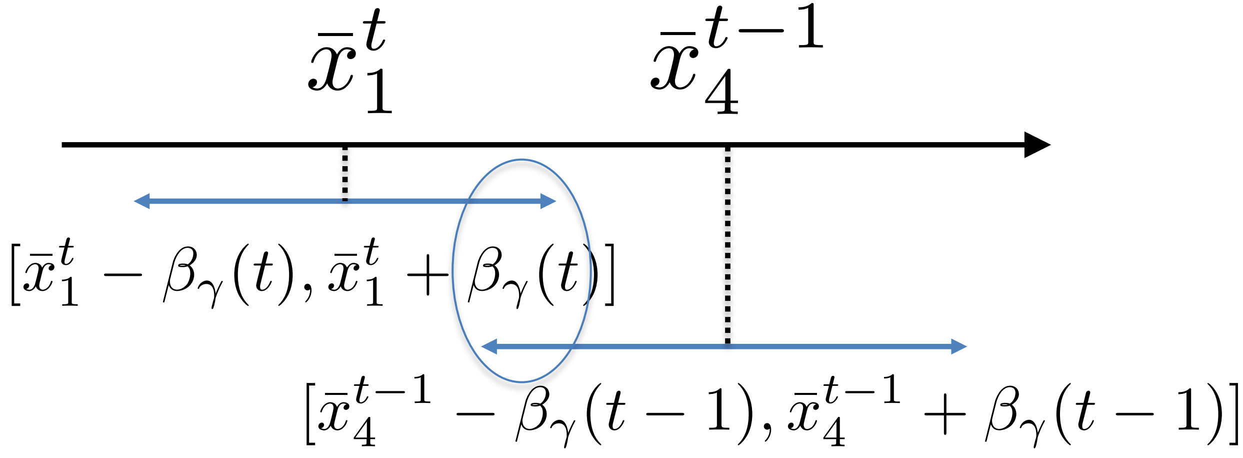
Asadi, Bellet, Maillard, Tommasi, Collaborative Algorithms for Online Personalized Mean Estimation, TMLR, 2023

Collaborative Mean Estimation (Asadi et al, 2023)

Same μ? YES w. confidence 1-γ

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$

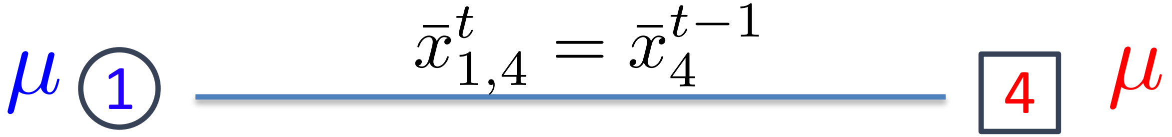


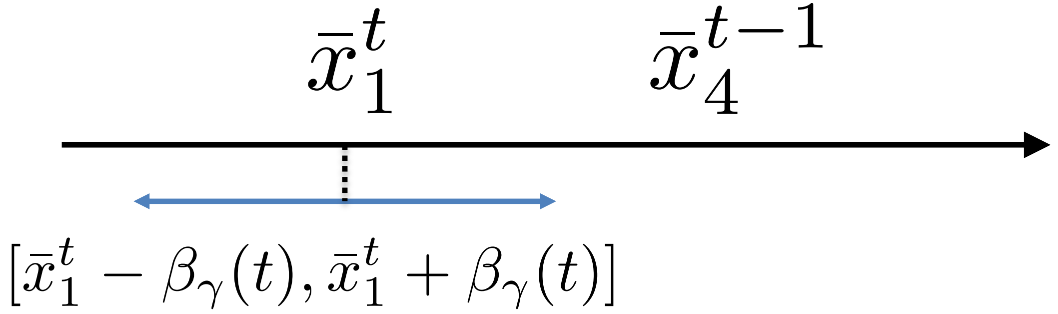


Collaborative Mean Estimation (Asadi et al, 2023)

Same μ? YES w. confidence 1-γ

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$





Collaborative Mean Estimation (Asadi et al, 2023)

Same μ? YES w. confidence 1-γ

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$

μ 1 — $\bar{x}_{1,4}^t = \bar{x}_4^{t-1}$ — 4 μ

$\bar{x}_1^t$ $\bar{x}_4^{t-1}$

$[\bar{x}_1^t - \beta_\gamma(t), \bar{x}_1^t + \beta_\gamma(t)]$

$$\mathbb{P}(\exists t : |\bar{x}_1^t - \mu_1| > \beta_\gamma(t)) \leq \gamma$$

small prob. to *ever* observe a false negative

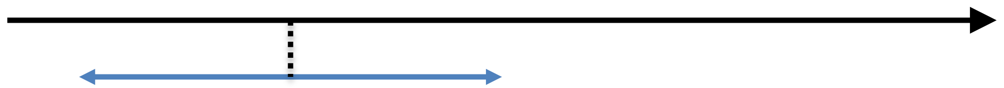
Collaborative Mean Estimation (Asadi et al, 2023)

Same μ? YES w. confidence 1-γ

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$

μ (1) —— $\bar{x}_{1,4}^t = \bar{x}_4^{t-1}$ —— [4] μ

$\bar{x}_1^t$ $\bar{x}_4^{t-1}$



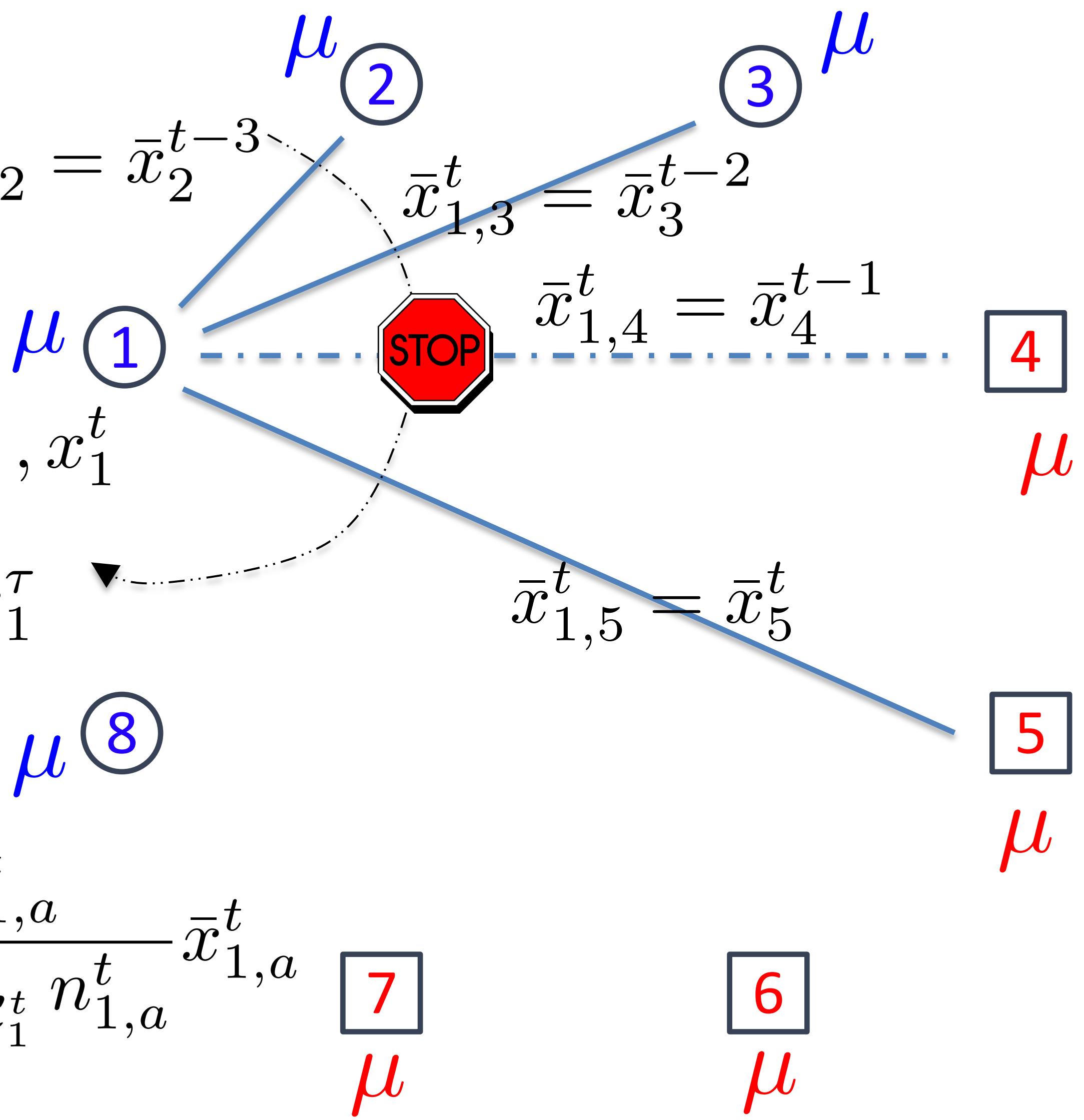
$[\bar{x}_1^t - \beta_\gamma(t), \bar{x}_1^t + \beta_\gamma(t)]$

$$\mathbb{P}(\exists t : |\bar{x}_1^t - \mu_1| > \beta_\gamma(t)) \leq \gamma$$

small prob. to *ever* observe a false negative

When we conclude that 2 distributions are different, there is no need to reconsider the decision

Collaborative Mean Estimation (Asadi et al, 2023)



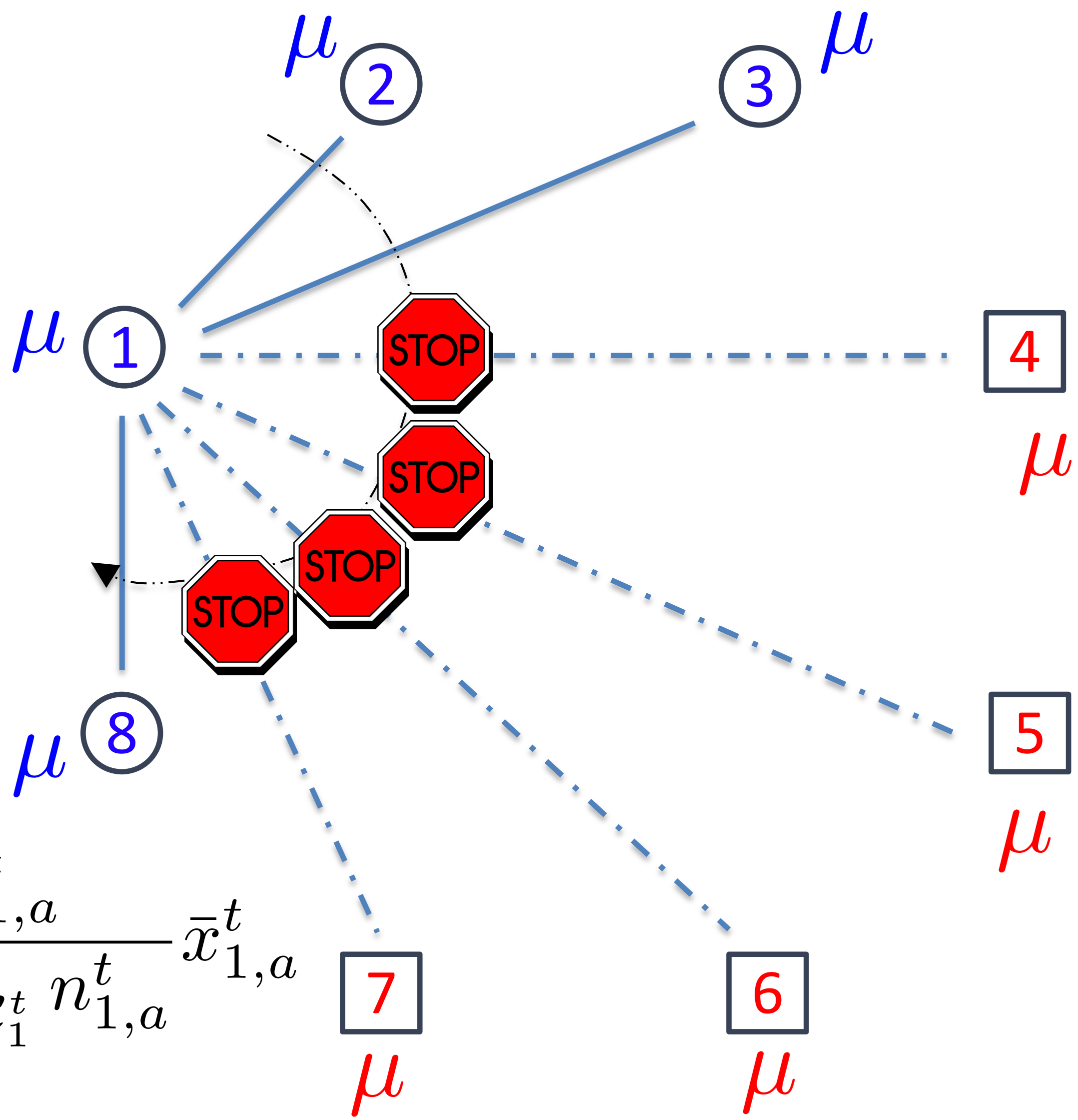
$$x_1^1, x_1^2, \cdots, x_1^t$$

$$\bar{x}_1^t = \frac{1}{t}\sum_{\tau=1}^{t} x_1^\tau$$

$$\hat{\mu}_1^t = \sum_{a \in \mathcal{C}_1^t} \frac{n_{1,a}^t}{\sum_{a \in \mathcal{C}_1^t} n_{1,a}^t} \bar{x}_{1,a}^t$$

- Each agent re-evaluates the set of potentially similar agents C_a^t

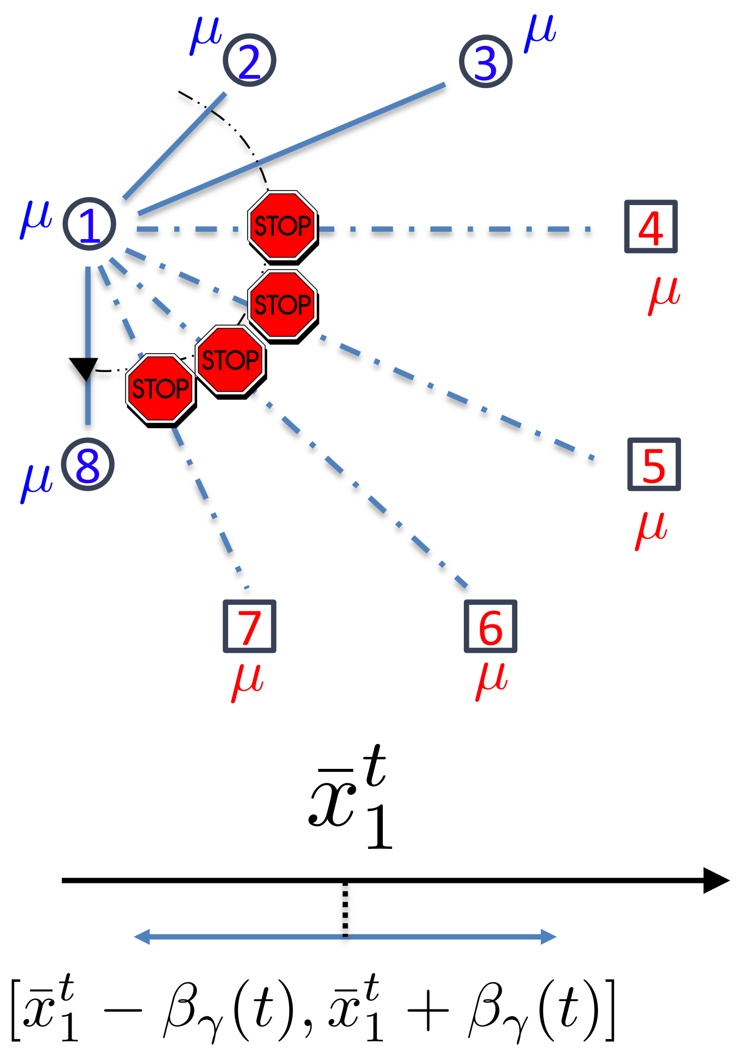
Collaborative Mean Estimation (Asadi et al, 2023)



$$\hat{\mu}_1^t = \sum_{a \in \mathcal{C}_1^t} \frac{n_{1,a}^t}{\sum_{a \in \mathcal{C}_1^t} n_{1,a}^t} \bar{x}_{1,a}^t$$

- Each agent re-evaluates the set of potentially similar agents C_a^t

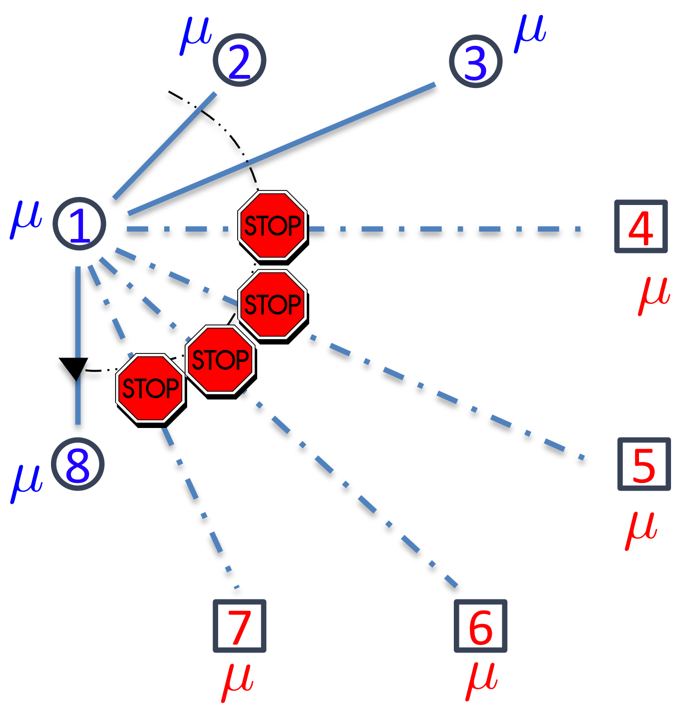
Collaborative Mean Estimation (Asadi et al, 2023)



If distributions are sub-gaussians w. parameter σ^2

- $\beta_\gamma(n) := \sigma\sqrt{\frac{2}{n}\left(1+\frac{1}{n}\right)\ln(\sqrt{(n+1)}/\gamma)}$

Collaborative Mean Estimation (Asadi et al, 2023)



If distributions are sub-gaussians w. parameter σ^2

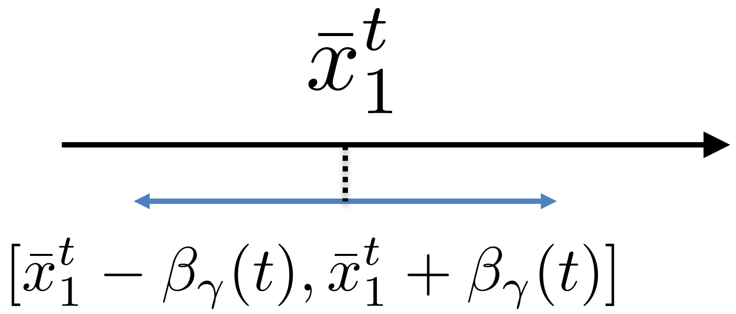
- $\beta_\gamma(n) := \sigma\sqrt{\frac{2}{n}\left(1+\frac{1}{n}\right)\ln(\sqrt{(n+1)}/\gamma)}$

$$\mathbb{P}(a \text{ keeps wrong collaborators after } \zeta_a) \leq \frac{\delta}{2}$$

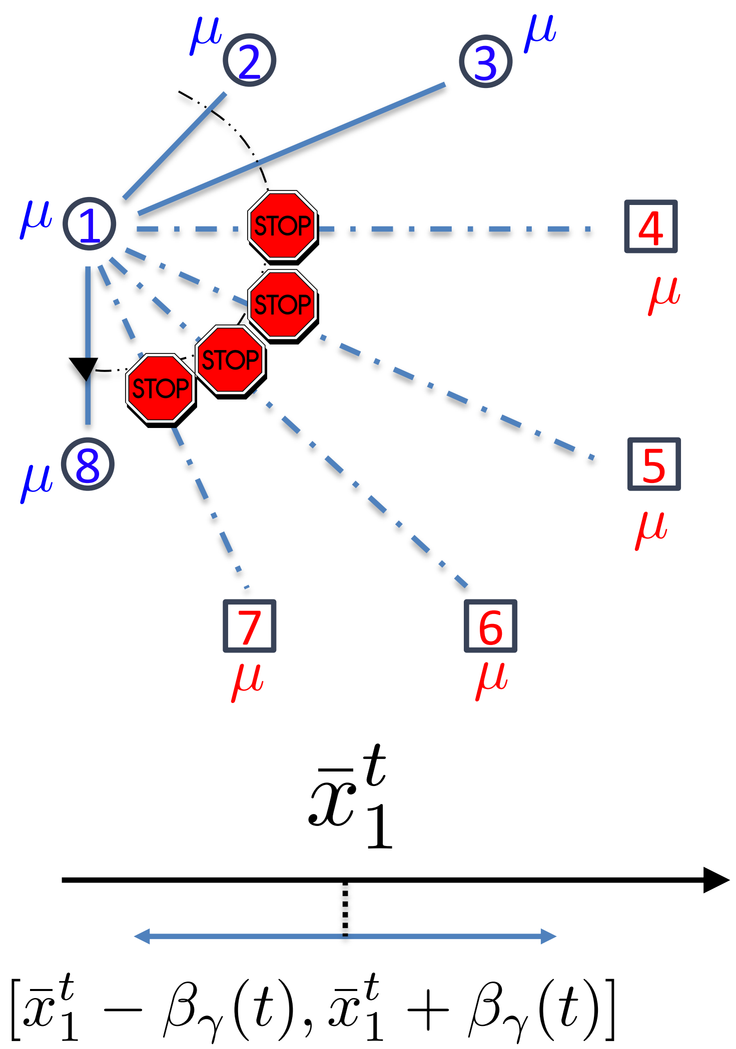
$$\zeta_a \approx n^\star_{\frac{\delta}{2|\mathcal{A}|}}(\Delta\mu_a) + |\mathcal{A}|$$

$$n^\star_\gamma(\Delta) := \beta_\gamma^{-1}(\Delta)$$

minimum #samples to distinguish difference **Δ**



Collaborative Mean Estimation (Asadi et al, 2023)

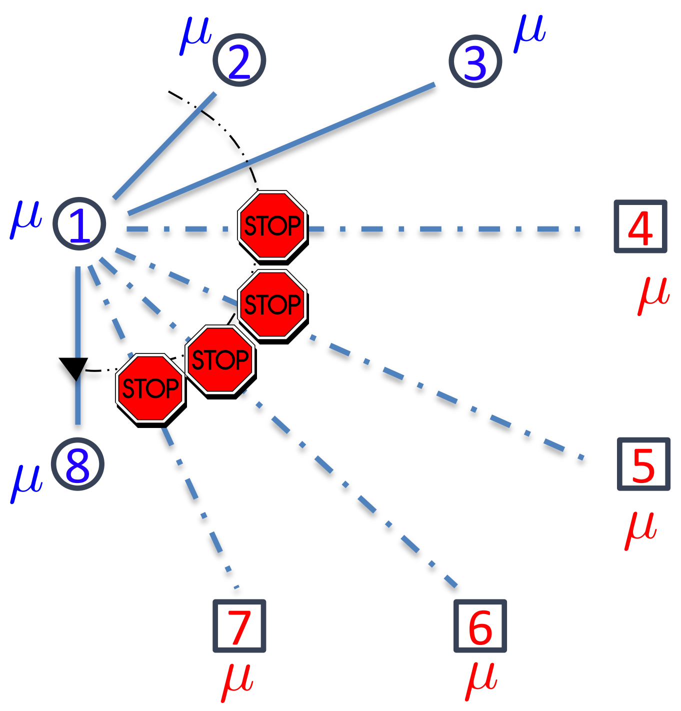


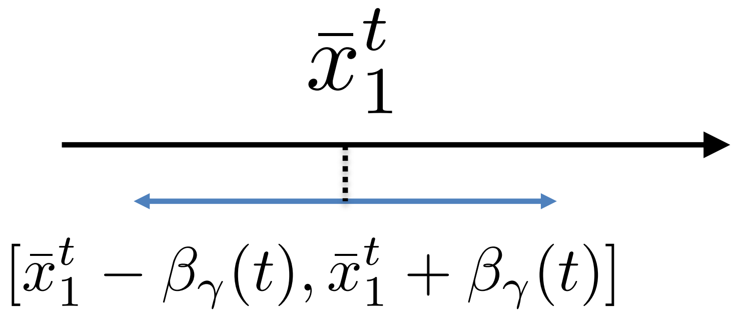
If the right collaborators C_a have been identified...

$$\mathbb{P}(\exists t > \zeta'_a : |\hat{\mu}_a^t - \mu_a| > \varepsilon) \leq \frac{\delta}{2}$$

$$\zeta'_a \approx \frac{n^\star_{\delta/2}(\epsilon)}{|\mathcal{C}_a|} + \frac{|\mathcal{C}_a|}{2}$$

Collaborative Mean Estimation (Asadi et al, 2023)





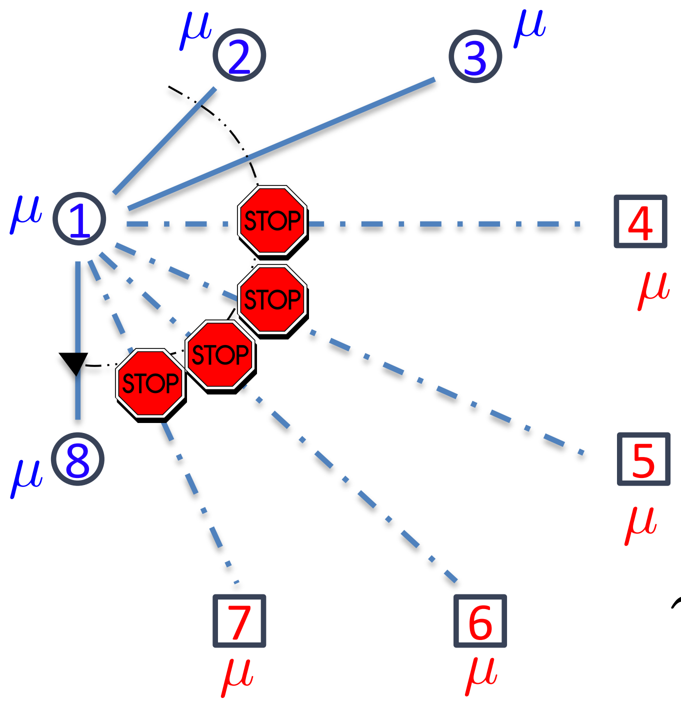
If the right collaborators C_a have been identified...

$$\mathbb{P}(\exists t > \zeta'_a : |\hat{\mu}_a^t - \mu_a| > \varepsilon) \leq \frac{\delta}{2}$$

$$\zeta'_a \approx \frac{n^\star_{\delta/2}(\epsilon)}{|\mathcal{C}_a|} + \frac{|\mathcal{C}_a|}{2}$$

cooperation speedup

Collaborative Mean Estimation (Asadi et al, 2023)



Putting all together

$$\mathbb{P}(\exists t > \tau_a : |\hat{\mu}_a^t - \mu_a| > \varepsilon) \leq \delta$$

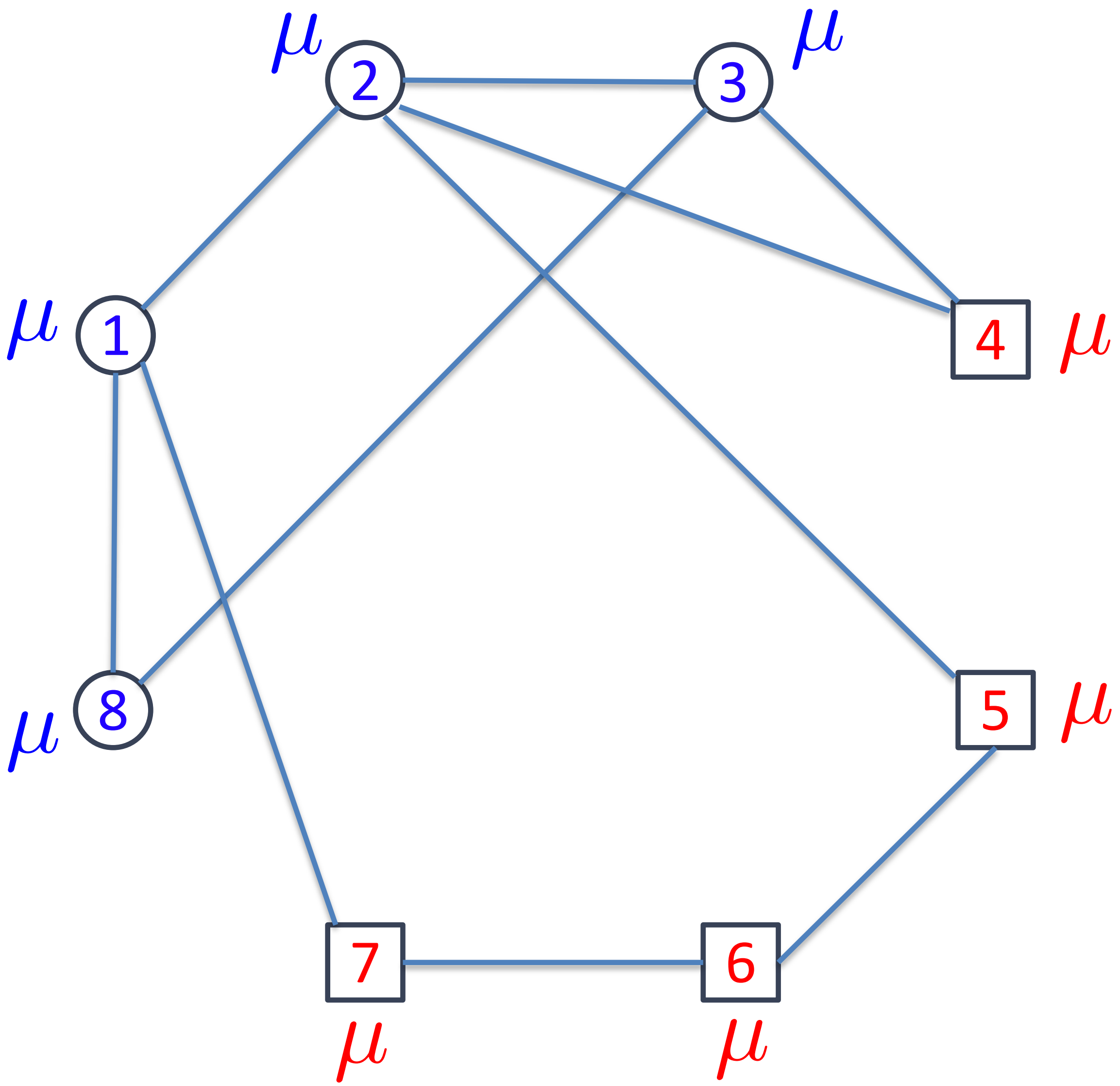
$$\tau_a \approx \max \left\{ n^\star_{\frac{\delta}{2|\mathcal{A}|}}(\Delta\mu_a) + |\mathcal{A}|, \frac{\tilde{n}^\star_{\delta/2}(\epsilon)}{|\mathcal{C}_a|} + \frac{|\mathcal{C}_a|}{2} \right\}$$

$$\tau_a \in \tilde{\mathcal{O}}\left(\frac{\log \frac{|\mathcal{A}|}{\delta}}{\Delta\mu_a^2} + |\mathcal{A}| + \frac{\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}}{|\mathcal{C}_a|} \right)$$

Summary

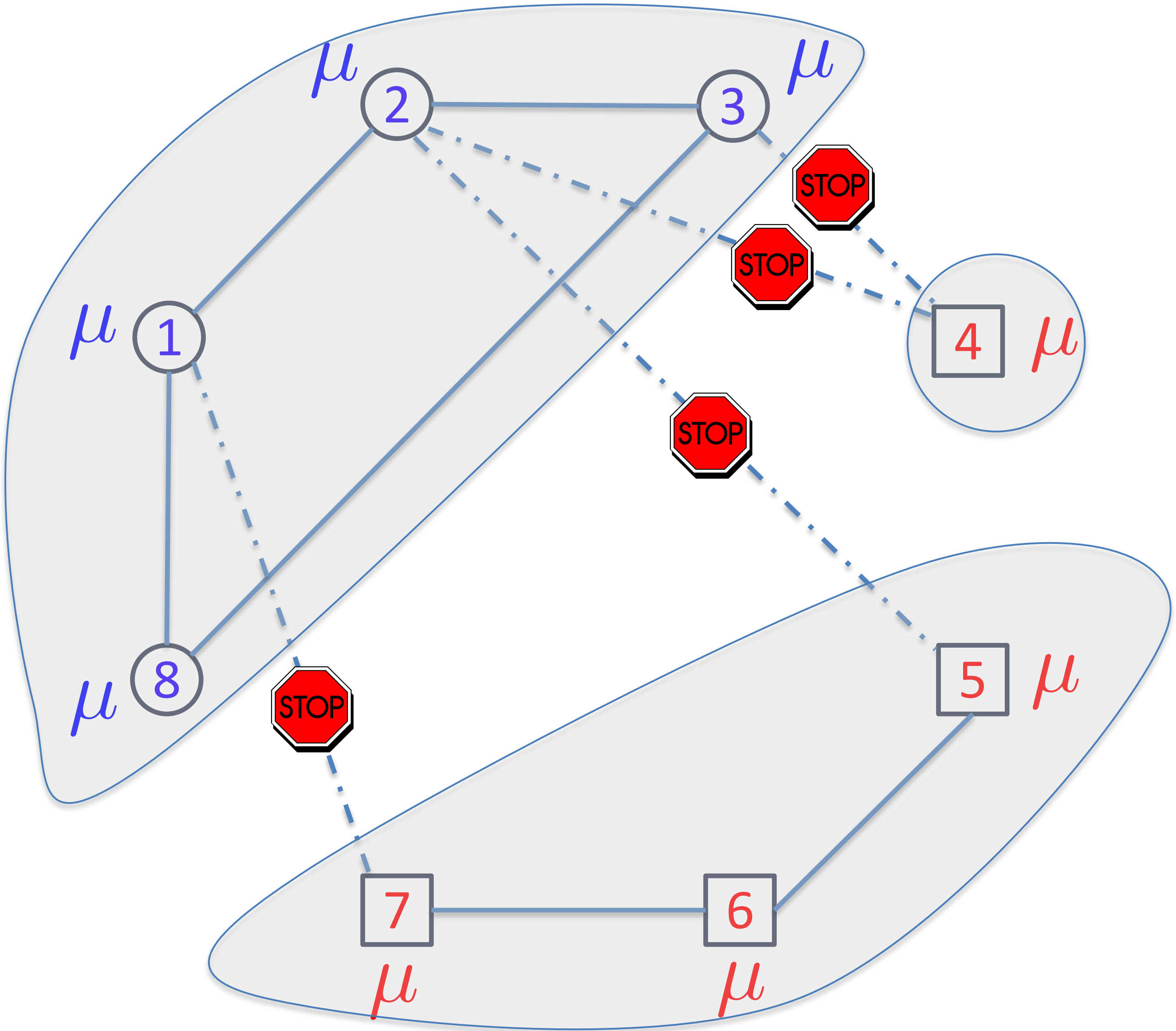
	Per-agent space/time complexity	Convergence time	
		sub-Gaussian	bounded 4-th moment
ColME	$\lvert\mathcal{A}\rvert$	$\frac{1}{\Delta\mu_a^2}\log\frac{\lvert\mathcal{A}\rvert}{\Delta\mu_a\delta} + \frac{\lvert\mathcal{A}\rvert}{r} + \frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}$	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{A}\rvert}{\delta} + \frac{\lvert\mathcal{A}\rvert}{r} + \frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\delta\varepsilon^4}$

What if communication over a graph?



- Maximum degree r
- Each agent can communicate in parallel with its neighbors

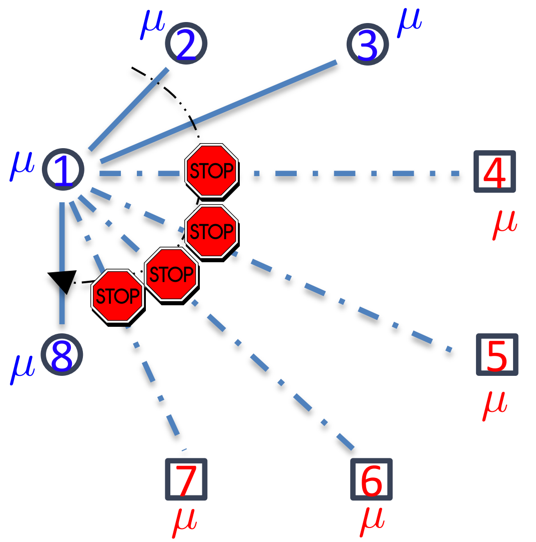
What if communication over a graph?

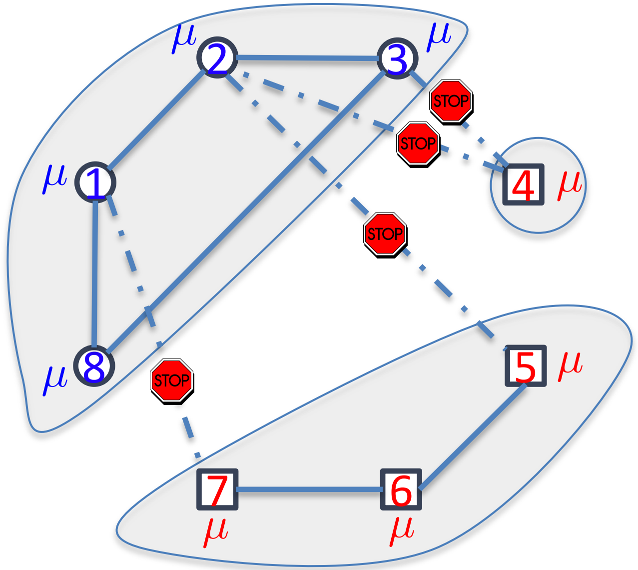


Expected tradeoff:

- A sparser graph may be learned faster
- But connected components may be smaller reducing collaboration speedup

Learning the right collaborators





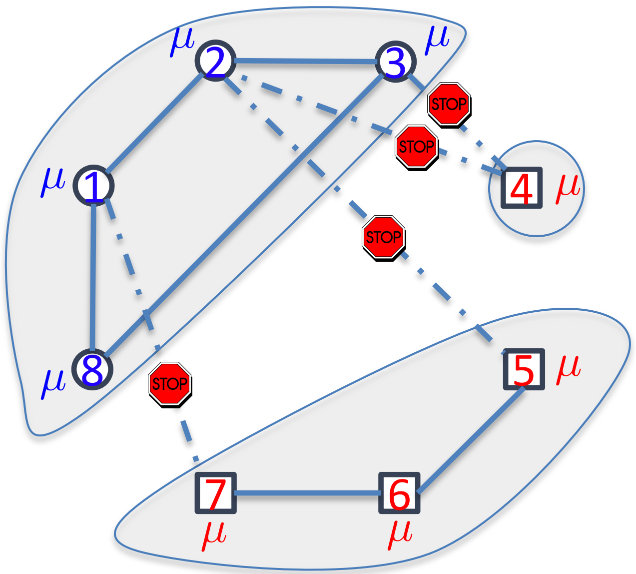
$$\mathbb{P}(a \text{ keeps wrong collaborators after } \zeta_a) \leq \frac{\delta}{2}$$

$$\zeta_a \approx n^{\star}_{\frac{\delta}{2|\mathcal{A}|}}(\Delta\mu_a) + \frac{|\mathcal{A}|}{r}$$

$$\mathbb{P}(\text{any wrong collaboration after } \zeta_D) \leq \frac{\delta}{2}$$

$$\zeta_D \approx n^{\star}_{\frac{\delta}{2|\mathcal{C}_a|r}}(\Delta\mu_a)$$

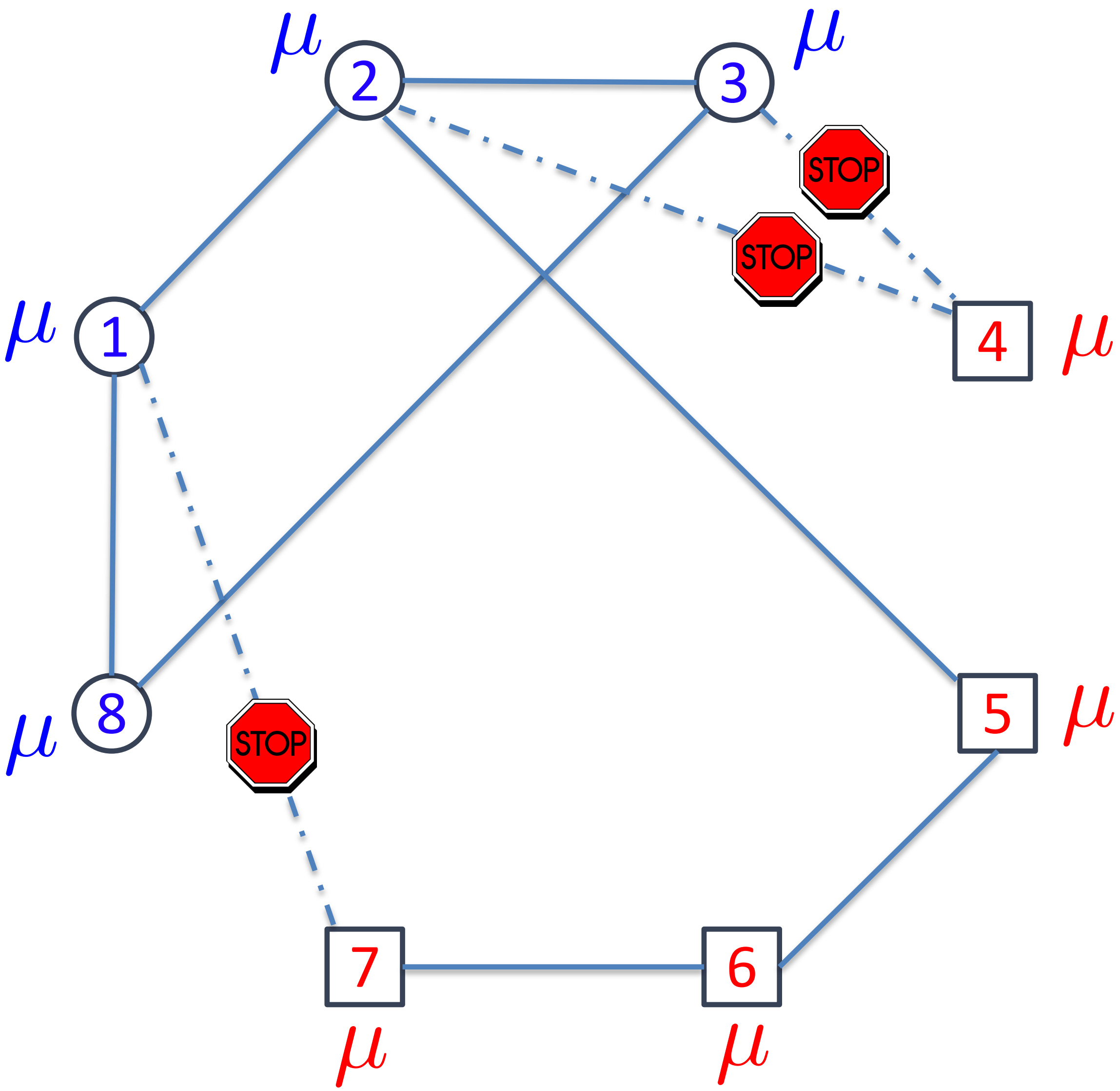
How to estimate over the graph



Two algorithms:

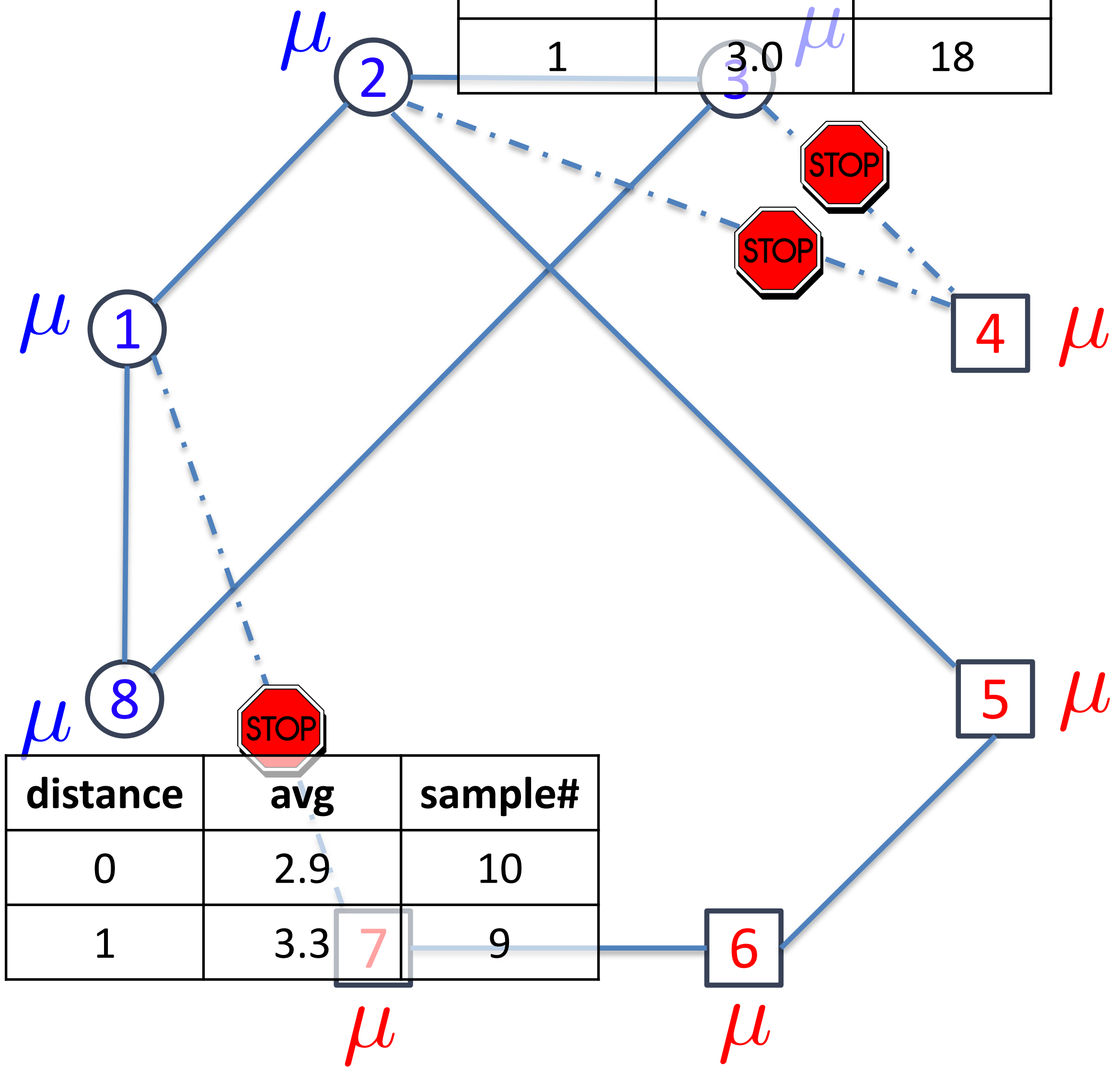
1. B-ColME, based on message passing as a belief algorithm
2. C-ColME, based on consensus

B-ColME



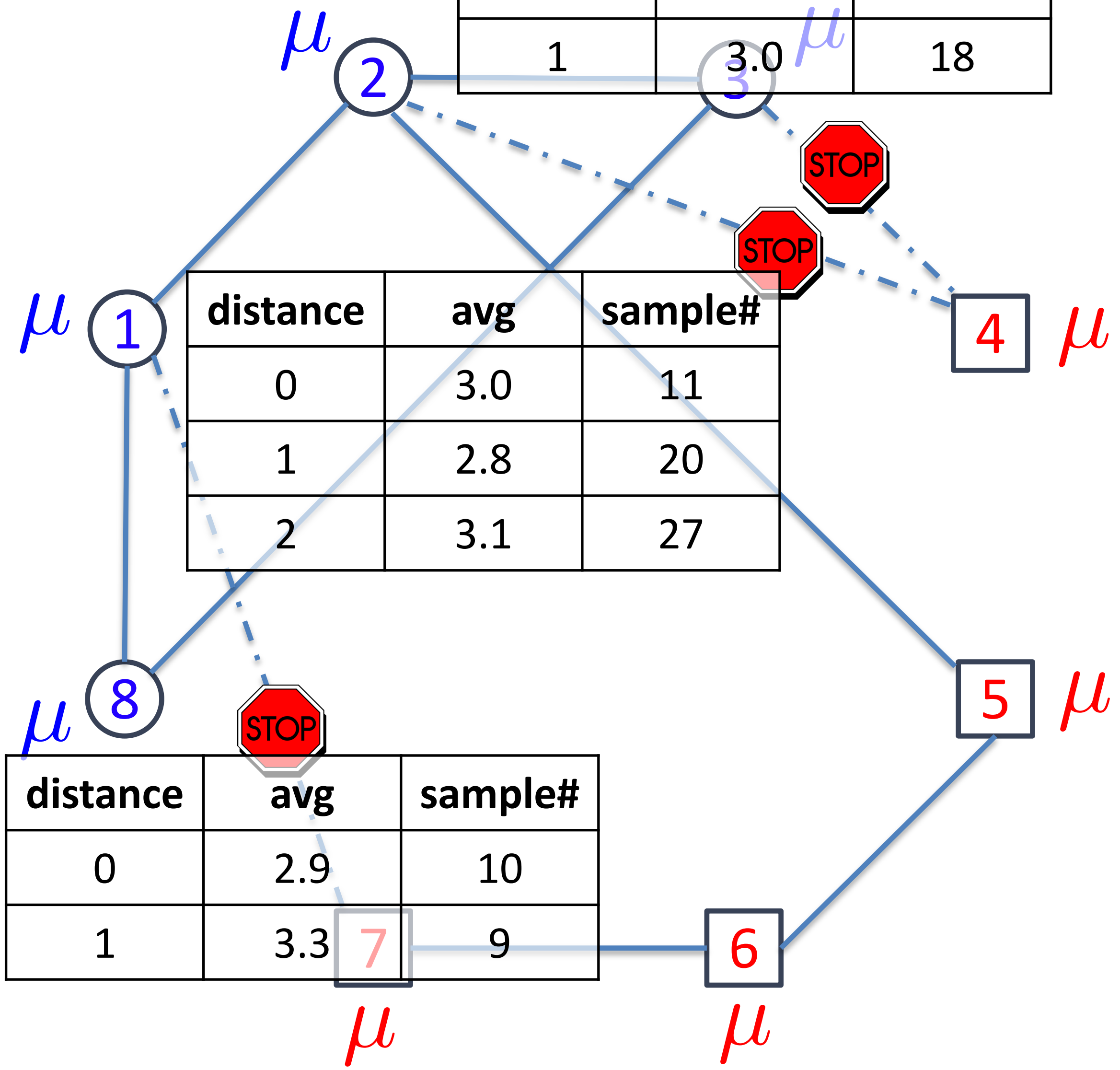
B-ColME

distance	avg	sample#
0	2.7	10
1	3.0	18



distance	avg	sample#
0	2.9	10
1	3.3	9

B-ColME



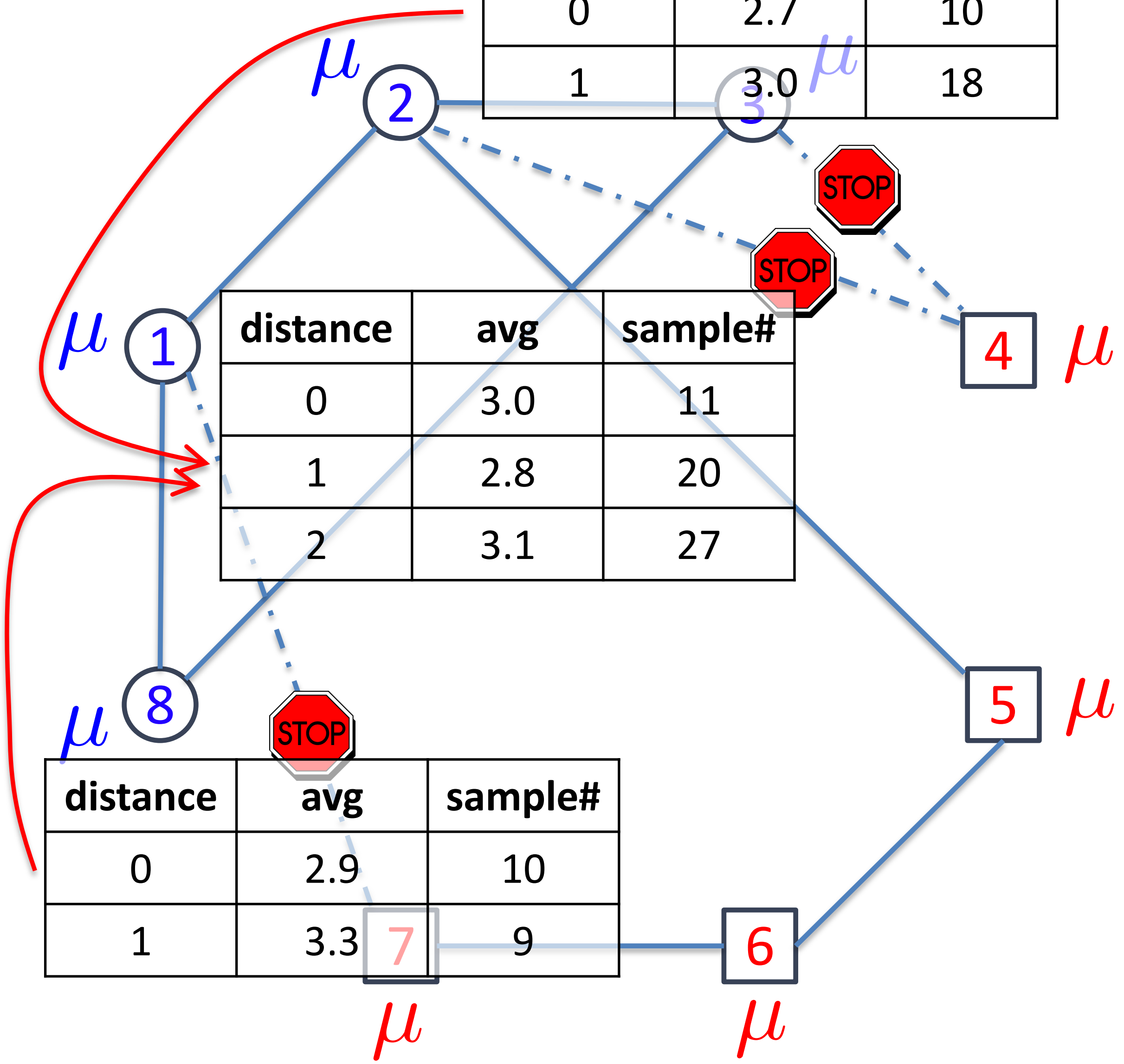
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
2	3.1	27

distance	avg	sample#
0	2.9	10
1	3.3	9

B-ColME

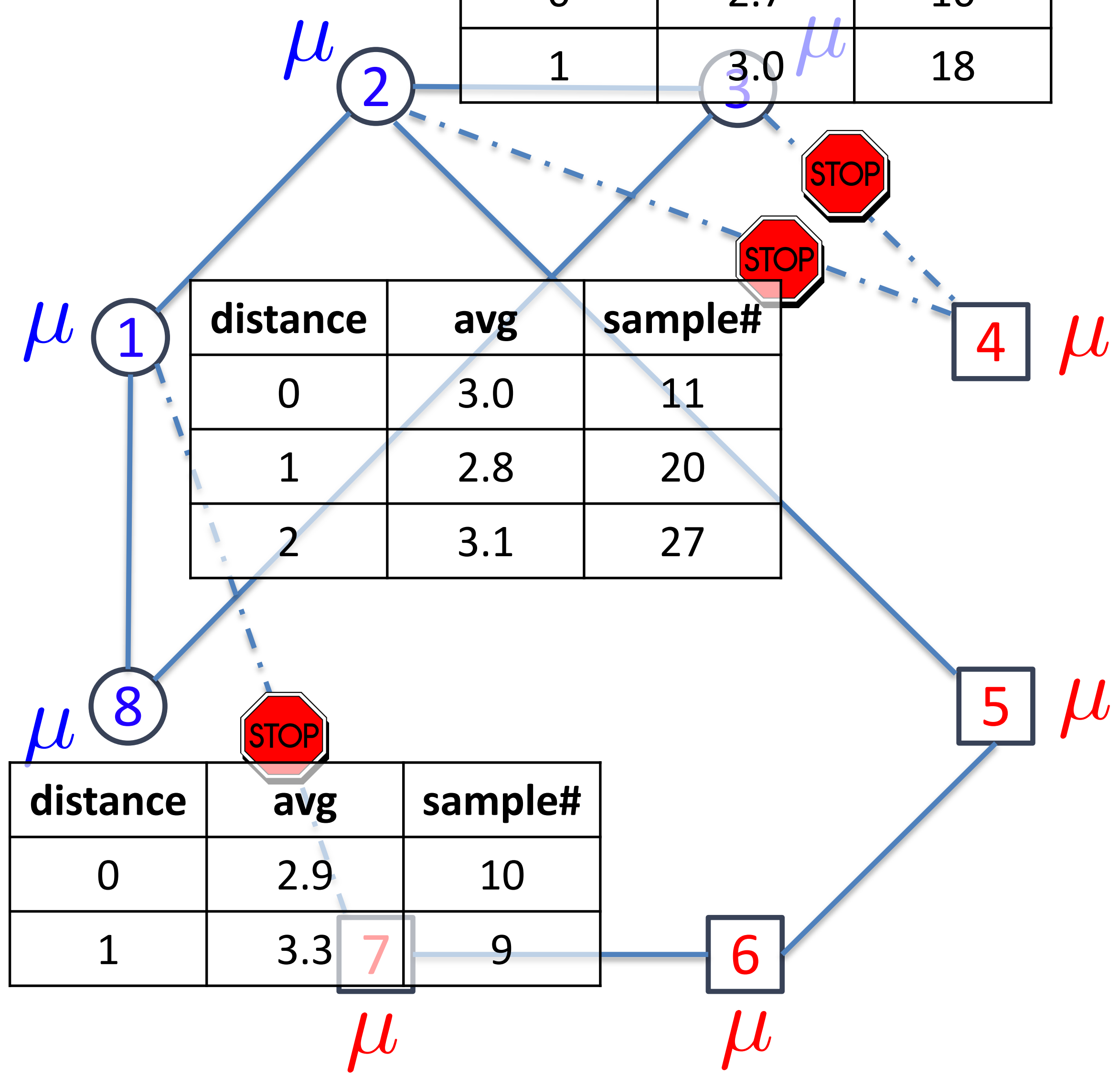
distance	avg	sample#
0	2.7	10
1	3.0	18



distance	avg	sample#
0	3.0	11
1	2.8	20
2	3.1	27

distance	avg	sample#
0	2.9	10
1	3.3	9

B-ColME



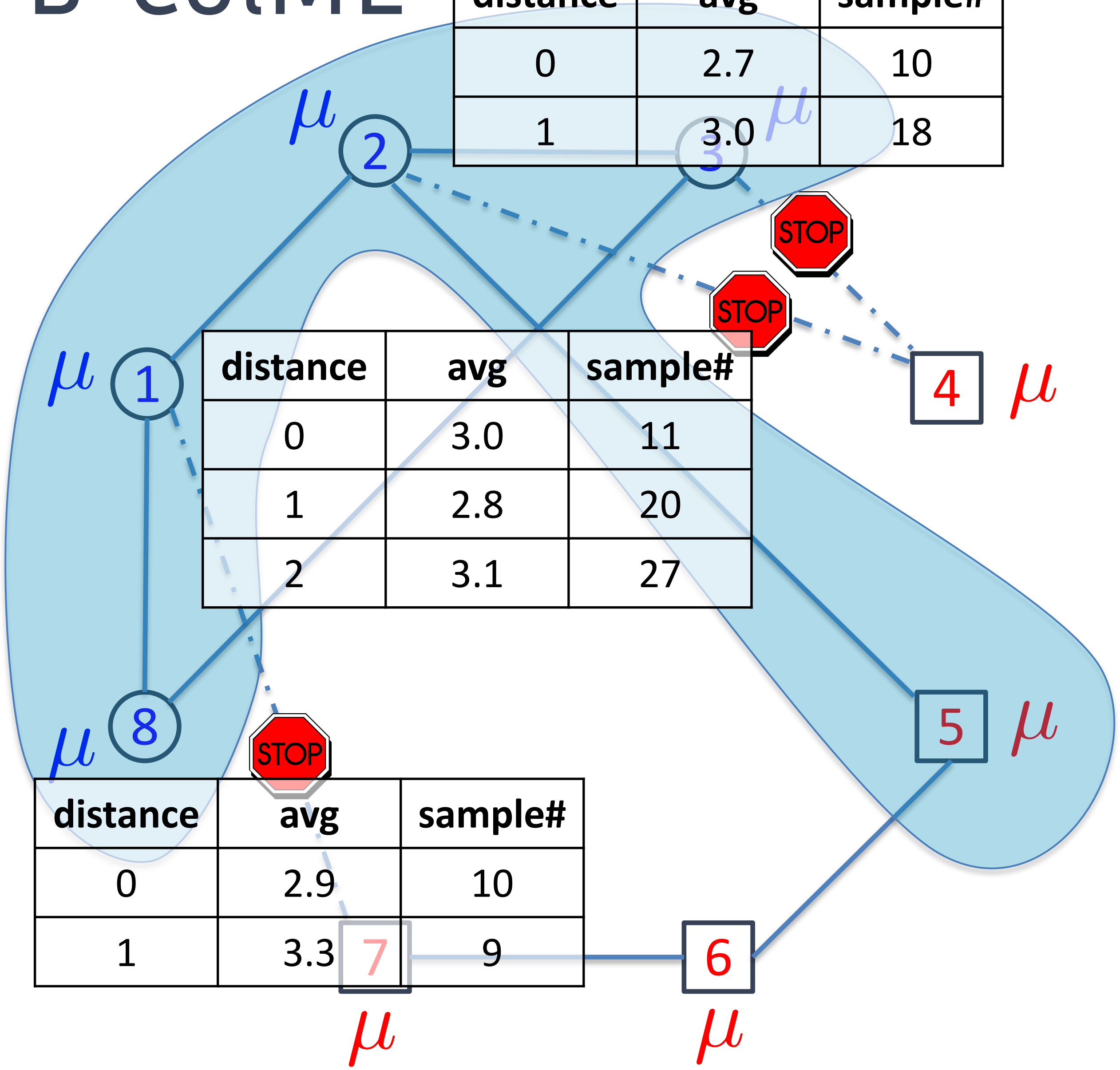
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
2	3.1	27

distance	avg	sample#
0	2.9	10
1	3.3	9

- Each agent estimates the empirical average over a h-hop neighborhood using estimates over (h-1)-hop neighborhoods of its direct neighbors

B-ColME



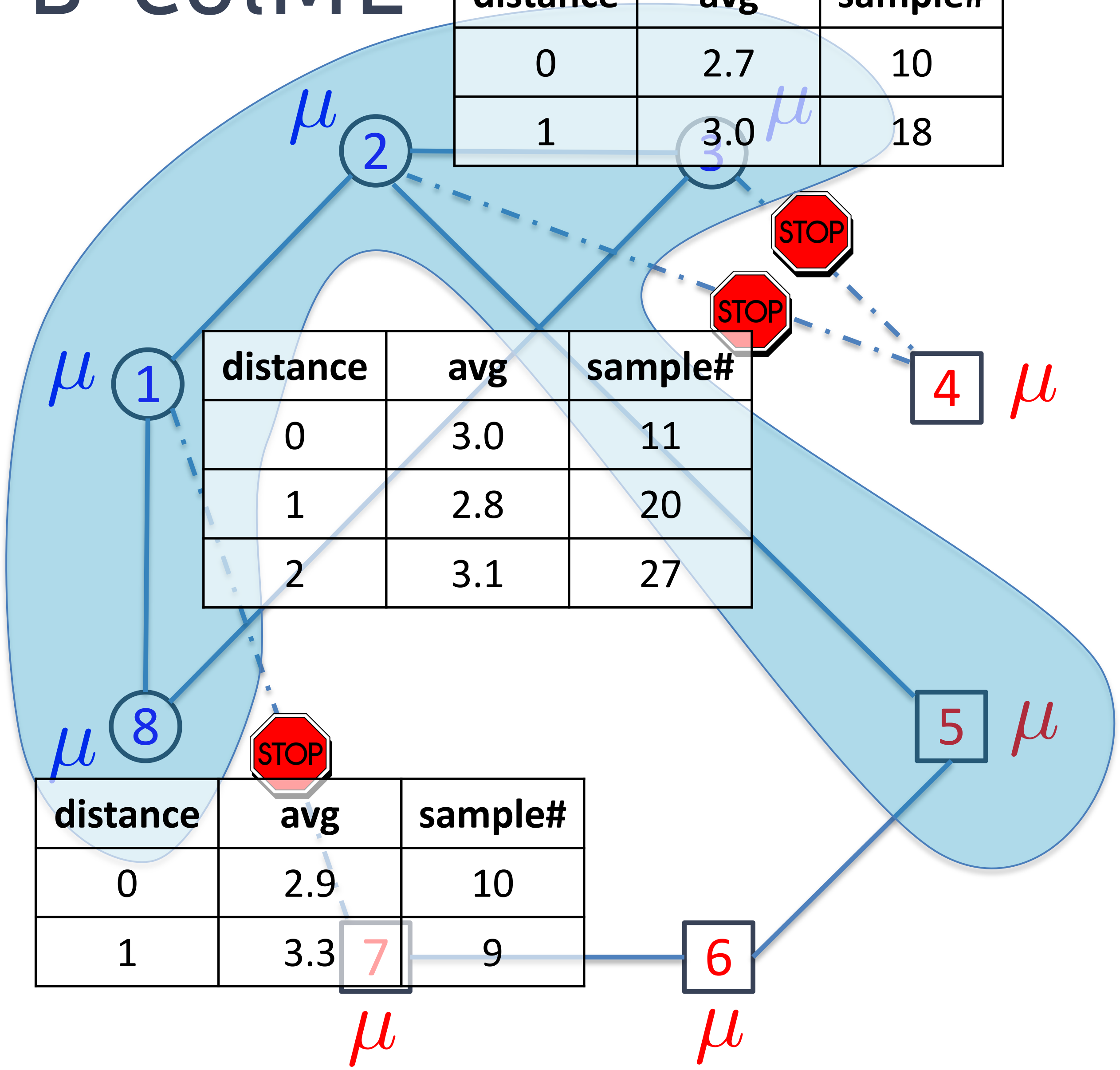
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
2	3.1	27

distance	avg	sample#
0	2.9	10
1	3.3	9

- Each agent estimates the empirical average over a h-hop neighborhood using estimates over (h-1)-hop neighborhoods of its direct neighbors

B-ColME



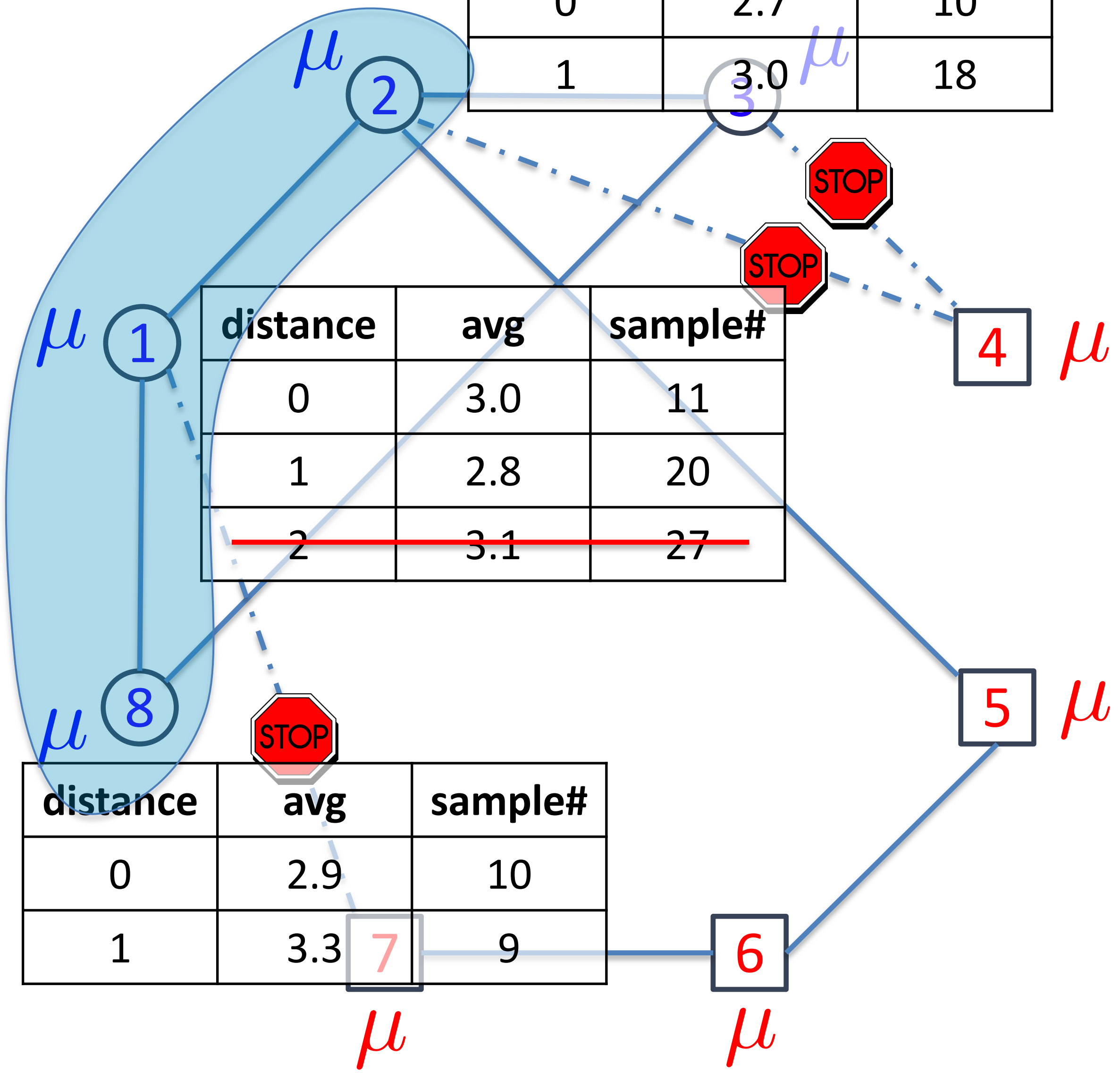
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
2	3.1	27

distance	avg	sample#
0	2.9	10
1	3.3	9

- Each agent estimates the empirical average over a h-hop neighborhood using estimates over (h-1)-hop neighborhoods of its direct neighbors
- Problem with loops
  - ⇒ restrain over a distance d s.t. the d-hop neighborhood is a tree

B-ColME



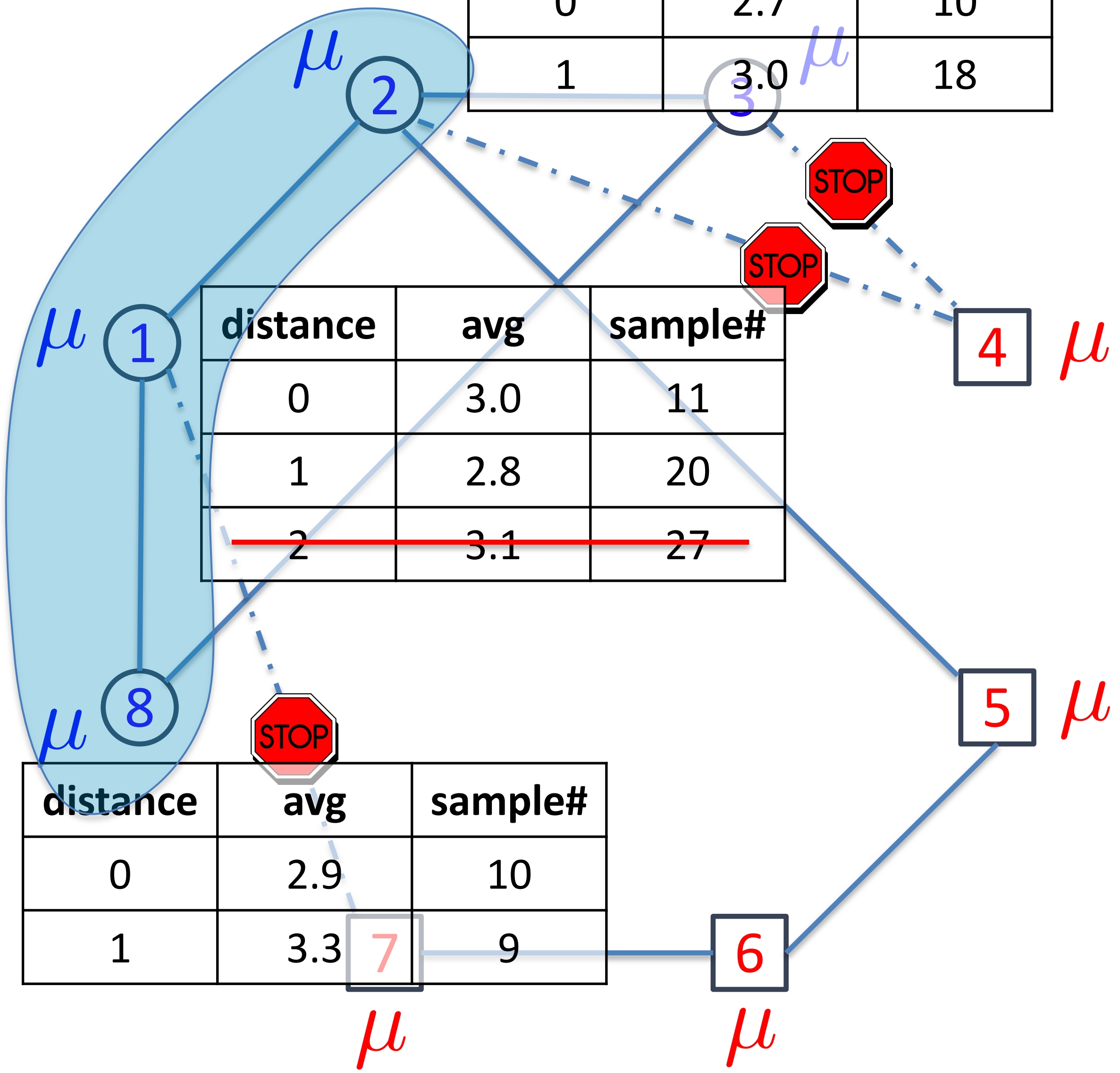
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
~~2~~	~~3.1~~	~~27~~

distance	avg	sample#
0	2.9	10
1	3.3	9

- Each agent estimates the empirical average over a h-hop neighborhood using estimates over (h-1)-hop neighborhoods of its direct neighbors
- Problem with loops
  - ⇒ restrain over a distance d s.t. the d-hop neighborhood is a tree

B-ColME



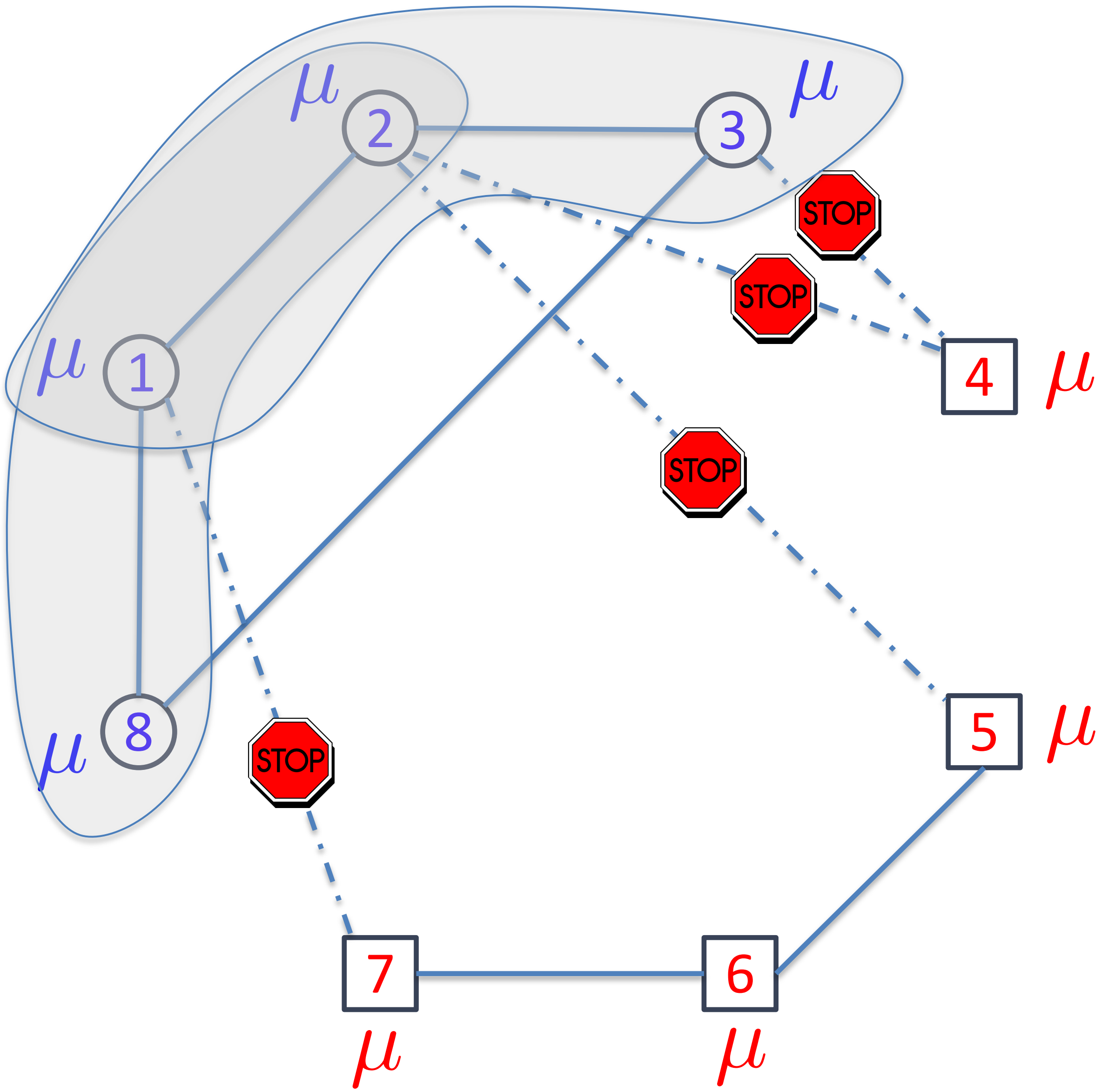
distance	avg	sample#
0	2.7	10
1	3.0	18

distance	avg	sample#
0	3.0	11
1	2.8	20
~~2~~	~~3.1~~	~~27~~

distance	avg	sample#
0	2.9	10
1	3.3	9

- Each agent estimates the empirical average over a h-hop neighborhood using estimates over (h-1)-hop neighborhoods of its direct neighbors
- Problem with loops
  - ⇒ restrain over a distance d s.t. the d-hop neighborhood is a tree
- Each stores and sends tables with d entries

B-ColME



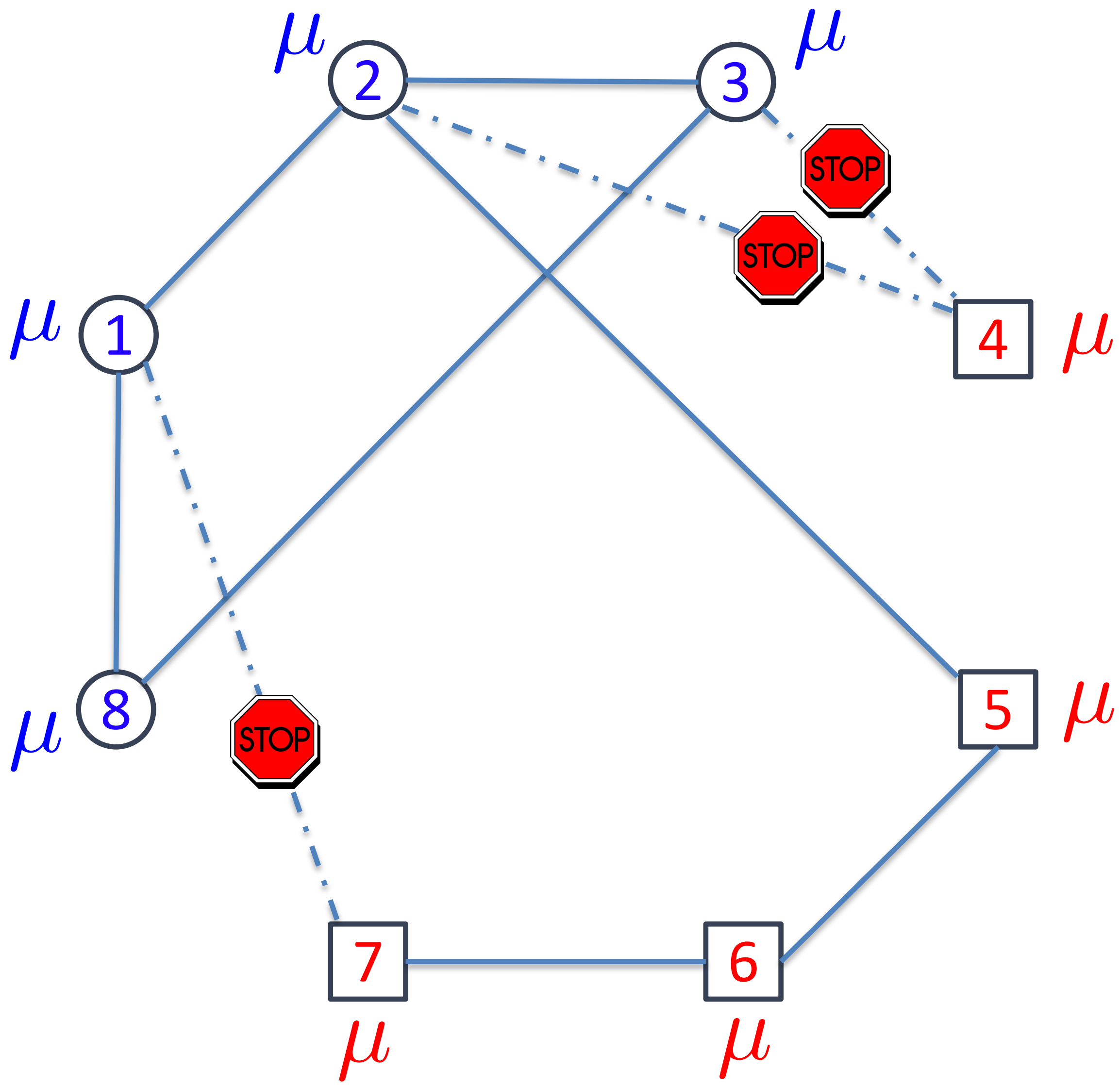
- Nodes in the same connected component will not compute estimates over the same d-hop neighborhood CC_a^d

- Convergence of the estimator is evident

Summary

	Per-agent space/time complexity	Convergence time	
		sub-Gaussian	bounded 4-th moment
ColME	$\lvert\mathcal{A}\rvert$	$\frac{1}{\Delta\mu_a^2}\log\frac{\lvert\mathcal{A}\rvert}{\Delta\mu_a\delta}+\frac{\lvert\mathcal{A}\rvert}{r}+\frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}$	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{A}\rvert}{\delta}+\frac{\lvert\mathcal{A}\rvert}{r}+\frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\delta\varepsilon^4}$
B-ColME	rd	$\frac{1}{\Delta\mu_a^2}\log\frac{\lvert\mathcal{C}\mathcal{C}_a\rvert r}{\Delta\mu_a\delta}+d+\frac{1}{\lvert\mathcal{C}\mathcal{C}_a^d\rvert}\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}$	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{C}\mathcal{C}_a\rvert r}{\delta}+d+\frac{1}{\lvert\mathcal{C}\mathcal{C}_a^d\rvert}\frac{1}{\delta\varepsilon^4}$

C-ColME

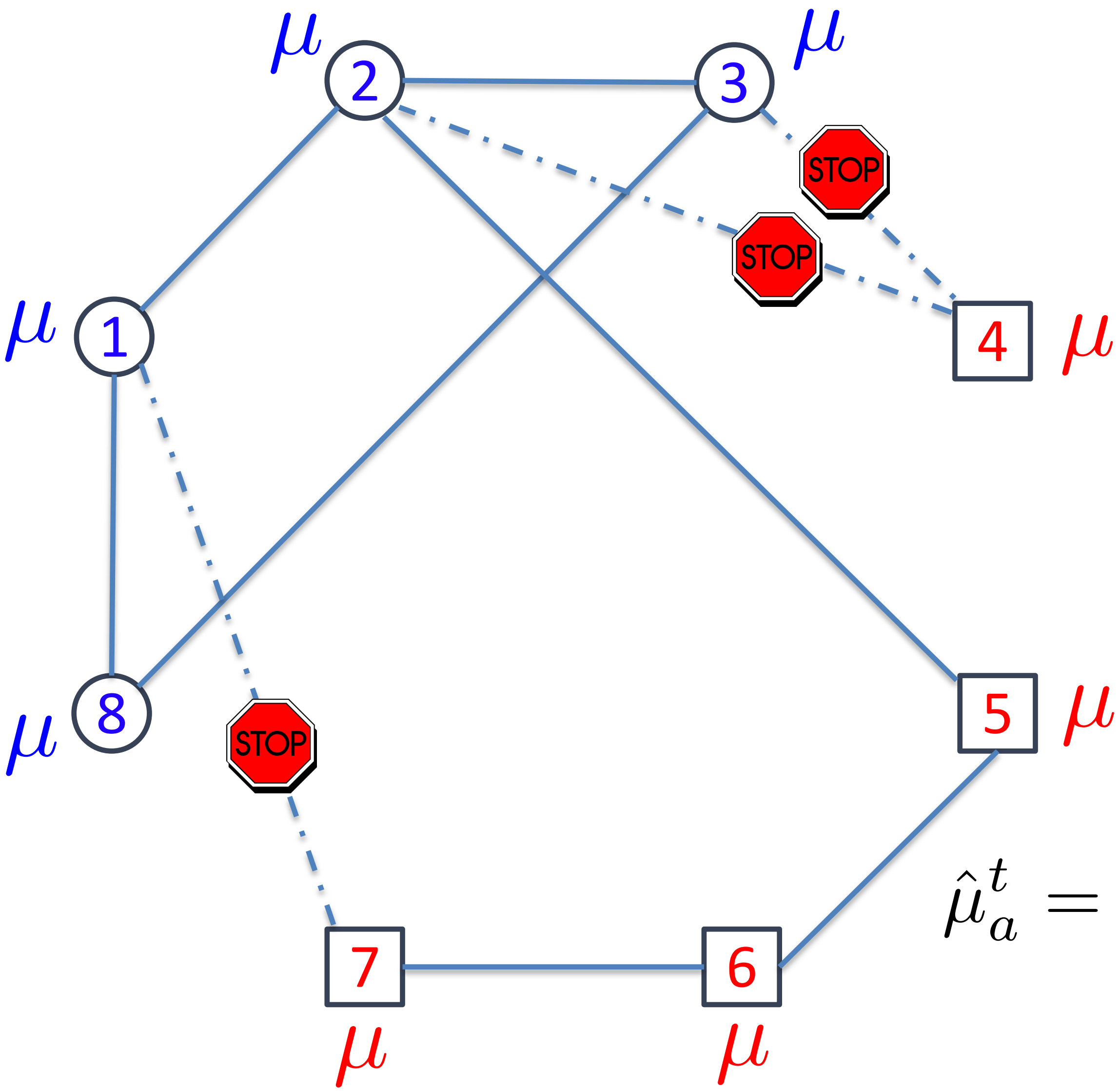


- Standard average consensus

$$\hat{\mu}_a^1 = x_a$$

$$\hat{\mu}_a^t = \sum_{a' \in \mathcal{C}_a^t \cup \{a\}} (W_t)_{a,a'} \hat{\mu}_{a'}^{t-1}$$

C-ColME



- Standard average consensus

$$\hat{\mu}_a^1 = x_a$$

$$\hat{\mu}_a^t = \sum_{a' \in \mathcal{C}_a^t \cup \{a\}} (W_t)_{a,a'} \hat{\mu}_{a'}^{t-1}$$

- Here need to track a moving average over a dynamic graph

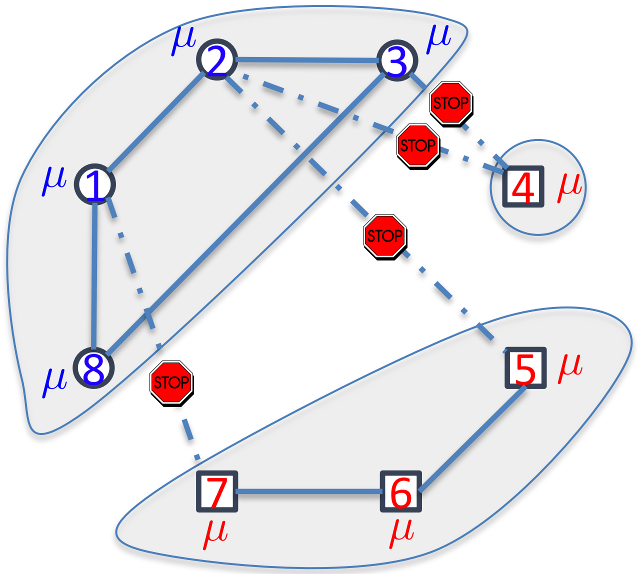
$$\hat{\mu}_a^1 = x_a$$

$$\hat{\mu}_a^t = (1-\alpha_t)\bar{x}_a^t + \alpha_t \sum_{a' \in \mathcal{C}_a^t \cup \{a\}} (W_t)_{a,a'} \hat{\mu}_{a'}^{t-1}$$

C-ColME

Convergence to true mean:

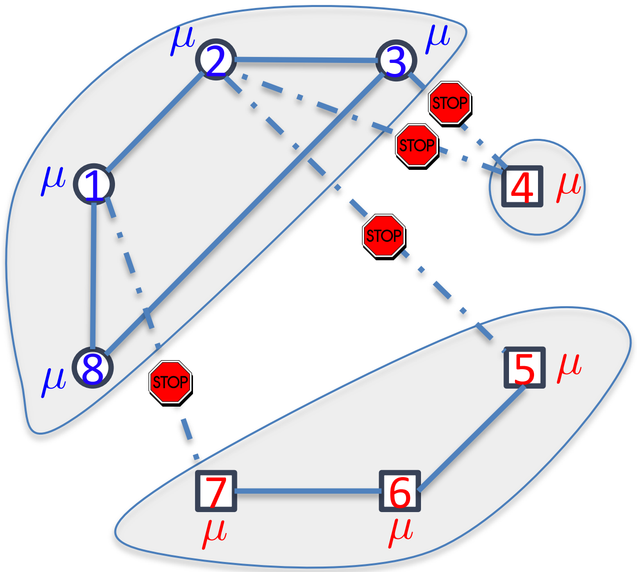
- W_t stochastic & symmetric
- $\alpha_t = \alpha$ or $\alpha_t = \dfrac{t}{t+1}$



C-ColME

Convergence to true mean:

- W_t stochastic & symmetric
- $\alpha_t = \alpha$ or $\alpha_t = \dfrac{t}{t+1}$



$$\mathbb{E}\left[\|{}_c\hat{\boldsymbol{\mu}}^{t+1} - {}_c\boldsymbol{\mu}\|^4\right] \in \mathcal{O}\left(\sup_{W_1,\cdots,W_{\zeta_D}} \frac{\mathbb{E}\left[\|{}_c\hat{\boldsymbol{\mu}}^{\zeta_D} - {}_c\boldsymbol{\mu}\|^4\right]}{(t+1)^4}\right) + \mathcal{O}\left(\frac{(1-1/\ln\lambda_{2,c})^2}{(1-\lambda_{2,c})^2}\frac{\mathbb{E}\left[\|{}_c\mathbf{x} - {}_cP\,{}_c\mathbf{x}\|^4\right]}{(t+1)^4}\right) + \mathcal{O}\left(\mathbb{E}\left[\|{}_cP\,{}_c\mathbf{x} - {}_c\boldsymbol{\mu}\|^4\right]\left(\frac{1+\ln t}{1+t}\right)^2\right)$$

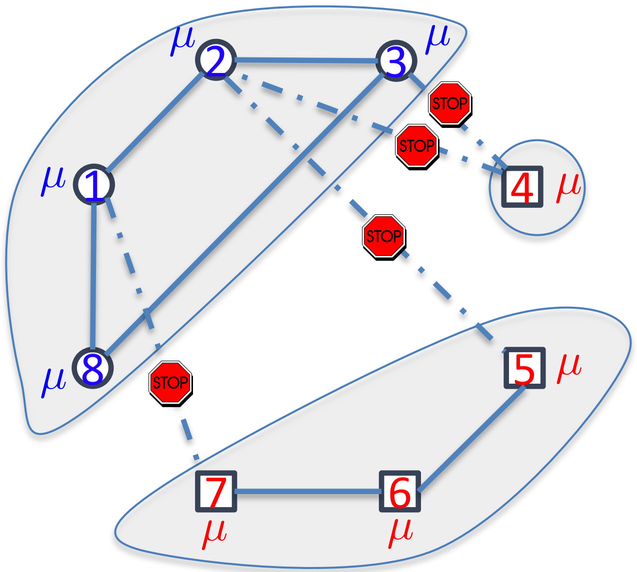
Summary

	Per-agent space/time complexity	Convergence time	
		sub-Gaussian	bounded 4-th moment
ColME	$\lvert\mathcal{A}\rvert$	$\frac{1}{\Delta\mu_a^2}\log\frac{\lvert\mathcal{A}\rvert}{\Delta\mu_a\delta}+\frac{\lvert\mathcal{A}\rvert}{r}+\frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}$	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{A}\rvert}{\delta}+\frac{\lvert\mathcal{A}\rvert}{r}+\frac{1}{\lvert\mathcal{C}_a\rvert}\frac{1}{\delta\varepsilon^4}$
B-ColME	rd	$\frac{1}{\Delta\mu_a^2}\log\frac{\lvert\mathcal{CC}_a\rvert r}{\Delta\mu_a\delta}+d+\frac{1}{\lvert\mathcal{CC}_a^d\rvert}\frac{1}{\varepsilon^2}\log\frac{1}{\delta\varepsilon^2}$	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{CC}_a\rvert r}{\delta}+d+\frac{1}{\lvert\mathcal{CC}_a^d\rvert}\frac{1}{\delta\varepsilon^4}$
C-ColME	r	–	$\frac{1}{\Delta\mu_a^4}\frac{\lvert\mathcal{CC}_a\rvert r}{\delta}+\frac{1}{\lvert\mathcal{CC}_a\rvert}\frac{1}{\delta\varepsilon^4}$

Choice of the graph and other parameters

Desiderata

- Large components CC_a and CC_a^d
- r small
- uniform load over the clients
- the largest d which guarantees the local tree structure

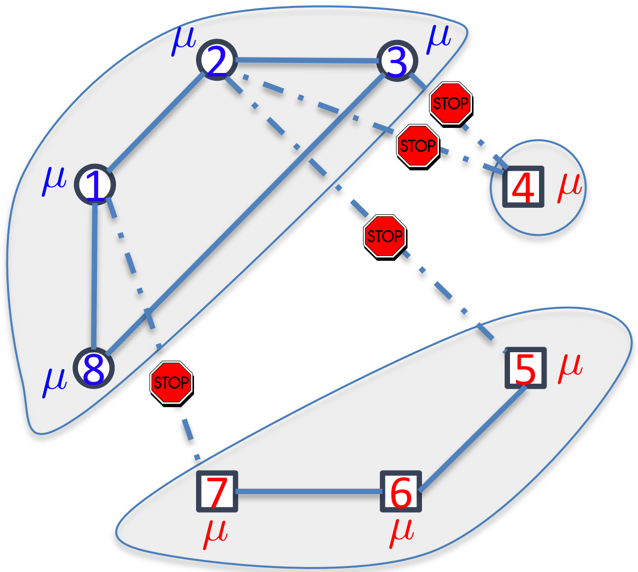


Choice of the graph and other parameters

Desiderata

- Large components CC_a and CC_a^d
- r small
- uniform load over the clients
- the largest d which guarantees the local tree structure

$G_0(|A|,r)$: class of simple random regular graphs

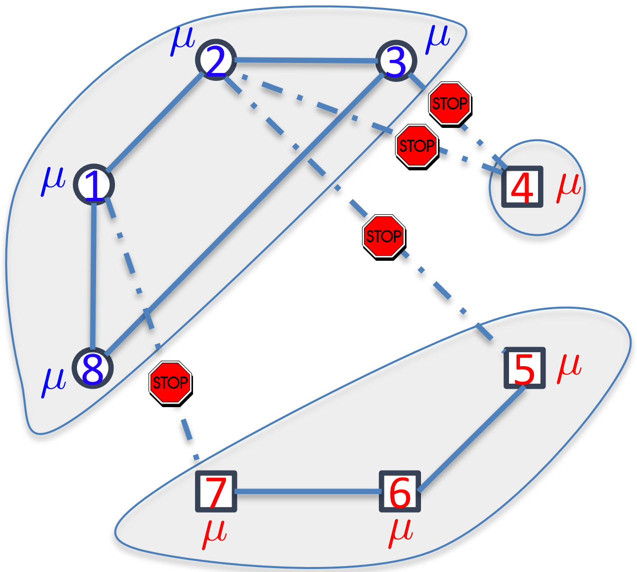


Choice of the graph and other parameters

Desiderata

- Large components CC_a and CC_a^d
- r small
- uniform load over the clients
- the largest d which guarantees the local tree structure

$G_0(|A|,r)$: class of simple random regular graphs



Theorem (informal)

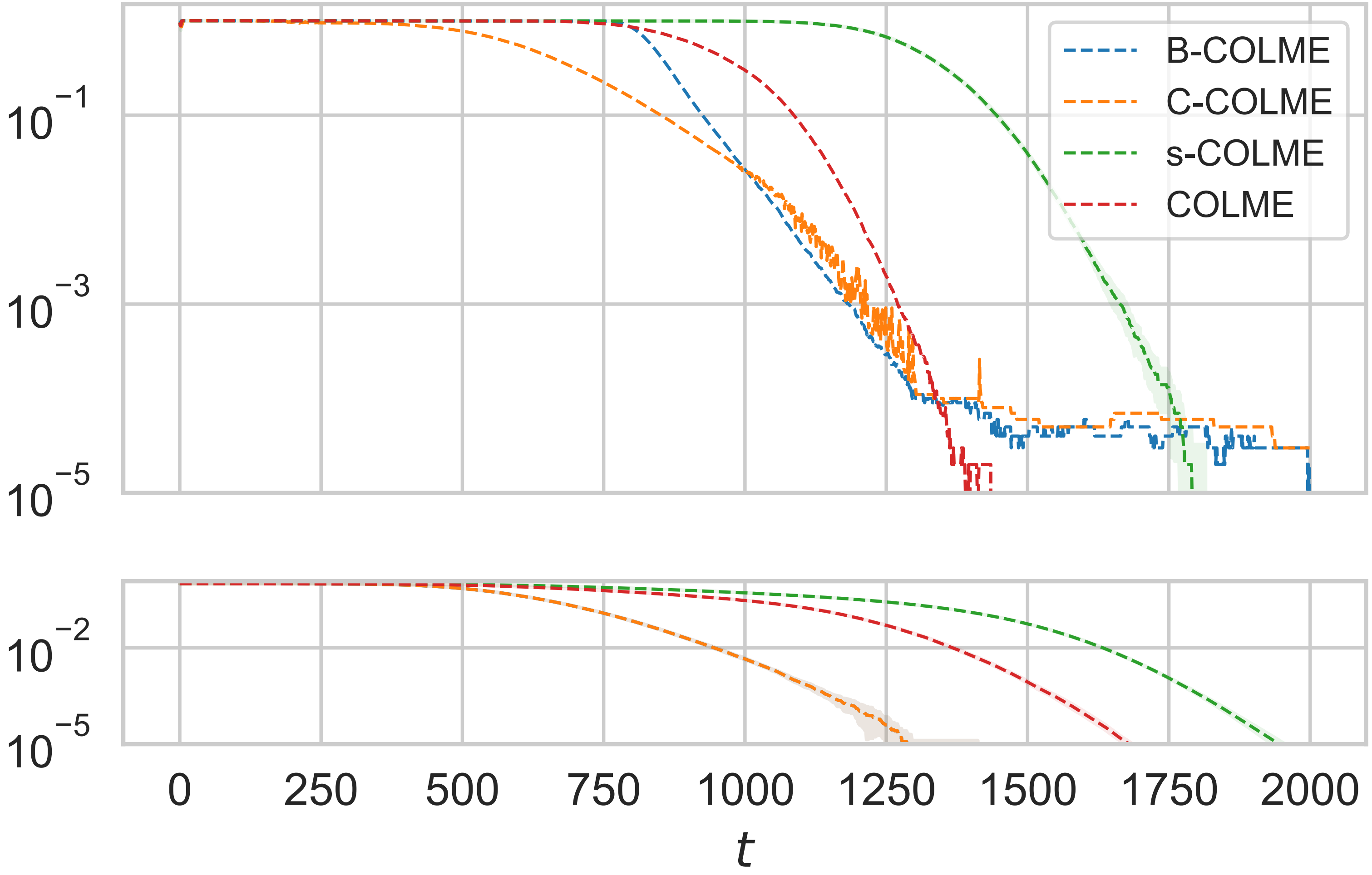
For d ~ log(|A|), r ~ log(1/δ) almost each agent has $|CC_a^d| > |A|^{1/2}$

Some numerical results

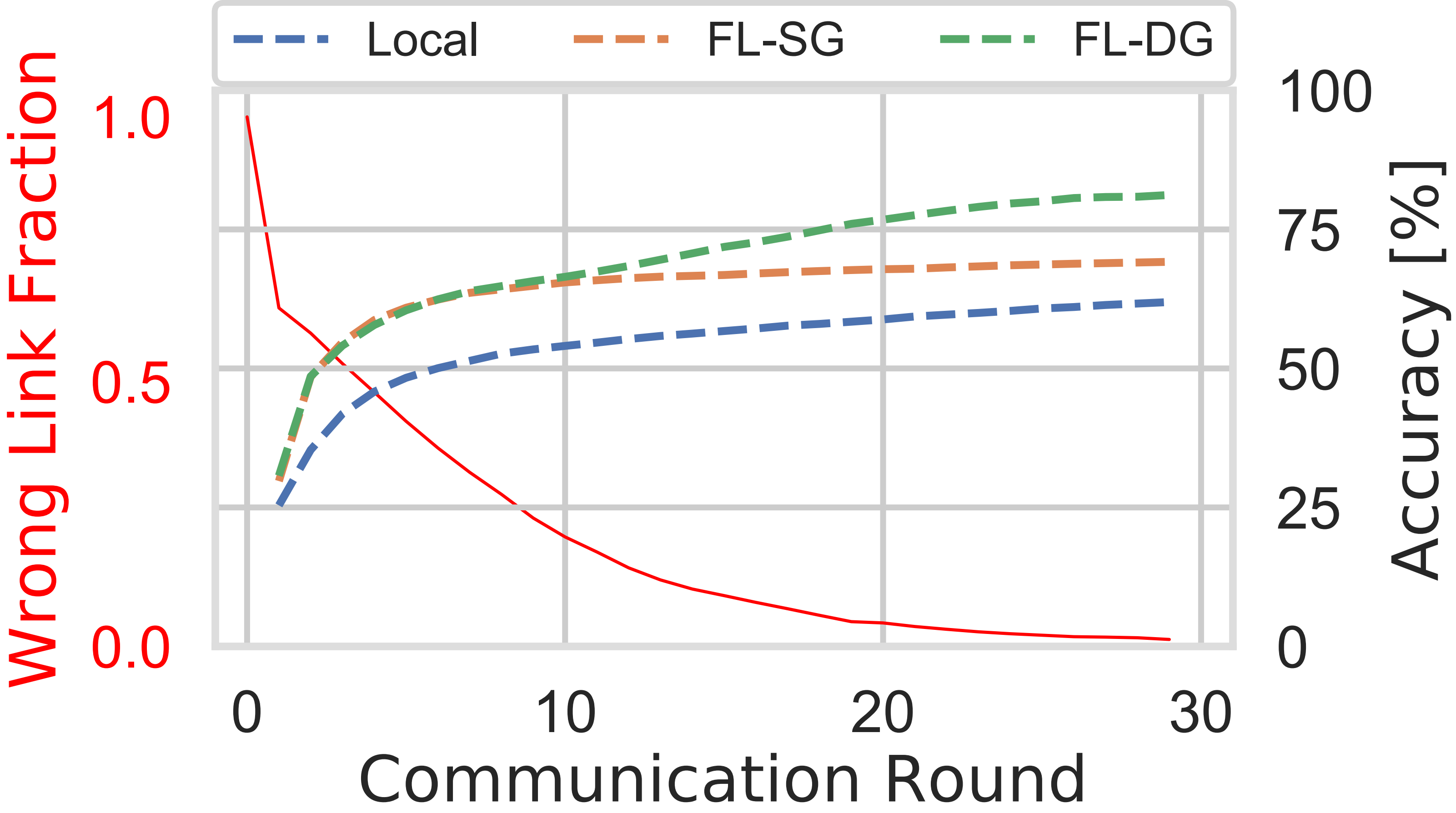
$|A| = 10^4$, $r = 10$, $d=5$, $\varepsilon = 0.1$, $\delta = 0.1$

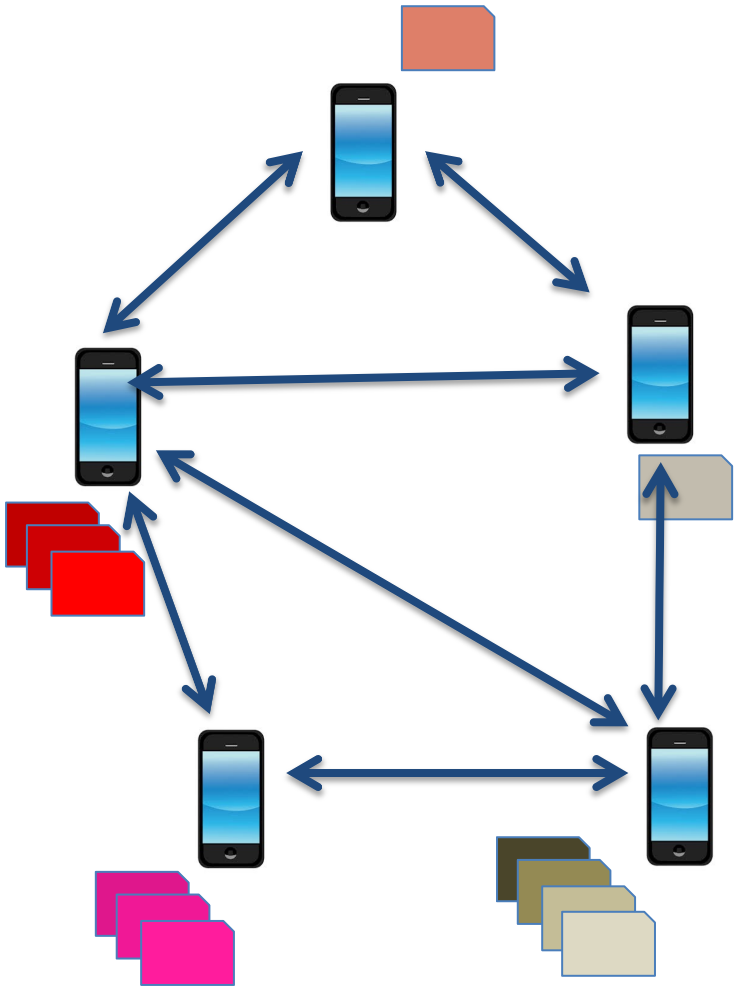
fraction of agents with wrong estimates

fraction of wrong links used



A FL training





Open questions

- Results for C-ColME under sub-gaussian distributions
- Rewire connections rather than pruning
- What if all agents have different distributions?
- How to extend this approach to more realistic FL problems?

Looking forward to discuss