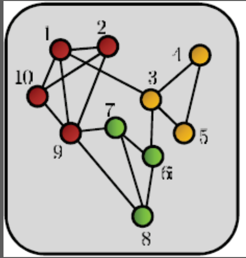


INFERENCE OVER NETWORKS

LECTURE #25: Extensions & Conclusions

Professor Ali H. Sayed
UCLA Electrical Engineering

Course EE210B
Spring Quarter 2015

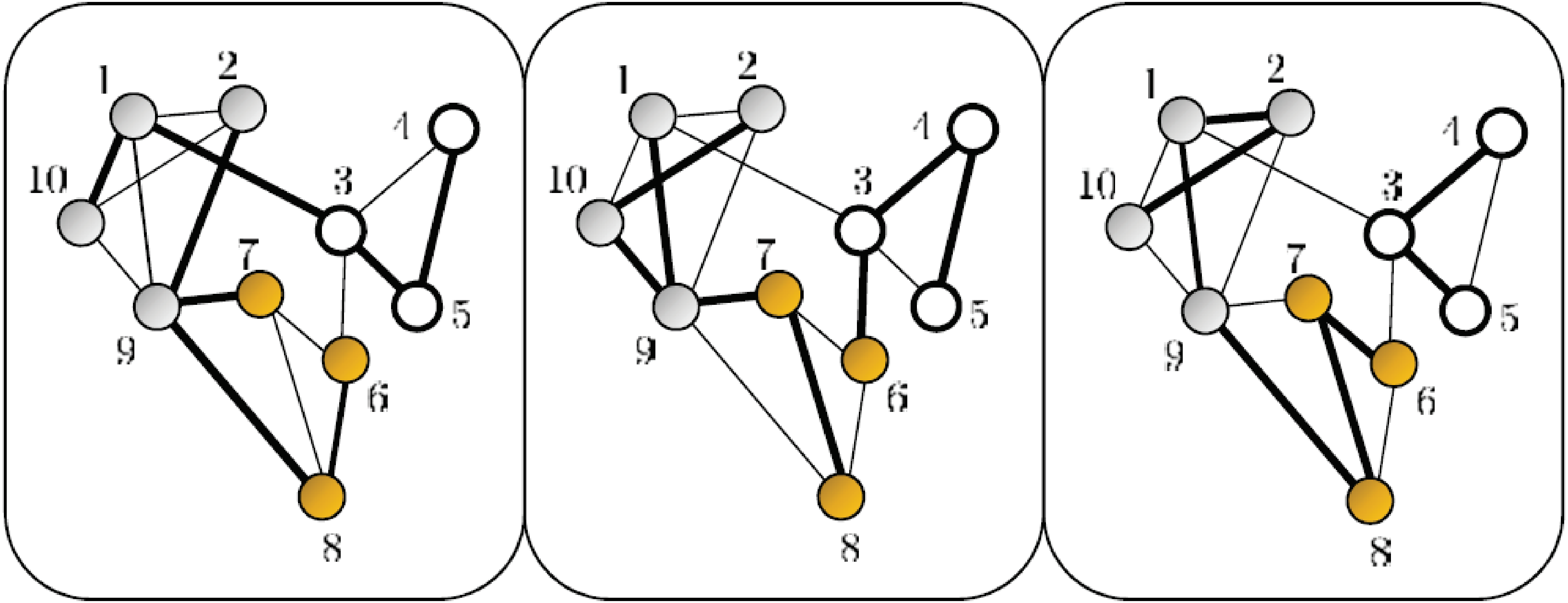
Proc. IEEE, vol. 102, no. 4, pp. 460-497, April 2014.
Foundations and Trends in Machine Learning, vol. 7, no. 4-5, pp. 311-801, July 2014.

Reference

Chapter 15 (Extensions & Conclusions, pp. 683-709):

A. H. Sayed, ``Adaptation, learning, and optimization over networks," ***Foundations and Trends in Machine Learning***, vol. 7, issue 4-5, pp. 311-801, NOW Publishers, 2014.

Gossip & Asynchronous Behavior



Gossip & Asynchronous Behavior

Gossip

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}}\left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}\right] \\ \text{agent } k \text{ picks randomly a neighbor } \ell_o \in \mathcal{N}_k \\ \boldsymbol{w}_{k,i} = a_k \boldsymbol{\psi}_{k,i} + (1-a_k)\boldsymbol{\psi}_{\ell_o,i} \end{cases}$$

Asynchronous

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \boldsymbol{\mu}_k(i)\widehat{\nabla_{w^{\mathsf{T}}} J}_k(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} = \displaystyle\sum_{\ell \in \boldsymbol{\mathcal{N}}_{k,i}} \boldsymbol{a}_{\ell k}(i)\, \boldsymbol{\psi}_{\ell,i} \end{cases}$$

Gossip & Asynchronous Behavior

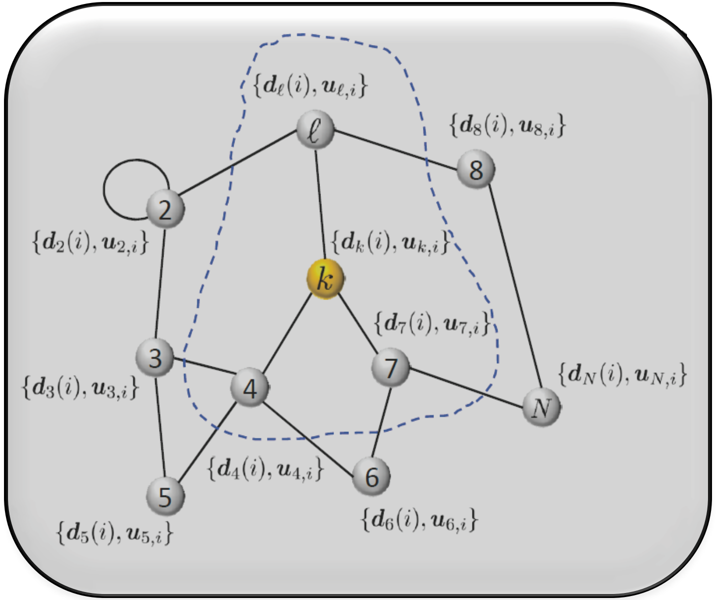
$$\begin{aligned} \alpha_{\text{async}} &= \alpha_{\text{sync}} + O\left(\mu_{\max}^{(N^2+1)/N^2}\right) \\ \text{MSD}_{\text{async}} &= \text{MSD}_{\text{sync}} + O\left(\mu_{\max}\right) \end{aligned}$$

Sparsity Constraints

$$J_k(w) = \mathbb{E}\,(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w)^2 + \gamma f(w)$$

$$\begin{cases} \boldsymbol{e}_k(i) = \boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1} \\ \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}}\boldsymbol{e}_k(i) - \mu_k\gamma\,\partial f(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{w}_{k,i} = \sum_{\ell\in\mathcal{N}_k} a_{\ell k}\,\boldsymbol{\psi}_{\ell,i} \end{cases}$$

$$\partial f(w) = \begin{bmatrix} \frac{\text{sign}(w_1)}{\epsilon+|w_1|} & \frac{\text{sign}(w_2)}{\epsilon+|w_2|} & \ldots & \frac{\text{sign}(w_M)}{\epsilon+|w_M|} \end{bmatrix}$$



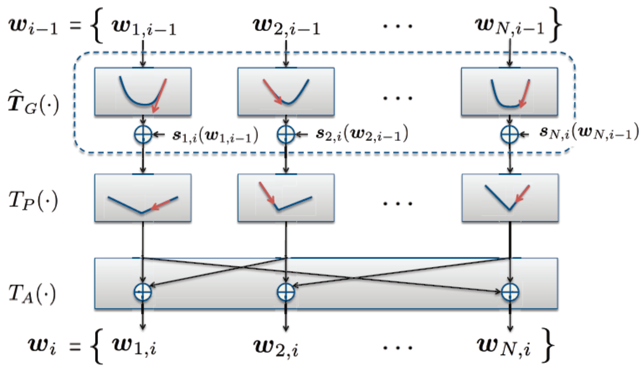
Proximal Diffusion

$$J_k^{\text{reg}}(w) \triangleq J_k(w) + \delta\mu^{\nu} R_k^{\text{org}}(w) \triangleq J_k(w) + R_k(w)$$

non-differentiable

$$\begin{cases} \phi_{k,i} = \text{prox}_{\mu R_k}\left\{ w_{k,i-1} - \mu \widehat{\nabla_w J}_k(w_{k,i-1}) \right\} \\ w_{k,i} = \sum_{\ell=1}^{N} a_{\ell k}\phi_{\ell,i} \end{cases}$$

$$\text{prox}_{\mu R_k}(x) \triangleq \arg\min_u \left(R_k(u) + \frac{1}{2\mu}\|x-u\|_2^2 \right)$$



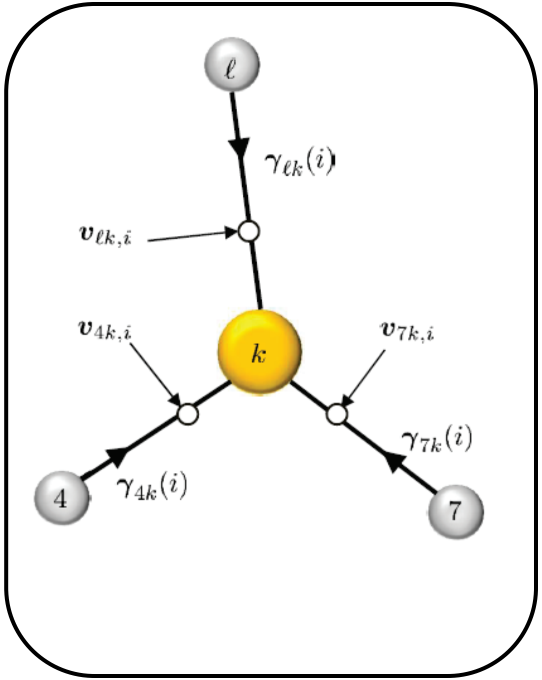
.. Proximal diffusion as a cascade of operators.

Noisy Exchanges

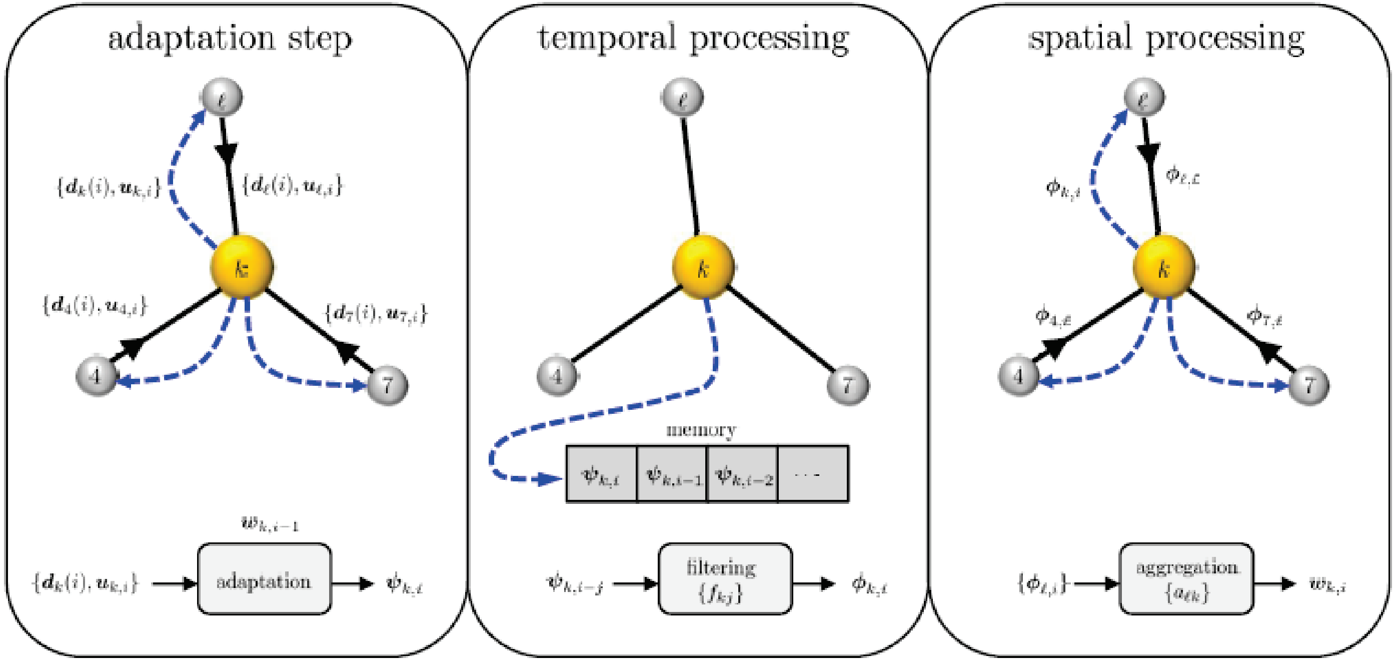
$$\boldsymbol{\psi}_{\ell k,i} \;=\; \boldsymbol{\gamma}_{\ell k}(i)\boldsymbol{\psi}_{\ell,i} + \boldsymbol{v}^{(\psi)}_{\ell k,i}$$

fading **noise**

$$\begin{cases} \boldsymbol{\psi}_{k,i} \;=\; \boldsymbol{\psi}_{k,i-1} + 2\mu_k \boldsymbol{u}^{\mathsf{T}}_{k,i}\left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{\psi}_{k,i-1}\right] \\ \boldsymbol{w}_{k,i} \;=\; \displaystyle\sum_{\ell\in\mathcal{N}_k} a_{\ell k}\, \boldsymbol{\psi}_{\ell k,i} \end{cases}$$



Temporal Diversity



Temporal Diversity

$$\begin{cases} \boldsymbol{\psi}_{k,i} &= \boldsymbol{w}_{k,i-1} + 2\mu_k \boldsymbol{u}_{k,i}^{\mathsf{T}} \left[\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}\boldsymbol{w}_{k,i-1}\right] \\ \boldsymbol{\phi}_{k,i} &= \displaystyle\sum_{j=0}^{L-1} f_{kj}\boldsymbol{\psi}_{k,i-j} \qquad \text{(temporal processing)} \\ \boldsymbol{w}_{k,i} &= \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k}\boldsymbol{\phi}_{\ell,i} \qquad \text{(spatial processing)} \end{cases}$$

$$f_{kj} \geq 0, \quad \sum_{j=0}^{L-1} f_{kj} = 1$$

Constrained Optimization

$$\min_{w} \sum_{k=1}^{N} J_k(w)$$
$$\text{subject to } w \in \mathbb{W}_1 \cap \mathbb{W}_2 \cap \ldots \cap \mathbb{W}_N$$

(convex sets)

$$\mathbb{W}_k \triangleq \left\{ w : \begin{array}{ll} h_{k,m}(w) = 0, & m = 1, 2, \ldots, U_k \\ g_{k,n}(w) \leq 0, & n = 1, 2, \ldots, L_k \end{array} \right.$$

(equality & inequality constraints)

Constrained Optimization

Select convex and twice-differentiable penalty functions satisfying:

$$\delta^{\mathrm{IP}}(x) = \begin{cases} 0, & x \le 0 \\ > 0, & x > 0 \end{cases} \qquad \delta^{\mathrm{EP}}(x) = \begin{cases} 0, & x = 0 \\ > 0, & x \ne 0 \end{cases}$$

$$\boxed{\delta^{\mathrm{IP}}(x) = \max\left\{0, \frac{x^3}{\sqrt{x^2+\rho^2}}\right\}, \qquad \delta^{\mathrm{EP}}(x) = x^2}$$

Constrained Optimization

Penalty function: $p_k(w) \triangleq \sum_{n=1}^{L_k} \delta^{\text{IP}}\left(g_{k,n}(w)\right) + \sum_{m=1}^{U_k} \delta^{\text{EP}}\left(h_{k,m}(w)\right)$

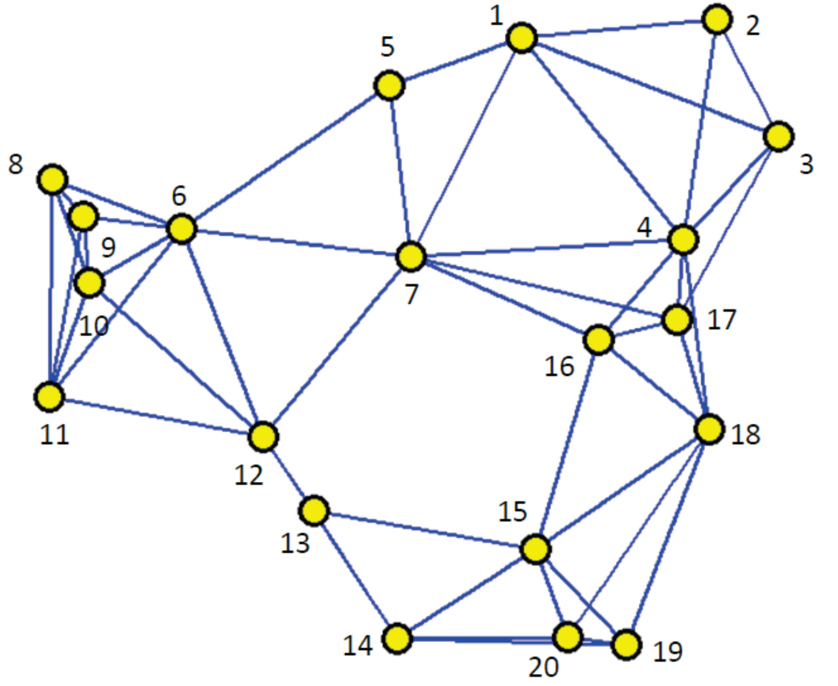
$$\begin{cases} \boldsymbol{\zeta}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu\, \widehat{\nabla_{w^\top} J}_k(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{\psi}_{k,i} = \boldsymbol{\zeta}_{k,i} - \mu^{1-\theta}\, \nabla_{w^\top} p_k(\boldsymbol{\zeta}_{k,i}) \\ \boldsymbol{w}_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i} \end{cases}$$

(penalized diffusion)

$0 < \theta < 1$

$$\lim_{\mu \to 0} \limsup_{i \to \infty} \mathbb{E}\,\|w^o - \boldsymbol{w}_{k,i}\|^2 = 0$$

Example #A: Constrained Optimization



$$J_k(w) = \mathbb{E}\left(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i}w\right)^2$$

$$\boldsymbol{d}_k(i) \;=\; \boldsymbol{u}_{k,i}w^\bullet + \boldsymbol{v}_k(i)$$

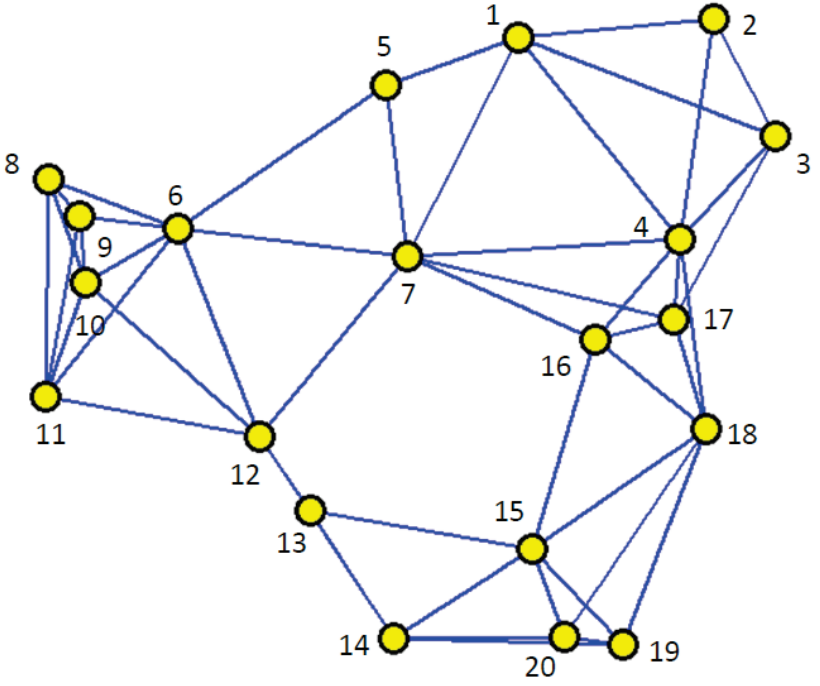
$$g_{k,i}(w) \;=\; b_{k,i}^{\mathsf{T}}w - z_k(i)$$

$$L_k = 1,\, U_k = 0$$

$$\mu = 0.002,\ \theta = 0.9,\ \text{and}\ \rho = 0.001$$

Metropolis rule

Example #A: Constrained Optimization



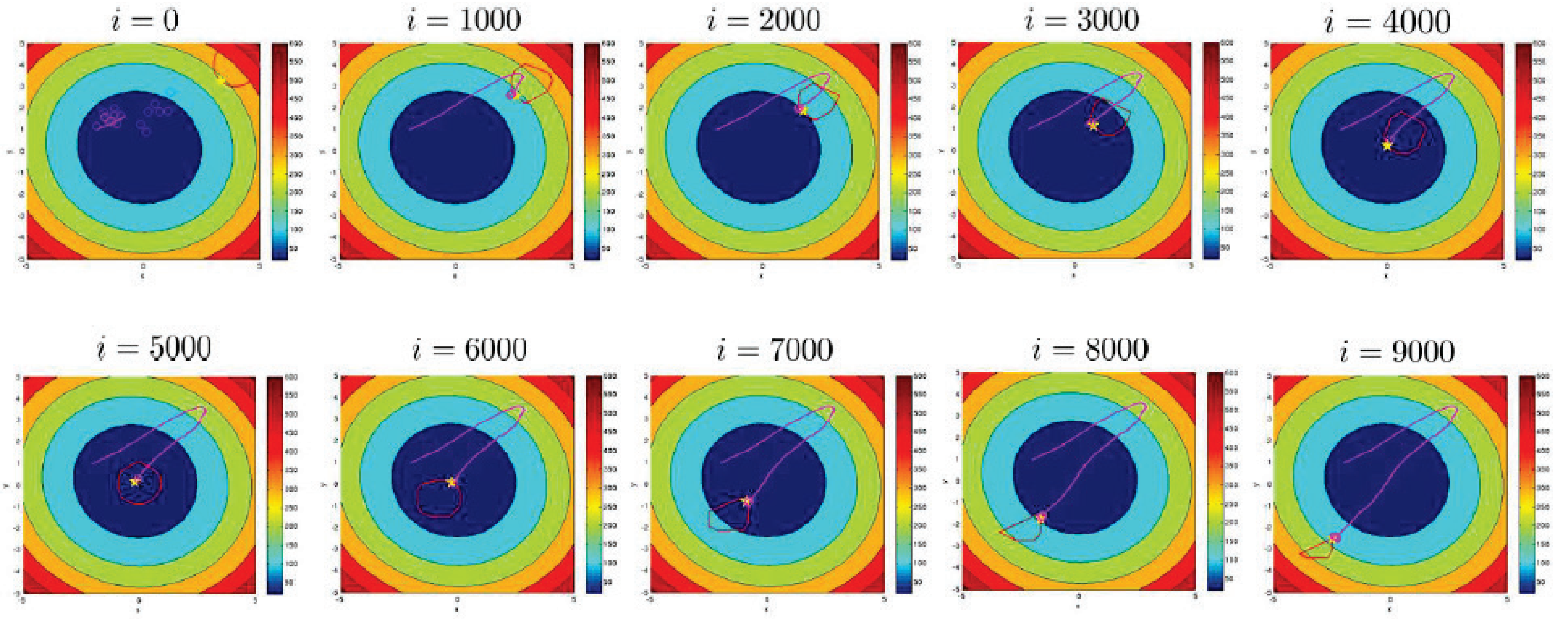
$$B_i \triangleq \text{col}\{b_{1,i}^{\mathsf{T}},\, b_{2,i}^{\mathsf{T}}, \ldots, b_{N,i}^{\mathsf{T}}\}$$

$$z_i \triangleq \text{col}\{z_1(i),\, z_2(i), \ldots, z_N(i)\}$$

$$\min_w \sum_{k=1}^{N} \mathbb{E}\,(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w)^2$$

$$\text{subject to } B_i w - z_i \preceq 0$$

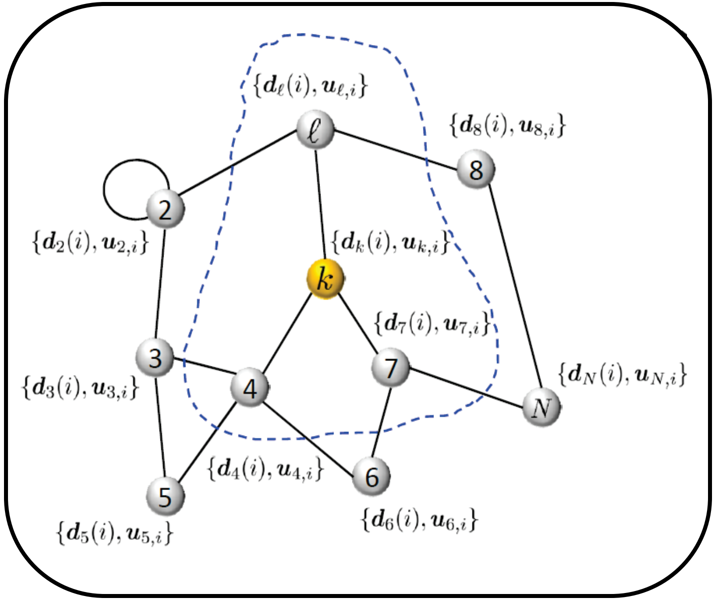
Example #A: Constrained Optimization



Distributed RLS

$$d_k(i) \;=\; u_{k,i}w^o \;+\; v_k(i)$$

$$\min_{w} \;\; \lambda^{i+1}\delta\|w\|^2 \;+\; \sum_{j=0}^{i}\lambda^{i-j}\left(\sum_{k=1}^{N}|d_k(j)-u_{k,j}w|^2\right)$$



Distributed RLS

Diffusion RLS strategy (ATC)

step 1 (initialization by agent k)

$$\psi_{k,i} \leftarrow w_{k,i-1}$$
$$P_{k,i} \leftarrow \lambda^{-1} P_{k,i-1}$$

step 2 (adaptation)

Update $\{\psi_{k,i}, P_{k,i}\}$ by iterating over $\ell \in \mathcal{N}_k$:

$$\psi_{k,i} \leftarrow \psi_{k,i} + \frac{c_{\ell k} P_{k,i} u^*_{\ell,i}}{1 + c_{\ell k} u_{\ell,i} P_{k,i} u^*_{\ell,i}} (d_{\ell,i} - u_{\ell,i}\psi_{k,i})$$

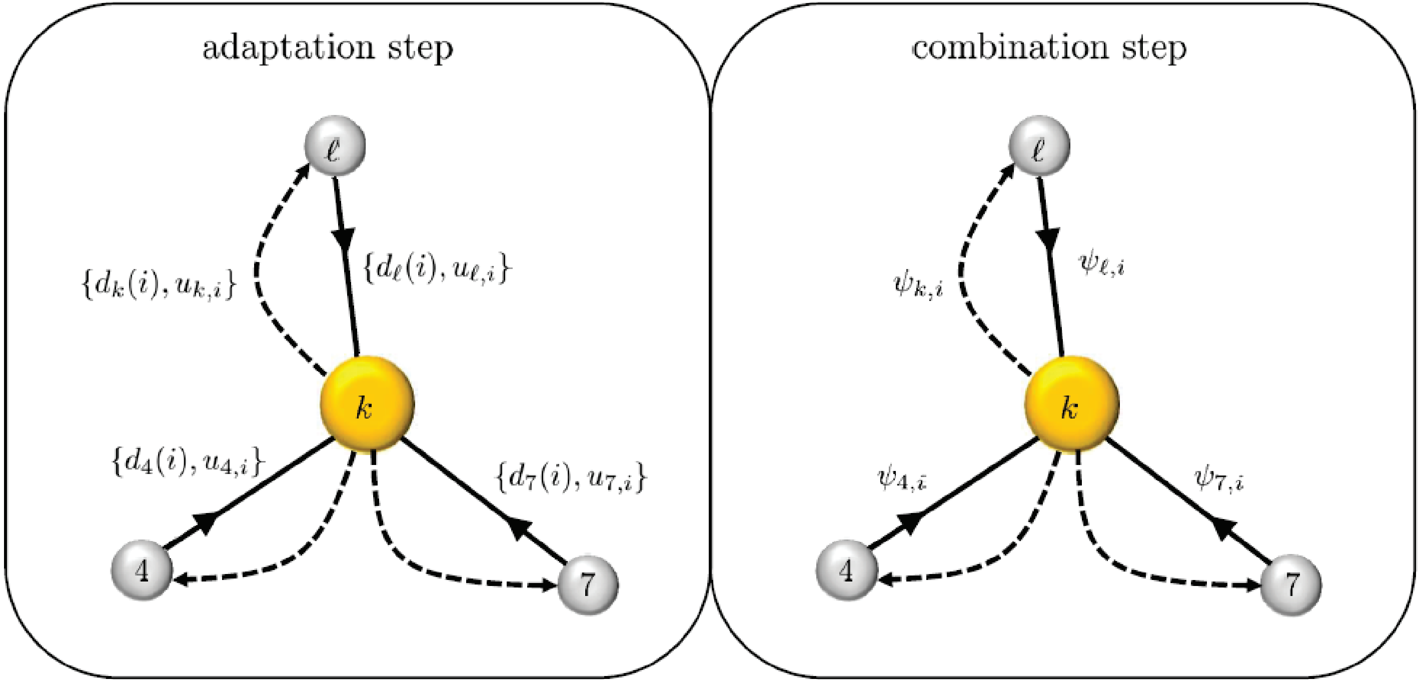
$$P_{k,i} \leftarrow P_{k,i} - \frac{c_{\ell k} P_{k,i} u^*_{\ell,i} u_{\ell,i} P_{k,i}}{1 + c_{\ell k} u_{\ell,i} P_{k,i} u^*_{\ell,i}}$$

end

step 3 (combination)

$$w_{k,i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$$

Distributed RLS



Distributed Kalman Filtering

$$
\begin{aligned}
\boldsymbol{x}_{i+1} &= F_i \boldsymbol{x}_i + G_i \boldsymbol{n}_i \\
\boldsymbol{y}_{k,i} &= H_{k,i} \boldsymbol{x}_i + \boldsymbol{v}_{k,i}, \quad k = 1, 2, \ldots, N
\end{aligned}
$$

$$
\mathbb{E} \begin{bmatrix} \boldsymbol{n}_i \\ \boldsymbol{v}_{k,i} \end{bmatrix} \begin{bmatrix} \boldsymbol{n}_j \\ \boldsymbol{v}_{k,j} \end{bmatrix}^* \triangleq \begin{bmatrix} Q_i & 0 \\ 0 & R_{k,i} \end{bmatrix} \delta_{ij}
$$

$$
\mathbb{E}\, \boldsymbol{x}_o \boldsymbol{x}_o^* = \Pi_o > 0
$$

Distributed Kalman Filtering

Time and measurement-form of diffusion Kalman filter

step 1 (initialization by agent k)

$$\psi_{k,i} \leftarrow \widehat{x}_{k,i|i-1}$$
$$P_{k,i} \leftarrow P_{k,i|i-1}$$

step 2 (adaptation)

Update $\{\psi_{k,i}, P_{k,i}\}$ by iterating over $\ell \in \mathcal{N}_k$:

$$R_e \leftarrow R_{\ell,i} + H_{\ell,i} P_{k,i} H_{\ell,i}^*$$
$$\psi_{k,i} \leftarrow \psi_{k,i} + P_{k,i} H_{\ell,i}^* R_e^{-1} \left(y_{\ell,i} - H_{\ell,i} \psi_{k,i} \right)$$
$$P_{k,i} \leftarrow P_{k,i} - P_{k,i} H_{\ell,i}^* R_e^{-1} H_{\ell,i} P_{k,i}$$

end

step 3 (combination)

$$\widehat{x}_{k,i|i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \psi_{\ell,i}$$
$$P_{k,i|i} = P_{k,i}$$
$$\widehat{x}_{k,i+1|i} = F_i \widehat{x}_{k,i|i}$$
$$P_{k,i+1|i} = F_i P_{k,i|i} F_i^* + G_i Q_i G_i^*$$

Distributed Kalman Filtering

Information form of the diffusion Kalman filter

step 1 (adaptation)

$$S_{k,i} = \sum_{\ell \in \mathcal{N}_k} H_{\ell,i}^* R_{\ell,i}^{-1} H_{\ell,i}$$

$$\boldsymbol{q}_{k,i} = \sum_{\ell \in \mathcal{N}_k} H_{\ell,i}^* R_{\ell,i}^{-1} \boldsymbol{y}_{\ell,i}$$

$$P_{k,i|i}^{-1} = P_{k,i|i-1}^{-1} + S_{k,i}$$

$$\boldsymbol{\psi}_{k,i} = \widehat{\boldsymbol{x}}_{k,i|i-1} + P_{k,i|i}\left(\boldsymbol{q}_{k,i} - S_{k,i}\widehat{\boldsymbol{x}}_{k,i|i-1}\right)$$

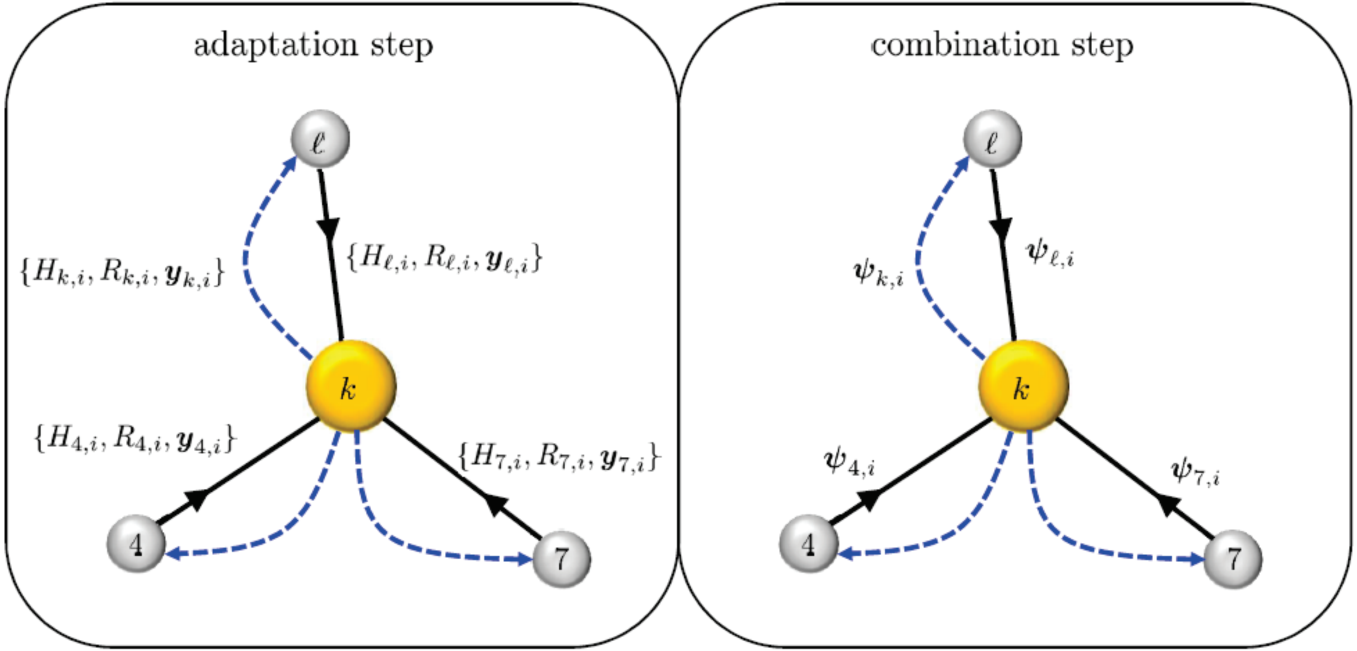
step 2: (combination)

$$\widehat{\boldsymbol{x}}_{k,i|i} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{\psi}_{\ell,i}$$

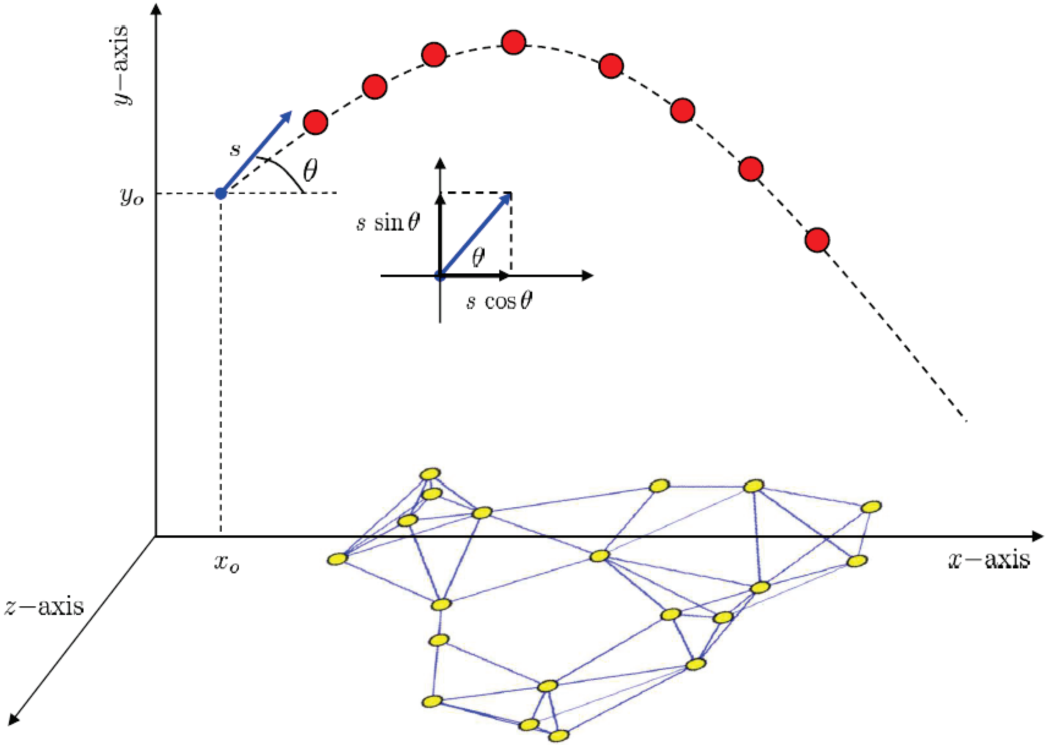
$$\widehat{\boldsymbol{x}}_{k,i+1|i} = F_i \widehat{\boldsymbol{x}}_{k,i|i}$$

$$P_{k,i+1|i} = F_i P_{k,i|i} F_i^* + G_i Q_i G_i^*$$

Distributed Kalman Filtering



Example #B (Target Tracking)



Example #B (Target Tracking)

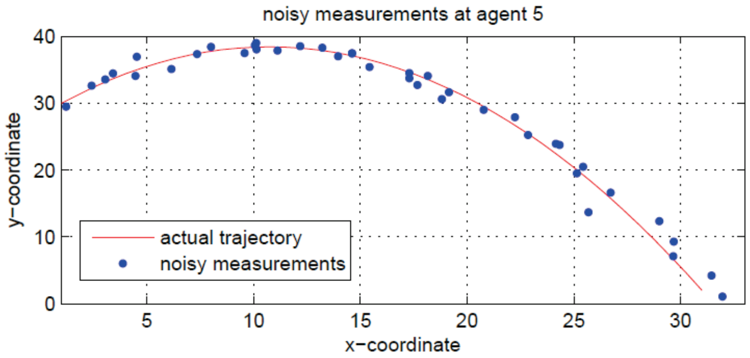
$$\underbrace{\begin{bmatrix} x(i+1) \\ y(i+1) \\ s_x(i+1) \\ s_y(i+1) \end{bmatrix}}_{\boldsymbol{x}_{i+1}} = \underbrace{\begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}}_{F} \underbrace{\begin{bmatrix} x(i) \\ y(i) \\ s_x(i) \\ s_y(i) \end{bmatrix}}_{\boldsymbol{x}_i} - \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}}_{d_i} gT$$

$$\boldsymbol{y}_{k,i} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}}_{H} \begin{bmatrix} x(i) \\ y(i) \\ s_x(i) \\ s_y(i) \end{bmatrix} + \boldsymbol{v}_{k,i}$$

Example #B (Target Tracking)

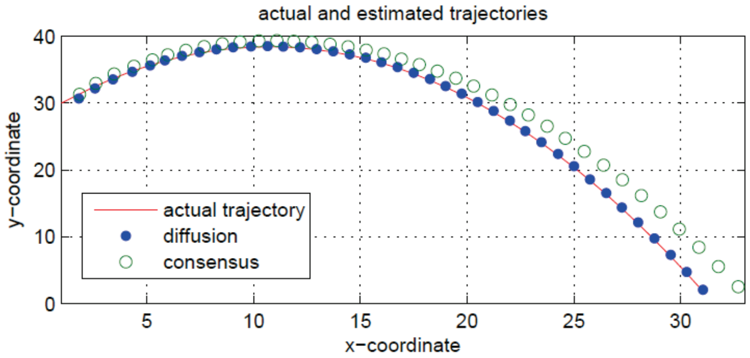
$$\Pi_o = I_4, \quad (x_o, y_o) = (1, 30)$$

$$s = 15, \quad T = 0.01, \quad \theta = 60^o$$



Consensus implementation:

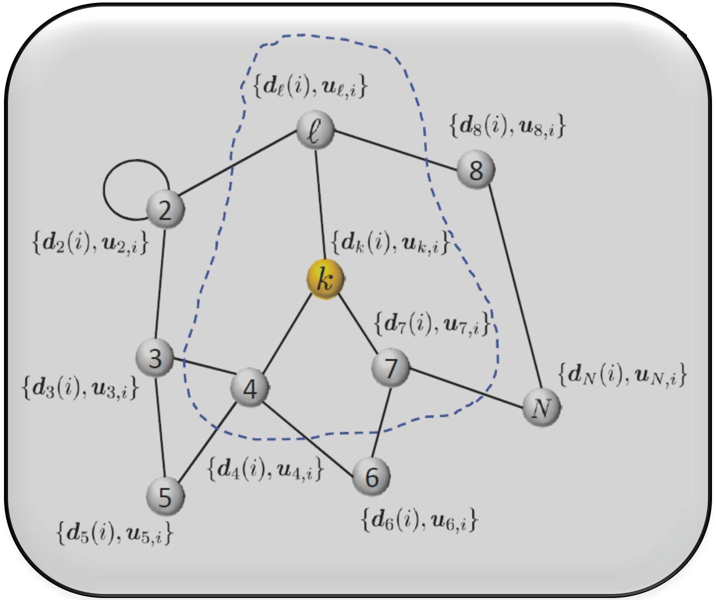
$$\begin{cases} \widehat{x}_{k,i|i} \;=\; (1+\epsilon - n_k\epsilon)\,\psi_{k,i} + \displaystyle\sum_{\ell\in\mathcal{N}_k\backslash\{k\}} \epsilon\,\psi_{\ell,i} \\ \epsilon = 0.001 \end{cases}$$



Primal-Dual Networks

We can also incorporate equality constraints explicitly:

$$\begin{cases} \min\limits_{w} & \sum\limits_{k=1}^{N} \mathbb{E}(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w_k)^2 \\ \text{s.t.} & w_1 = w_2 = \cdots = w_N \end{cases}$$

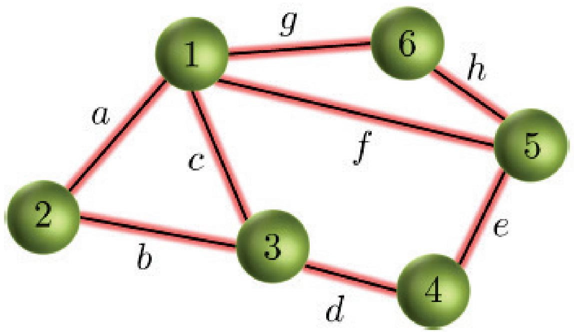


Primal-Dual **Networks**

Incidence matrix C (edges x agents):

$$c_{ek} = \begin{cases} +1, & k \text{ is the lower indexed node connected to } e \\ -1, & k \text{ is the higher indexed node connected to } e \\ 0, & \text{otherwise} \end{cases}$$

$$C = \begin{array}{c} \begin{array}{cccccc} \mathbf{1} & \mathbf{2} & \mathbf{3} & \mathbf{4} & \mathbf{5} & \mathbf{6} \end{array} \\ \left[\begin{array}{cccccc} 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{array}\right] \end{array} \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \\ g \\ h \end{array}$$



Primal-Dual Networks

Equivalent characterization:

$$\begin{aligned} \min_{w} \quad & \sum_{k=1}^{N} \mathbb{E}(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w_k)^2 \\ \text{s.t.} \quad & \mathcal{C}\mathcal{W} = 0_{MN} \end{aligned}$$

$$\mathcal{C} \triangleq C \otimes I_M$$

$$\mathcal{W} \triangleq \text{col}\{w_1, \ldots, w_N\}$$

Augmented Lagrangian:

$$f(\mathcal{W}, \lambda) = \sum_{k=1}^{N} \mathbb{E}(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w_k)^2 + \lambda^{\mathsf{T}} \mathcal{C}\mathcal{W} + \frac{\eta}{2} \|\mathcal{C}\mathcal{W}\|^2$$

Primal-Dual Networks

$$f(\mathcal{W},\lambda) = \sum_{k=1}^{N} \mathbb{E}(\boldsymbol{d}_k(i) - \boldsymbol{u}_{k,i} w_k)^2 + \lambda^{\mathsf{T}} \mathcal{C}\mathcal{W} + \frac{\eta}{2}\|\mathcal{C}\mathcal{W}\|^2$$

<u>Dual function:</u> $g(\lambda) = \min_{\mathcal{W}} f(\mathcal{W},\lambda)$

<u>Dual variable:</u> $\lambda^o = \arg\max_{\lambda} g(\lambda)$

First-order augmented Lagrangian algorithm:

$$\begin{aligned} \boldsymbol{\mathcal{W}}_i &= \boldsymbol{\mathcal{W}}_{i-1} - \mu \widehat{\nabla_w} f(\boldsymbol{\mathcal{W}}_{i-1}, \boldsymbol{\lambda}_{i-1}) \\ \boldsymbol{\lambda}_i &= \boldsymbol{\lambda}_{i-1} + \mu \nabla_{\lambda} f(\boldsymbol{\mathcal{W}}_{i-1}, \boldsymbol{\lambda}_{i-1}) \end{aligned}$$

Example #C (Primal-Dual Networks)

Distributed Augmented Lagrangian (AL) Algorithm

$$\psi_{k,i-1} = w_{k,i-1} - \mu \sum_{e=1}^{E} c_{ek}\lambda_{e,i-1} - \mu\eta \sum_{\ell\in\mathcal{N}_k} l_{k\ell} w_{\ell,i-1}$$

$$w_{k,i} = \psi_{k,i-1} + \mu u_{k,i}^{\mathsf{T}}(d_k(i) - u_{k,i}w_{k,i-1})$$

$$\lambda_{e,i} = \lambda_{e,i-1} + \mu(w_{k,i-1} - w_{\ell,i-1}) \quad [\ell > k, \ell\in\mathcal{N}_k]$$

$\mathcal{L} = \mathcal{C}^{\mathsf{T}}\mathcal{C}$ (Laplacian)

Example #C (Primal-Dual Networks)

Distributed Arrow-Hurwicz (AH) Algorithm $(\eta = 0)$

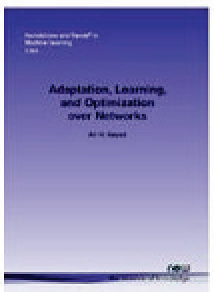
$$
\begin{aligned}
\psi_{k,i-1} &= w_{k,i-1} - \mu \sum_{e=1}^{E} c_{ek} \lambda_{e,i-1} \\
w_{k,i} &= \psi_{k,i-1} + \mu u_{k,i}^{\mathsf{T}} (d_k(i) - u_{k,i} w_{k,i-1}) \\
\lambda_{e,i} &= \lambda_{e,i-1} + \mu (w_{k,i-1} - w_{\ell,i-1}) \quad [\ell > k, \ell \in \mathcal{N}_k]
\end{aligned}
$$

Example #C (Primal-Dual Networks)

Algorithm	Handles Partial Observation	Stability Range	Steady-state MSD
Diffusion Strategy	✓	$0 < \mu < \bar{\mu}$	$\frac{1}{N}\text{MSD}^{NC}$
Consensus Strategy	✓	$0 < \mu < \mu^{c}(L) < \bar{\mu}$	$\frac{1}{N}\text{MSD}^{NC}$
No Cooperation	×	$0 < \mu < \bar{\mu}$	$\text{MSD}^{NC} \triangleq \frac{\mu M}{2N}\sum_{k=1}^{N}\sigma_{v,k}^{2}$
Arrow-Hurwicz Method	×[a]	$0 < \mu < \mu^{AH}(L) < \bar{\mu}$	MSD^{NC}
Augmented Lagrangian Method	✓[b]	$0 < \mu < \frac{\mu^{AL}(L)}{\eta}$	$\frac{1}{N}\text{MSD}^{NC} + O\left(\frac{1}{\eta}\right)$

References

A. H. Sayed, ``Adaptation, learning, and optimization over networks,'' ***Foundations and Trends in Machine Learning***, vol. 7, issue 4-5, pp. 311-801, NOW Publishers, 2014.



A. H. Sayed, ``Adaptive networks,'' ***Proceedings of the IEEE***, vol. 102, no. 4, pp. 460-497, April 2014.

A. H. Sayed et al, ``Diffusion strategies for adaptation and learning over networks,'' ***IEEE Signal Process. Mag.***, vol. 30, no. 3, pp. 155-171, May 2013.

A. H. Sayed, ``Diffusion adaptation over networks,'' in ***Academic Press Library in Signal Processing***, vol. 3, pp. 323-454, Academic Press, Elsevier, 2014.

Former PhD Students

Cassio Lopes (PhD, 2008)

Federico Cattivelli (PhD, 2010)

Sheng-Yuan Tu (PhD, 2013)

Jianshu Chen (PhD, 2014)

Zaid Towfic (PhD, 2014)

Xiaochuan Zhao (PhD, 2014)

End of Course
Thank you!

Proc. IEEE, vol. 102, no. 4, pp. 460-497, April 2014.
Foundations and Trends in Machine Learning, vol. 7, no. 4-5, pp. 311-801, July 2014.