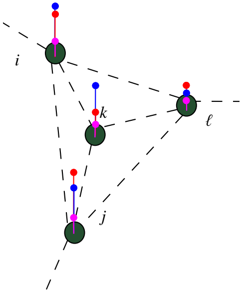
An Adaptive Sampling Technique for Graph Diffusion LMS Algorithm

Daniel Gilio Tiglea, Renato Candido, and Magno T. M. Silva

Polytechnic School - University of São Paulo, Brazil

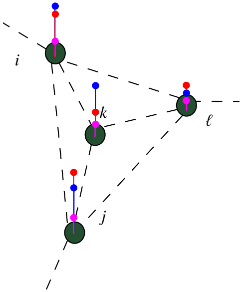
September 4^{th}, 2019

Introduction & Problem Formulation



- $\mathbf{A} \rightarrow\ V \times V$ adjacency matrix
- $\mathbf{x}(n) = [x_1(n),\ \cdots,\ x_k(n),\ \cdots,\ x_V(n)]$
- $\mathbf{v}(n) = [v_1(n),\ \cdots,\ v_k(n),\ \cdots,\ v_V(n)]$
- $\mathbf{y}(n) = [y_1(n),\ \cdots,\ y_k(n),\ \cdots,\ y_V(n)]$

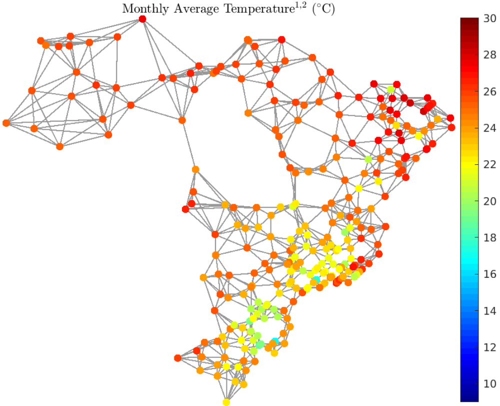
Introduction & Problem Formulation



- $\mathbf{A} \rightarrow \; V \times V$ adjacency matrix
- $\mathbf{x}(n) = [x_1(n), \; \cdots, \; x_k(n), \; \cdots, \; x_V(n)]$
- $\mathbf{v}(n) = [v_1(n), \; \cdots, \; v_k(n), \; \cdots, \; v_V(n)]$
- $\mathbf{y}(n) = [y_1(n), \; \cdots, \; y_k(n), \; \cdots, \; y_V(n)]$

Information spreads from one node to another
Optimal system processes information

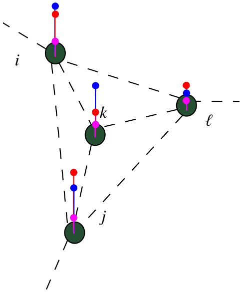
Example: Evolution of Temperature over Time



[1] Instituto Nacional de Meteorologia (INMET), "Normais Climatológicas do Brasil." http://www.inmet.gov.br/portal/index.php?r=clima/normaisClimatologicas.

[2] M. J. M. Spelta, "Brazilian weather stations." https://github.com/mspelta/brazilian-weather-stations#brazilian-weather-stations, 2018.

Problem Formulation



- $\mathbf{A} \rightarrow \ V \times V$ adjacency matrix
- $\mathbf{x}(n) = [x_1(n), \ \cdots, \ x_k(n), \ \cdots, \ x_V(n)]$
- $\mathbf{v}(n) = [v_1(n), \ \cdots, \ v_k(n), \ \cdots, \ v_V(n)]$
- $\mathbf{y}(n) = [y_1(n), \ \cdots, \ y_k(n), \ \cdots, \ y_V(n)]$

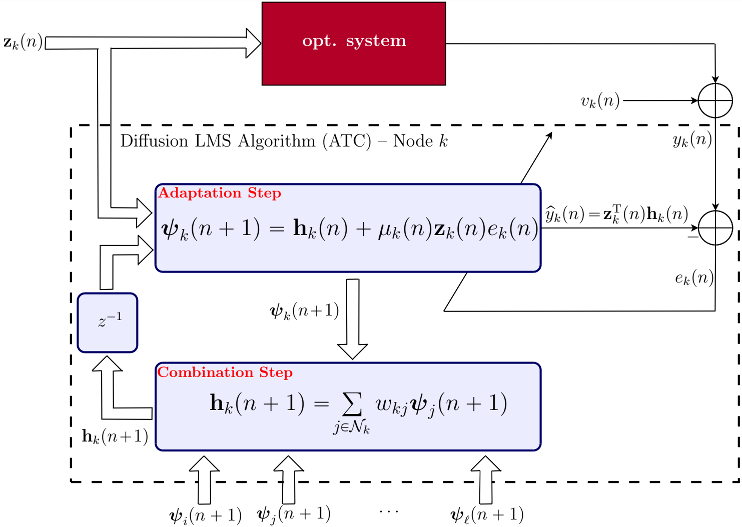
Information spreads from one node to another
Optimal system processes information

$$\mathbf{z}_k(n) \triangleq \mathrm{col}\Big\{[\mathbf{x}(n)]_k, \underbrace{[\mathbf{A}^1\mathbf{x}(n-1)]_k, \ \cdots, [\mathbf{A}^{M-1}\mathbf{x}(n-M+1)]_k}_{\text{information spreading}}\Big\}$$

$$\mathbf{h}^{\mathrm{o}} = [h_0^{\mathrm{o}}, \ \cdots, \ h_{M-1}^{\mathrm{o}}] \rightarrow \text{ opt. system}$$

$$y_k(n) = \mathbf{h}^{\mathrm{o}} \cdot \mathbf{z}_k(n) + v_k(n)$$

dLMS Algorithm[3]



[3] R. Nassif, C. Richard, J. Chen, and A.H. Sayed, "Distributed diffusion adaptation over graph signals," in Proc. IEEE ICASSP, 2018, pp. 4129-4133

Modifying the dLMS algorithm

Modification: introduction of $\bar{s}_k(n) \in \{0,\ 1\}$

$$\begin{cases} \boldsymbol{\psi}_k(n+1) = \mathbf{h}_k(n) + \bar{s}_k(n)\mu_k(n)\mathbf{z}_k(n)e_k(n) \\ \mathbf{h}_k(n+1) = \sum_{j\in\mathcal{N}_k} w_{kj}\boldsymbol{\psi}_j(n+1) \end{cases}$$

If $\bar{s}_k(n) = 0$:

- $y_k(n)$ is not sampled
- $\mu_k(n)$, $\mathbf{z}_k(n)$ and $e_k(n)$ are not computed
- $\boldsymbol{\psi}_k(n+1) = \mathbf{h}_k(n)$

Calculating $\bar{s}_k(n)$

Introducing $s_k(n) \in [0,\ 1]$ such that

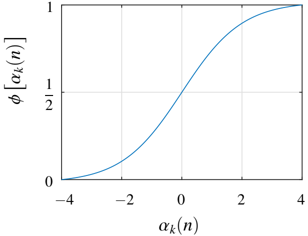
$$\bar{s}_k(n) = \begin{cases} 1, \text{ if } s_k(n) > 0{,}5 \\ 0, \text{ otherwise} \end{cases}$$

$$J_{s,k}(n) = [s_k(n)]\beta s_k(n) + [1 - s_k(n)]\,\frac{1}{|\mathcal{N}_k|}\sum_{i \in \mathcal{N}_k} e_i^2(n)$$

- β: introduced to control how much we penalize sampling
- $\sum e_i^2(n)$ is large: $J_{s,k}(n)$ is minimized by making $s_k(n) \approx 1 \;\rightarrow$ node k is sampled
- $\sum e_i^2(n)$ is small: $J_{s,k}(n)$ is minimized by making $s_k(n) \approx 0 \;\rightarrow$ node k is not sampled

Calculating $s_k(n)$

Auxiliary variable $\alpha_k(n)$ such that $s_k(n) = \phi\left[\alpha_k(n)\right]$



By taking $\dfrac{\partial J_{t,k}(n)}{\partial \alpha_k(n)}$ and applying the gradient method:

$$\alpha_k(n+1) = \alpha_k(n) + \mu_s \phi'\left[\alpha_k(n)\right]\left[\frac{1}{|\mathcal{N}_k|}\sum_{i\in\mathcal{N}_k}\varepsilon_i^2(n) - \beta\bar{s}_k(n)\right]$$

- μ_s: step size
- ε_i: last measurement of e_i

AS-dLMS Algorithm

Choosing β

$$\alpha_k(n+1) = \alpha_k(n) + \mu_s \phi'\left[\alpha_k(n)\right] \left[\frac{1}{|\mathcal{N}_k|} \sum_{i \in \mathcal{N}_k} \varepsilon_i^2(n) - \boxed{\beta}\,\bar{s}_k(n) \right]$$

In order for the sampling to cease in the steady state, $\Delta\alpha_k(n)$ must be negative

Assuming:

- $\phi'\left[\alpha_k(n)\right]$ statistically independent from $e_i(n)$ and $\bar{s}_k(n)$
- $\mathrm{E}\{e_i^2(n)\} \approx \sigma_{v_i}^2$ in steady state

$$\beta > \sigma_{\max}^2 \triangleq \max_{i \in \mathcal{V}} \sigma_{v_i}^2$$

- $\beta \in]\sigma_{\max}^2, 10\sigma_{\max}^2] \rightarrow$ performance preserved

Choosing μ_s

$$\alpha_k(n+1) = \alpha_k(n) + \boxed{\mu_s}\phi'\left[\alpha_k(n)\right]\left[\frac{1}{|\mathcal{N}_k|}\sum_{i\in\mathcal{N}_k}\varepsilon_i^2(n) - \beta\bar{s}_k(n)\right]$$

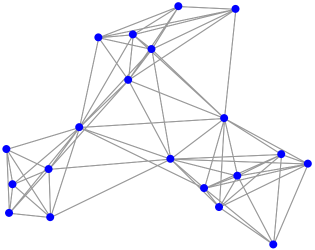
Assuming $\beta > \sigma^2_{\max}$, we wish to choose μ_s such that the sampling ceases in at most Δn iterations after the steady state is achieved

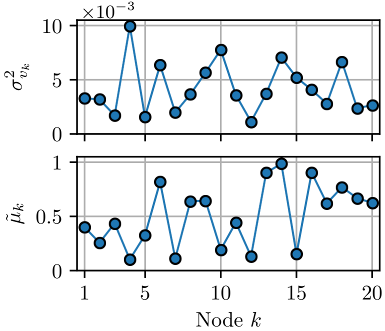
$$\mu_s \gtrapprox \frac{\xi}{\beta - \sigma^2_{\max}}\left[\rho^{1/\Delta n} - 1\right]$$

- ξ and ρ: constants that depend on $\phi[\cdot]$

Simulation Conditions

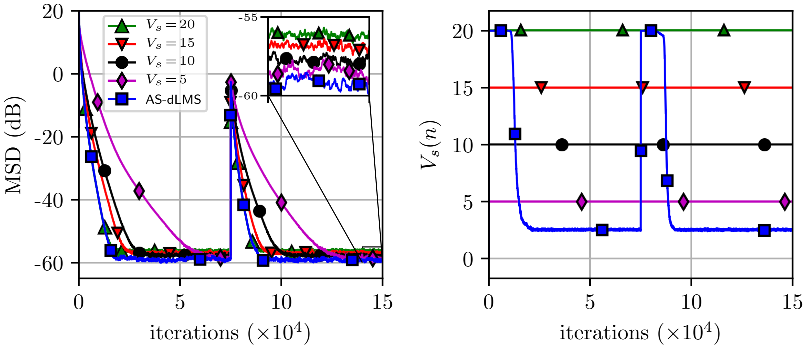
- Randomly generated graphs with 20 nodes
- Different values of $\sigma^2_{v_k}$ and $\tilde{\mu}_k$ for each node k





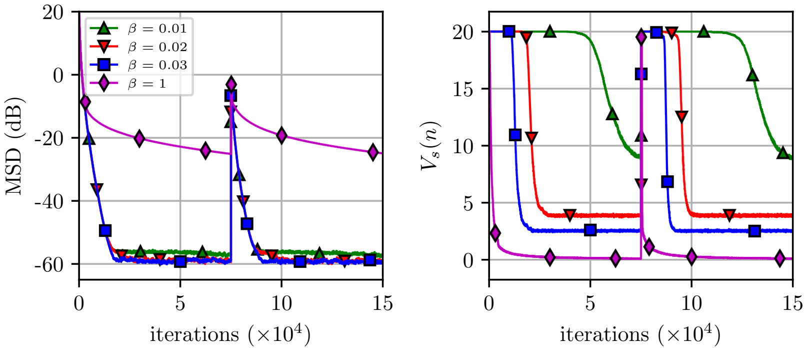
Comparison with random sampling

- Random sampling: V_s nodes chosen randomly every iteration
- AS-dLMS ($\beta = 0.03$ and $\mu_s = 0.22$)
  - Slightly superior steady state performance
  - Same convergence rate as dLMS with all 20 nodes sampled
  - Computational cost: ↑ during transient, ↓↓ during steady state



Different values for β, $\mu_s = 0.22$

$\beta > \sigma^2_{\max} = 0.01$

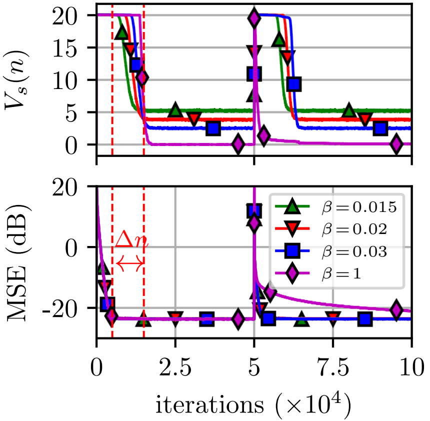


- $\uparrow\ \beta$, $\downarrow$ sampled nodes in steady state
- $\beta > 0.01 = \sigma^2_{\max} \rightarrow$ nodes cease to be sampled
- $\beta = 1 \rightarrow$ poor performance

Testing the adjustment of μ_s ($\Delta n = 10^4$)

$$\mu_s \gtrapprox \frac{\xi}{\beta - \sigma^2_{\max}} \left[\rho^{1/\Delta n} - 1\right]$$

β	μ_s
0.015	0.88
0.02	0.44
0.03	0.22
1	0.0044

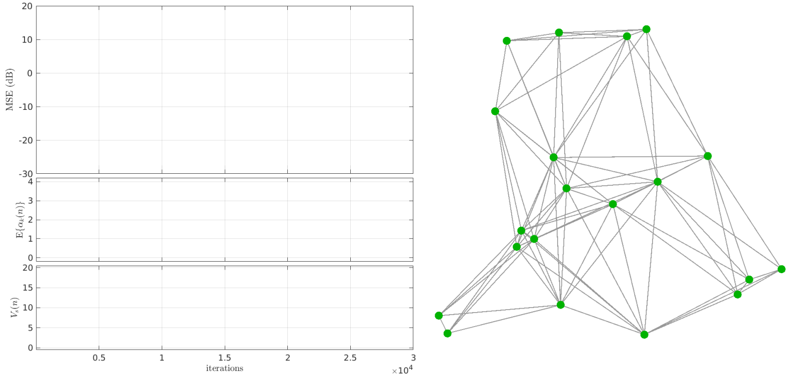


- Nodes cease to be sampled $\approx \Delta n$ iterations after steady state
- $\beta = 1 \rightarrow$ poor performance after abrupt change

Illustrative Example - One Realization

$$\beta = 0.03,\ \Delta n = 5 \cdot 10^3 \rightarrow \mu_s = 0.44$$

•: sampled •: not sampled



Conclusions

- AS-dLMS $\times$ dLMS with all nodes sampled:
  - Slight improvement in steady state performance
  - Same convergence rate
  - Computational cost: $\uparrow$ during transient, $\downarrow\downarrow$ during steady state
- $\uparrow \beta$ $\downarrow$ sampled nodes in steady state
- $\uparrow\uparrow \beta \rightarrow$ poor performance even with proper μ_s
- $\beta \in]\sigma^2_{\max}, 10\sigma^2_{\max}]$
- Theoretical result for $\mu_s \rightarrow$ supported by simulation results

Acknowledgements

Thank you!

Acknowledgements:

- National Council for Scientific and Technological Development (CNPq)
- National Council for the Improvement of Higher Education (CAPES)
- São Paulo Research Foundation (FAPESP)

Choosing μ_s

Assuming $\beta > \sigma^2_{\max}$, how can we choose μ_s such that the sampling ceases in at most Δn iterations after the steady state is achieved?

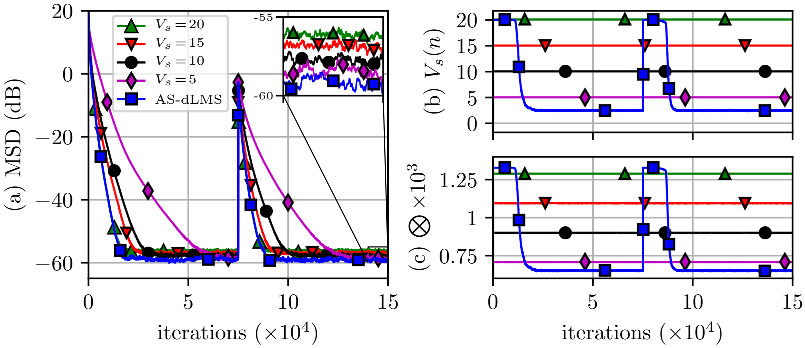
Maintaining previous assumptions & considering a linear approximation for $\phi'\left[\alpha_k(n)\right]$

$$\phi'\left[\alpha_k(n)\right] \approx \rho\alpha_k(n) + \phi'_0,$$

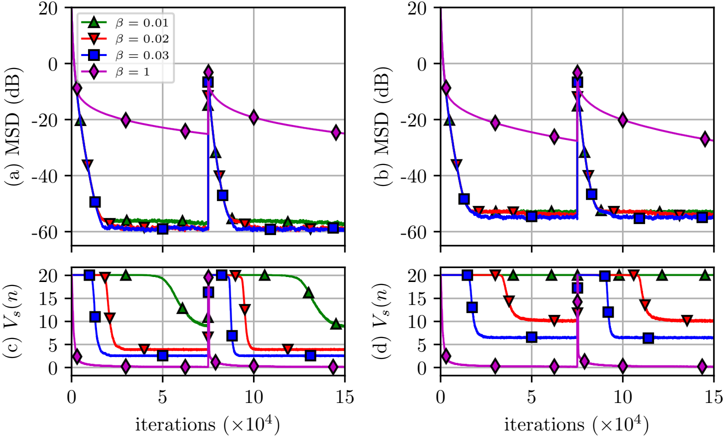
$$\mu_s \gtrapprox \frac{\alpha^+}{(\beta - \sigma^2_{\max})(\phi'_0 - \phi'_{\alpha^+})}\left[\left(\frac{\phi'_0}{\phi'_{\alpha^+}}\right)^{1/\Delta n} - 1\right]$$

Comparison with random sampling

- Random sampling: V_s nodes chosen randomly every iteration
- AS-dLMS ($\beta = 0.03$ and $\mu_s = 0.22$)
  - Slightly superior steady state performance
  - Same convergence rate as dLMS with all 20 nodes sampled
  - Computational cost: ↑ during transient, ↓↓ during steady state



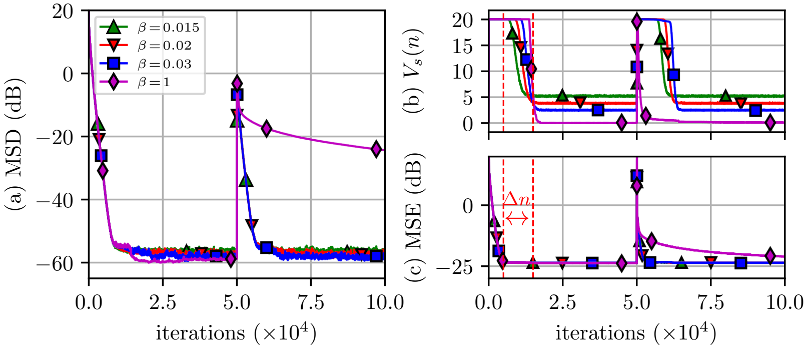
Different values for β, $\mu_s = 0.22$ ($\beta > \sigma^2_{\max}$)



- $\uparrow$ β, $\downarrow$ sampled nodes in steady state
- $\sigma^2_{v_i} = 0{,}01\ \forall\ i \rightarrow \beta = 0{,}01$ all nodes are always sampled
- $\beta > 0{,}01 = \sigma^2_{\max} \rightarrow$ nodes cease to be sampled
- $\beta = 1 \rightarrow$ poor performance

Testing the adjustment of μ_s

$$\mu_s \gtrsim \frac{\xi}{\beta - \sigma^2_{\max}} \left[\rho^{1/\Delta n} - 1 \right]$$



- Nodes cease to be sampled $\approx \Delta n$ iterations after steady state
- $\beta = 1 \rightarrow$ poor performance after abrupt change