D4L: Decentralized Dynamic Discriminative Dictionary Learning

Alec Koppel[⋆], Garrett Warnell[†], Ethan Stump[†], Alejandro Ribeiro[⋆]

[⋆]Dept. of Electrical and Systems Engineering, University of Pennsylvania

[†]U.S. Army Research Laboratory, Adelphi, MD

Int. Conf. on Intelligent Robots and Systems

Hamburg, Germany, Sept. 30, 2015

- Goal: visual awareness in mobile robotic teams in unknown domains
  - $\Rightarrow$ Focus on cases where external state information unavailable
  - $\Rightarrow$ Little a priori knowledge of environment on platform
- Online (real-time) training algorithms necessary for this setting
  - $\Rightarrow$ gain awareness of operating environment
  - $\Rightarrow$ potentially leverage this info. for closed-loop control
- Dist. protocol useful when can't afford latency of centralization
  - $\Rightarrow$ no base-station $\Rightarrow$ better suited to distributed control

Online Discriminative Learning

- Robot i observes signals $\boldsymbol{\theta}_{i,t} \in \boldsymbol{\Theta}$, $t = 1, \ldots$ based on path it takes
  - $\Rightarrow$ predict environmental properties $\mathbf{y}_{i,t} \in \mathcal{Y}$ with this info
  - $\Rightarrow$ formulate as stoch. opt. problem: $\min_{\mathbf{x}_i} \mathbb{E}_{\boldsymbol{\theta}_i, \mathbf{y}_i}[f(\mathbf{x}_i; (\boldsymbol{\theta}_i, \mathbf{y}_i))]$
  - $\Rightarrow$ loss function f, regressor $\mathbf{x}_i$ $\Rightarrow$ discriminative model
- Wrinkle: individual robots only have info. based on traversed path
  - $\Rightarrow$ may omit regions of feature space crucial for effective prediction.
- Communicate with robotic network
  - $\Rightarrow$ greater domain "understanding" among individual robots
- This work: distributed online predictive algorithms in robotic teams
- Develop new capability: individual robots make global inferences
  - $\Rightarrow$ only observe distinct subsets of feature space

Pattern Recognition

- If relationship between random pair $(\boldsymbol{\theta}, \mathbf{y})$ is complicated. . .
  $\Rightarrow$ use alternative encoding of $\boldsymbol{\theta}$ $\Rightarrow$ reveal latent data structure
- DSP methods mostly rely on alternative signal representations
  $\Rightarrow$ Based on processing task (e.g. Fourier basis, wavelets, PCA)
- In dictionary learning, learn representation directly from data
  $\Rightarrow$ Task-driven: tailor dictionary to learning task (Mairal '12)
- We extend task-driven dictionary learning to multi-robot settings
  $\Rightarrow$ online visual awareness in mobile robotic teams

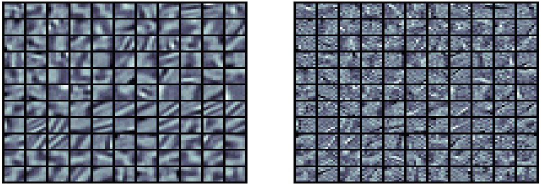


Figure : Initialized (left) and learned (right) dictionary for small image patches.

Dictionary Learning

- Represent signals $\boldsymbol{\theta}_t$ as combos. of k basis elements $\{\mathbf{d}_l\}_{l=1}^k$
  - $\Rightarrow$ learn dictionary $\mathbf{D} \in \mathbb{R}^{m \times k}$ from data
  - $\Rightarrow$ Denote coding (coefficients) of $\boldsymbol{\theta}_t$ w.r.t. dictionary as $\boldsymbol{\alpha}_t \in \mathbb{R}^k$
- Representation loss $g(\boldsymbol{\alpha}_t, \mathbf{D}; \boldsymbol{\theta}_t)$ $\Rightarrow$ small if $\mathbf{D}\boldsymbol{\alpha}_t$ and $\boldsymbol{\theta}_t$ close
  - $\Rightarrow$ $\mathbf{D}\boldsymbol{\alpha}_t$ is representation of $\boldsymbol{\theta}_t$ w.r.t dictionary $\mathbf{D}$
- Formulate the coding problem (lasso, elastic net)

$$\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t) := \underset{\boldsymbol{\alpha}_t \in \mathbb{R}^k}{\operatorname{argmin}}\, g(\boldsymbol{\alpha}_t, \mathbf{D}; \boldsymbol{\theta}_t) \; .$$

- Dictionary learning
  - $\Rightarrow$ seek $\mathbf{D}$ such that signals $\boldsymbol{\theta}_t$ well-represented by $\mathbf{D}\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t)$

Discriminative Dictionary Learning

- Tailor dictionary to discriminative modeling task
- Use coding $\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t)$ as representation of signal $\boldsymbol{\theta}_t$

- Decision variable $\mathbf{x}$ $\Rightarrow$ predict the label/vector $\mathbf{y}_t$ given $\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t)$.
- Loss function $f\big(\mathbf{D}, \mathbf{x}; (\boldsymbol{\theta}_t, \mathbf{y}_t)\big) = f\big(\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t), \mathbf{D}, \mathbf{x}; (\boldsymbol{\theta}_t, \mathbf{y}_t)\big)$

  $\Rightarrow$ predictive quality of $\mathbf{x}$ for output var. $\mathbf{y}_t$ given coding $\boldsymbol{\alpha}^*(\mathbf{D}; \boldsymbol{\theta}_t)$

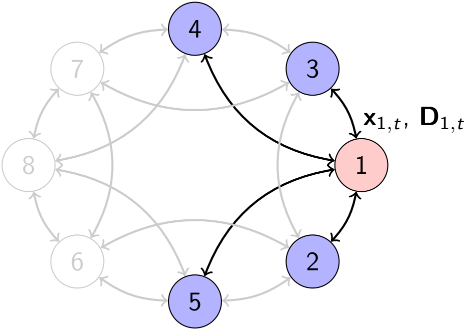
- Discriminative dictionary learning

$$(\mathbf{D}^*, \mathbf{x}^*) := \underset{\mathbf{D} \in \mathcal{D}, \mathbf{x} \in \mathcal{X}}{\operatorname{argmin}} \, \mathbb{E}_{\boldsymbol{\theta}, \mathbf{y}} \Big[f\big(\mathbf{D}, \mathbf{x}; (\boldsymbol{\theta}, \mathbf{y})\big) \Big].$$

$\Rightarrow$ Learn jointly regression weights $\mathbf{x}$ and dictionary $\mathbf{D}$

$\Rightarrow$ Non-convex stochastic program

- Robotic team
  - $\Rightarrow$ graph $\mathcal{G} = (V, \mathcal{E})$
  - $\Rightarrow |V| = N$, $|\mathcal{E}| = M$
- Neighborhood of robot i
  - $\Rightarrow n_i = \{j : (j, i) \in \mathcal{E}\}$



- Each robot i aims to learn a regressor $\mathbf{x}$ and dictionary $\mathbf{D}$
  - $\Rightarrow$ over observations of whole network $\{(\boldsymbol{\theta}_i, \mathbf{y}_i)\}_{i=1}^N$

$$(\mathbf{x}^*, \mathbf{D}^*) = \underset{\mathbf{x},\mathbf{D}}{\text{argmin}} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}_i, \mathbf{y}_i}\Big[f\big(\mathbf{D}, \mathbf{x}; (\boldsymbol{\theta}_i, \mathbf{y}_i)\big)\Big]$$

- We develop distributed online iterative methods for this problem

Collaboration via Lagrange Duality

- Incentivize agreement via constraint $\mathbf{D}_i = \mathbf{D}_j, \mathbf{x}_i = \mathbf{x}_j$ for all $j \in n_i$
- Decentralized task-driven dictionary learning problem

$$\{\mathbf{D}_i^*, \mathbf{x}_i^*\}_{i=1}^N := \underset{\mathbf{D}_i \in \mathcal{D}, \mathbf{x}_i \in \mathcal{X}}{\text{argmin}} \sum_{i=1}^{N} \mathbb{E}_{\boldsymbol{\theta}_i, \mathbf{y}_i} \left[f\left(\mathbf{D}_i, \mathbf{x}_i; (\boldsymbol{\theta}_i, \mathbf{y}_i)\right)\right].$$
$$\text{such that} \quad \mathbf{D}_i = \mathbf{D}_j, \mathbf{x}_i = \mathbf{x}_j \text{ for all } j \in n_i$$

- Enforcing agreement constraint would require global coordination

$\Rightarrow$ Define stochastic Lagrangian relaxation

$$\hat{\mathcal{L}}_t(\mathbf{D}, \mathbf{x}, \boldsymbol{\Lambda}, \boldsymbol{\nu}) = \sum_{i=1}^{N} \left[f\left(\mathbf{D}_i, \mathbf{x}_i; (\boldsymbol{\theta}_{i,t}, \mathbf{y}_{i,t})\right)\right] + \text{tr}\left(\boldsymbol{\Lambda}^T \mathbf{C}_D \mathbf{D}\right) + \boldsymbol{\nu}^T \mathbf{C}_x \mathbf{x}$$

$\Rightarrow$ Apply saddle point to stochastic Lagrangian $\Rightarrow$ distributed alg.

Stochastic Saddle Point Algorithm

- At robot i, time t, observe $(\boldsymbol{\theta}_{i,t}, \mathbf{y}_{i,t})$,
- Compute coding $\boldsymbol{\alpha}^{\star}_{i,t+1} = \text{argmin}_{\boldsymbol{\alpha}\in\mathbb{R}^k}\, g(\boldsymbol{\alpha}, \mathbf{D}_{i,t}; \boldsymbol{\theta}_{i,t})$
  $\Rightarrow$ In practice chosen as *sparse coding* via lasso or elastic-net
- Update primal variables at robot i

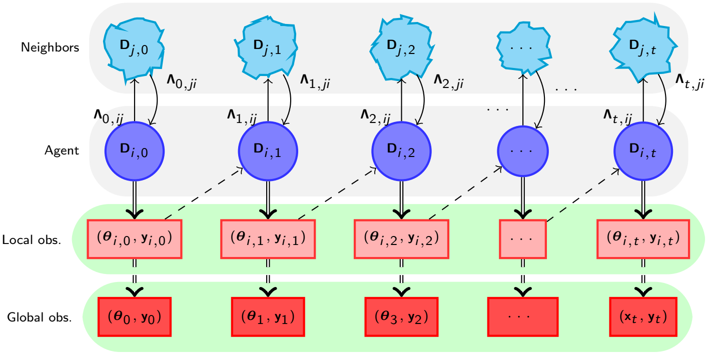
$$\mathbf{D}_{i,t+1} = \mathbf{D}_{i,t} - \epsilon_t \Big(\nabla_{\mathbf{D}_i} f_i(\mathbf{D}_{i,t}, \mathbf{x}_{i,t}; (\boldsymbol{\theta}_{i,t}, \mathbf{y}_{i,t})) + \sum_{j\in n_i} (\boldsymbol{\Lambda}_{ij,t} - \boldsymbol{\Lambda}_{ji,t}) \Big) \,,$$

$$\mathbf{x}_{i,t+1} = \mathbf{x}_{i,t} - \epsilon_t \Big(\nabla_{\mathbf{x}_i} f_i(\mathbf{D}_{i,t}, \mathbf{x}_{i,t}; (\boldsymbol{\theta}_{i,t}, \mathbf{y}_{i,t})) + \sum_{j\in n_i} (\boldsymbol{\nu}_{ij,t} - \boldsymbol{\nu}_{ji,t}) \Big) \,,$$

- Update dual variables at network communication link (i, j)

$$\boldsymbol{\Lambda}_{ij,t+1} = \boldsymbol{\Lambda}_{ij,t} + \epsilon_t \left(\mathbf{D}_{i,t} - \mathbf{D}_{j,t} \right)$$
$$\boldsymbol{\nu}_{ij,t+1} = \boldsymbol{\nu}_{ij,t} + \epsilon_t \left(\mathbf{x}_{i,t} - \mathbf{x}_{j,t} \right)$$

Network Protocol



- Dictionary learning scheme depicted above
  - $\Rightarrow$ model parameters work in same manner
- only exchange local decision variables and Lagrange multipliers

Convergence Result

Theorem

Saddle pt. seq. $(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)$ *converges to stationarity in expectation:*

$$\lim_{t\to\infty} \mathbb{E}[\|\nabla_{\mathbf{D}}\mathcal{L}(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)\|] = 0 \; ,$$

$$\lim_{t\to\infty} \mathbb{E}[\|\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)\|] = 0$$

Asymptotic feasibility condition achieved in expectation:

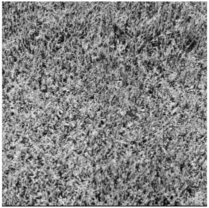
$$\lim_{t\to\infty} \mathbb{E}[\|\nabla_{\mathbf{\Lambda}}\mathcal{L}(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)\|] = 0$$

$$\lim_{t\to\infty} \mathbb{E}[\|\nabla_{\boldsymbol{\nu}}\mathcal{L}(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)\|] = 0$$

- Performance guarentee for D4L
  - $\Rightarrow$ convergence in non-convex stochastic opt.
  - $\Rightarrow$ sensitive to data distribution, step-size, network structure

Image Processing Experiments

- Texture database classification problem $\Rightarrow$ Brodatz textures
  - $\Rightarrow$ Insight into dynamic image processing problems
  - $\Rightarrow$ Toy model of real-time navigability analysis in robotic teams
- Real-time image data $\Rightarrow$ train multi-class logistic regression weights
- Decentralized dynamic texture classification
  - $\Rightarrow$ Subset of textures: {grass, bark, straw, herringbone_weave}



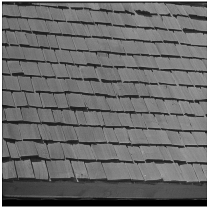


Figure : Sample images from Brodatz textures.

Incomplete Sampling

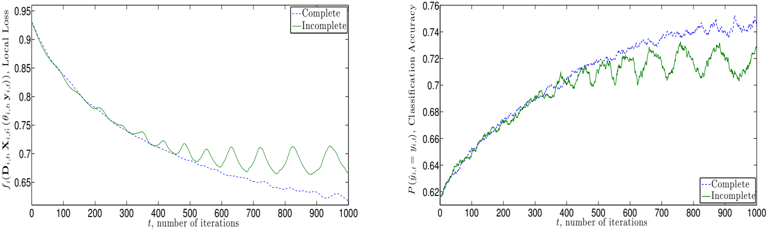


Figure : Log-likelihood (left) and classification accuracy (right) vs. time t.

- $N = 10$ node random network, results shown for random $j \in V$
- Agents observe random incomplete subsets of feature space
- Still learn global information and reach consensus
- Moderate classifier performance
  - $\Rightarrow$ due to small step-size required for convergence
  - $\Rightarrow$ Small step-sizes required for convergence

Robotic Field Setting

- $N = 3$ robotic network of Huskies, sequentially observes images
  $\Rightarrow$ partitions them into small patches $\Rightarrow$ classify patches.
- Robotic network dynamically analyzes navigability of environment
  $\Rightarrow$ Textures correspond to terrains of varying traversability
- Experiments at Lejeune Robotics Test Facility $\Rightarrow$ Thanks to ARL!

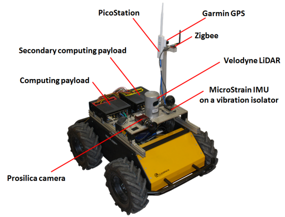


Figure : Sample image (left) from a $N = 3$ robot network of Huskies (right).

Results on Robotic Network

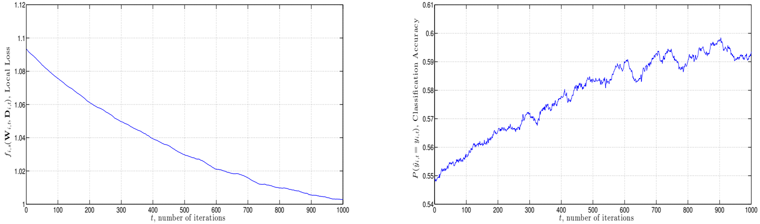


Figure : Log-likelihood (left) and classification accuracy (right) versus time t.

- Experimental setting: $N = 3$ complete graph
- Robotic implementation $\Rightarrow$ promising initial results

Conclusion

- Pattern recognition $\Rightarrow$ finding good signal representation
- Online task-driven dictionaries $\Rightarrow$ visual awareness in robotic teams
- Decentralized non-convex stochastic opt. problem
- Block-variant of saddle pt. method $\Rightarrow$ convergence in expectation
- Implementation on robotic network of Huskies
  $\Rightarrow$ distributed online protocol for gaining environmental awareness

http://seas.upenn.edu/~akoppel/

References

- A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "D4L: Decentralized Dynamic Discrminative Dictionary Learning," in Proc. Int. Conf. Intelligent Robotics and Systems, Hamburg, Germany, Sep 28-Oct2 2015
- A. Koppel, G. Warnell, and E. Stump. "A Stochastic Primal-Dual Algorithm for Task-Driven Dictionary Learning in Networks." in Proc. Asilomar Conf. on Signals Systems Computers, Pacific Grove, CA, November 8-11 2015. (to appear)
- A. Koppel, G. Warnell, E. Stump, and A. Ribeiro, "D4L: Decentralized Dynamic Discrminative Dictionary Learning," IEEE Trans. Signal Process., July. 2015. (submitted).

Stochastic Saddle Point Method

- Decentralized dynamic dictionary learning $\Rightarrow$ Block saddle point alg.
- Stochastic approximation: $\mathcal{L}(\mathbf{D}, \mathbf{w}, \mathbf{\Lambda}, \boldsymbol{\nu}) = \mathbb{E}_{\boldsymbol{\theta}, \mathbf{y}}[\hat{\mathcal{L}}_t(\mathbf{D}, \mathbf{x}, \mathbf{\Lambda}, \boldsymbol{\nu})]$

  $\Rightarrow$ primal stochastic gradient descent

$$\mathbf{D}_{t+1} = \mathbf{D}_t - \epsilon_t \nabla_{\mathbf{D}} \hat{\mathcal{L}}_t(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t) \ ,$$
$$\mathbf{x}_{t+1} = \mathbf{x}_t - \epsilon_t \nabla_{\mathbf{x}} \hat{\mathcal{L}}_t(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t) \ .$$

  $\Rightarrow$ dual stochastic gradient ascent

$$\mathbf{\Lambda}_{t+1} = \mathbf{\Lambda}_t + \epsilon_t \nabla_{\mathbf{\Lambda}} \hat{\mathcal{L}}_t(\mathbf{D}_{t+1}, \mathbf{x}_{t+1}, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t) \ ,$$
$$\boldsymbol{\nu}_{t+1} = \boldsymbol{\nu}_t + \epsilon_t \nabla_{\boldsymbol{\nu}} \hat{\mathcal{L}}_t(\mathbf{D}_{t+1}, \mathbf{x}_{t+1}, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t) \ .$$

- $\nabla_{\mathbf{D}} \hat{\mathcal{L}}_t(\mathbf{D}_t, \mathbf{x}_t, \mathbf{\Lambda}_t, \boldsymbol{\nu}_t)$ $\Rightarrow$ Projected stoch. Lagrangian grad. w.r.t. $\mathbf{D}$

  $\Rightarrow$ gradient approximated with current signals $\{\boldsymbol{\theta}_{i,t}, \mathbf{y}_{i,t}\}_{i=1}^N$

Technical Assumptions

- Network $\mathcal{G}$ $\Rightarrow$ symmetric and connected with diameter D.
- Diminishing step-size rules: $\sum_{t=0}^{\infty} \epsilon_t = \infty$ and $\sum_{t=0}^{\infty} \epsilon_t^2 < \infty$
- Mean and variance conditions of Lagrangian stochastic gradients

$$\mathbb{E}[\|\boldsymbol{\delta}_{\mathbf{D},t}\| \mid \mathcal{F}_t] \leq A\epsilon_t \ ,$$
$$\mathbb{E}[\|\nabla_{\mathbf{D}}\hat{\mathcal{L}}_t(\mathbf{D}_t, \mathbf{x}_t, \boldsymbol{\Lambda}_t, \boldsymbol{\nu}_t)\|^2 \mid \mathcal{F}_t] \leq \sigma^2.$$

- Feasible dictionary set is restricted to those with unit column-norms

$$\mathcal{D} = \{\mathbf{D} \in \mathbb{R}^{m \times k} : \|\mathbf{d}_j\| \leq 1, j = 1 \dots k\}.$$

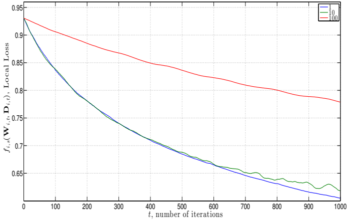
Sparse Multi-class texture classification

- Multi-class logistic regression prob. $\Rightarrow$ Robot i receives signals $\boldsymbol{\theta}_{i,t}$
  $\Rightarrow$ output a decision variable $\mathbf{y}_{i,t} \in \{0,1\}^C$ $\Rightarrow$ C no. of classes
- $[\mathbf{y}_{i,t}]_c$ $\Rightarrow$ binary indicator of whether signal falls in class c.
- Local loss f_i $\Rightarrow$ negative log-likelihood of prob. model

$$f_i(\mathbf{D}_i, \mathbf{X}_i; (\boldsymbol{\theta}_i, \mathbf{y}_i)) = \log\left(\sum_{c=1}^{C} e^{\mathbf{x}_{i,c}^T \boldsymbol{\alpha}_i^* + x_{i,c}^0}\right) - \sum_{c=1}^{C}\left(y_{i,c}\mathbf{x}_{i,c}^T \boldsymbol{\alpha}_i^* + w_{i,c}^0\right) + \xi \|\mathbf{X}_i\|_F^2 \ ,$$

- $\boldsymbol{\alpha}_i^*$ $\Rightarrow$ sparse coding via elastic-net min. prob.
- $g_c(\boldsymbol{\alpha}_i^*) = e^{\mathbf{x}_{i,c}^T \boldsymbol{\alpha}_i^* + x_{i,c}^0}$ is activation function;
  $\Rightarrow$ $g_c(\mathbf{z}_i)/\sum_{c'} g_{c'}(\mathbf{z}_i)$ $\Rightarrow$ prob. $\mathbf{z}_i$ in class c
  $\Rightarrow$ $\mathbf{z}_i$ $\Rightarrow$ average of image sub-patches
- Classification decision $\Rightarrow$ maximum likelihood class label
  $\Rightarrow$ $\tilde{c} = \text{argmax}_c \, g_c(\mathbf{z}_i)/\sum_{c'} g_{c'}(\mathbf{z}_i)$; $[\mathbf{y}_{i,t}]_c = 0$ for $c \neq \tilde{c}$

D4L and Network Size



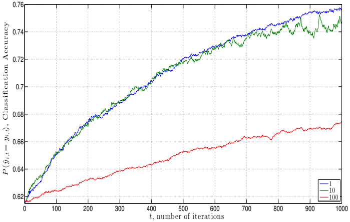


Figure : Local loss (left) and classification accuracy (right) versus iteration t.

- Slower learning in larger networks ($N \uparrow$)
- Non-convexity hurts more in larger-networks

  $\Rightarrow$ Smaller step-sizes required for convergence
- Initialize at stationary point $\Rightarrow$ effective tracking for any N

  $\Rightarrow$ Dist. learning lags. in robotic networks work for *solution tracking*