AUTOMATICA.IT 2011

convegnoautomaticaitaliana.org

Filippo Zanella, Damiano Varagnolo, Angelo Cenedese, Gianluigi Pillonetto, Luca Schenato

Dipartimento di Ingegneria dell'Informazione – Università di Padova

Newton-Raphson consensus for distributed convex optimization

Distributed optimization and its importance

Problem formulation

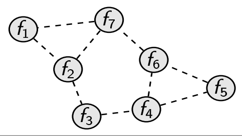
$\min_x \quad f(x) = \sum_{i=1}^N f_i(x)$

subject to $\quad x \in \mathcal{X}$,

$\mathcal{X}, f_i(x)$ are convex

Multi-agents scenario

Networked system where neighbors cooperate to find the optimum



Illustrative example: quadratic local cost functions

Derivation of the algorithm - step 1 on 3

Simplified scalar scenario

$$f_i(x) = \frac{1}{2} a_i (x - b_i)^2 + c_i \qquad a_i > 0$$

Corresponding solution

$$x^* = \frac{\sum_{i=1}^N a_i b_i}{\sum_{i=1}^N a_i} = \frac{\frac{1}{N}\sum_{i=1}^N a_i b_i}{\frac{1}{N}\sum_{i=1}^N a_i}$$

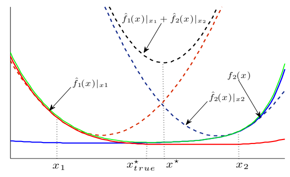
i.e. *parallel of 2 average consensi!*

And for generic convex cost functions?

Derivation of the algorithm - step 2 on 3

- $a_i b_i = f_i''(x_i) x_i - f_i'(x_i)$
- $b_i = f_i''(x_i)$

$$\Rightarrow \quad x^* = \frac{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)x_i - f_i'(x_i)}{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)}$$



Initial proposal

Derivation of the algorithm - step 3 on 3

Algorithm

1. run 2 *average* consensi (*P doubly stochastic*):
   - $y_i(0) := f_i''(x_i)x_i - f_i'(x_i) \qquad y_i(k+1) = P y_i(k)$,
   - $z_i(0) := f_i''(x_i) \qquad z_i(k+1) = P z_i(k)$
2. locally compute $x_i(k) = \dfrac{y_i(k)}{z_i(k)}$

That's all?

No, *must provide 2 little modifications*:

- track the changing $x_i(k)$
- make local estimation step $x_i = \dfrac{y_i}{z_i}$ less aggressive

The complete algorithm

Definitions

- $g_i(x_i(k)) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$
- $h_i(x_i(k)) = f_i''(x_i(k))$
- bold font = vectorization

Initialization

$$\mathbf{x}(k) = \mathbf{y}(k) = \mathbf{z}(k) = \mathbf{g}(\mathbf{x}(-1)) = \mathbf{h}(\mathbf{x}(-1)) = \mathbf{0}$$

Main procedure

$$\begin{cases} \mathbf{y}(k+1) = P^M\left(\mathbf{y}(k) + \mathbf{g}(\mathbf{x}(k)) - \mathbf{g}(\mathbf{x}(k-1))\right) \\ \mathbf{z}(k+1) = P^M\left(\mathbf{z}(k) + \mathbf{h}(\mathbf{x}(k)) - \mathbf{h}(\mathbf{x}(k-1))\right) \\ \mathbf{x}(k+1) = (1-\varepsilon)\mathbf{x}(k) + \varepsilon \dfrac{\mathbf{y}(k+1)}{\mathbf{z}(k+1)} \end{cases}$$

Convergence properties

Proposition

Assume that $f_i \in \mathcal{C}^2$, f_i strictly convex, $x^* \neq \pm\infty$, and *zero initial condition*. There is a positive $\bar{\varepsilon}$ s.t. if $\varepsilon < \bar{\varepsilon}$ then

$$\lim_{k \to +\infty} \mathbf{x}(k) = x^* \mathbb{1}, \quad \text{exponentially}$$

Sketch of the proof

1. transform the algorithm in a continuous-time system
2. recognize the existence of a two-time scales dynamical system
3. analyze separately fast and slow dynamics (standard singular perturbation model analysis approach)

1) transformation in a continuous-time system

$$\begin{cases} \mathbf{y}(k+1) = P^M\left(\mathbf{y}(k) + \mathbf{g}(\mathbf{x}(k)) - \mathbf{g}(\mathbf{x}(k-1))\right) \\ \mathbf{z}(k+1) = P^M\left(\mathbf{z}(k) + \mathbf{h}(\mathbf{x}(k)) - \mathbf{h}(\mathbf{x}(k-1))\right) \\ \mathbf{x}(k+1) = (1-\varepsilon)\mathbf{x}(k) + \varepsilon \dfrac{\mathbf{y}(k+1)}{\mathbf{z}(k+1)} \end{cases}$$

$$\downarrow P = I - K, M = 1$$

$$\begin{cases} \varepsilon\dot{\mathbf{v}}(t) = -\mathbf{v}(t) + \mathbf{g}(\mathbf{x}(t)) \\ \varepsilon\dot{\mathbf{w}}(t) = -\mathbf{w}(t) + \mathbf{h}(\mathbf{x}(t)) \\ \varepsilon\dot{\mathbf{y}}(t) = -K\mathbf{y}(t) + (I-K)\left[\mathbf{g}(\mathbf{x}(t)) - \mathbf{v}(t)\right] \\ \varepsilon\dot{\mathbf{z}}(t) = -K\mathbf{z}(t) + (I-K)\left[\mathbf{h}(\mathbf{x}(t)) - \mathbf{w}(t)\right] \\ \dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \dfrac{\mathbf{y}(t)}{\mathbf{z}(t)} \end{cases}$$

2) two-time scales dynamical system

$$\begin{cases} \varepsilon\dot{\mathbf{v}}(t) = -\mathbf{v}(t) + \mathbf{g}(\mathbf{x}(t)) \\ \varepsilon\dot{\mathbf{w}}(t) = -\mathbf{w}(t) + \mathbf{h}(\mathbf{x}(t)) \\ \varepsilon\dot{\mathbf{y}}(t) = -K\mathbf{y}(t) + (I-K)\left[\mathbf{g}(\mathbf{x}(t)) - \mathbf{v}(t)\right] \\ \varepsilon\dot{\mathbf{z}}(t) = -K\mathbf{z}(t) + (I-K)\left[\mathbf{h}(\mathbf{x}(t)) - \mathbf{w}(t)\right] \\ \hline \dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \dfrac{\mathbf{y}(t)}{\mathbf{z}(t)} \end{cases}$$

If ε is sufficiently small ...

- first subsystem is much faster than second one
- first subsystem is globally exponentially stable

3) Fast Dynamics ($\epsilon \to 0$).

$$\mathbb{1}^T\dot{\mathbf{y}}(t) = \mathbb{1}^T\dot{\mathbf{v}}(t) \Longrightarrow y_{ave}(t) = v_{ave}(t) + y_{ave}(0) - v_{ave}(0), \quad \left(y_{ave} = \tfrac{1}{N}\mathbb{1}^T\mathbf{y}\right)$$

$$\mathbb{1}^T\dot{\mathbf{z}}(t) = \mathbb{1}^T\dot{\mathbf{w}}(t) \Longrightarrow z_{ave}(t) = z_{ave}(t) + w_{ave}(0) - w_{ave}(0), \quad \textit{true } \forall\epsilon$$

$$\begin{cases} \mathbf{v}(t) \to \mathbf{g}(\mathbf{x}(t)) & \text{fast dynamics} \\ \mathbf{w}(t) \to \mathbf{h}(\mathbf{x}(t)) & \\ \dot{\mathbf{y}}(t) \to \left(\frac{1}{N}\mathbb{1}^T\mathbf{g}(\mathbf{x}(t))\right)\mathbb{1} \quad \text{if } y_{ave}(0) = v_{ave}(0) & \\ \dot{\mathbf{z}}(t) \to \left(\frac{1}{N}\mathbb{1}^T\mathbf{h}(\mathbf{x}(t))\right)\mathbb{1} \quad \text{if } z_{ave}(0) = w_{ave}(0) & \\ \hline \dot{\mathbf{x}}(t) = -\mathbf{x}(t) + \dfrac{\frac{1}{N}\mathbb{1}^T\mathbf{g}(\mathbf{x}(t))}{\frac{1}{N}\mathbb{1}^T\mathbf{h}(\mathbf{x}(t))}\mathbb{1}, \Longrightarrow \mathbf{x}(t) \to x_{ave}(t)\mathbb{1} & \text{slow dyn.} \end{cases}$$

If ε is sufficiently small ...

- first subsystem is much faster than second one
- first subsystem is globally exponentially stable

3) Slow dynamics

If ε is sufficiently small ...

$$\dot{x}_{ave} \approx -\frac{f'_{ave}(x_{ave})}{f''_{ave}(x_{ave})} = -\frac{\sum_{i=1}^N f_i'(x_{ave})}{\sum_{i=1}^N f_i''(x_{ave})}$$

i.e. a *continuous-time Newton-Raphson strategy*

Other important properties

- do not require topological knowledge / agents synchronization
- robust to numerical error and communication noise

Robustness properties

Proposition

Assume that $f_i \in \mathcal{C}^2$, f_i strictly convex, $x^* \neq \pm\infty$, and

$$||\mathbf{x}(0) - x^*\mathbb{1}|| = \rho$$
$$|\mathbb{1}^T(\mathbf{v}(0) - \mathbf{y}(0))| = \alpha$$
$$|\mathbb{1}^T(\mathbf{v}(0) - \mathbf{y}(0))| = \beta$$

There is a positive scalars $\bar{\varepsilon}, \bar{\rho}, \bar{\alpha}, \bar{\beta}$ and $\phi(\alpha,\beta) : \mathbb{R}^2 \to \mathbb{R} \in \mathcal{C}^0$, s.t. if $\varepsilon < \bar{\varepsilon}, \rho < \bar{\rho}, \alpha < \bar{\alpha}, \beta < \bar{\beta}$ then

$$\lim_{k \to +\infty} \mathbf{x}(k) = \phi(\alpha,\beta)\mathbb{1}, \quad \text{exponentially}$$

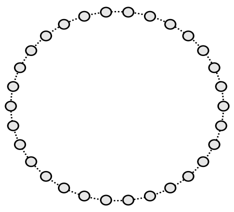
and $\phi(0,0) = x^*$

Experiments description

- circulant graph, $N = 30$

- $$P = \begin{bmatrix} 0.5 & 0.25 & & & 0.25 \\ 0.25 & 0.5 & 0.25 & & \\ & \ddots & \ddots & \ddots & \\ & & 0.25 & 0.5 & 0.25 \\ 0.25 & & & 0.25 & 0.5 \end{bmatrix}$$

- f_i = sum of exponentials

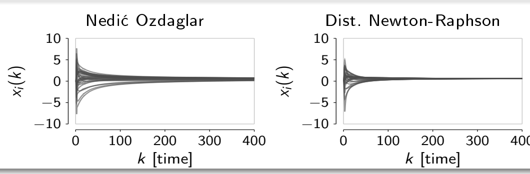


Comparisons with a Distributed Subgradient

Algorithm from Nedić Ozdaglar, *Distributed subgradient methods for multi-agent optimization* (2009)

1. $\mathbf{x}^{(c)}(k) = P\mathbf{x}(k)$ (consensus step)
2. $x_i(k+1) = x_i^{(c)}(k) - \frac{\rho}{k} f_i'\left(x_i^{(c)}(k)\right)$ (local gradient descent)

Numerical comparison

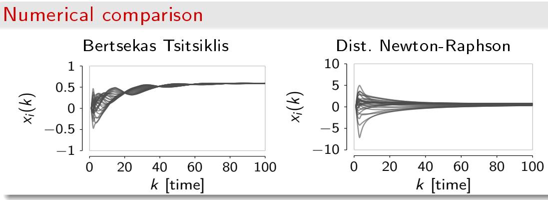


Comparisons with an ADMM (first-order)

Algorithm from Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation* (1997)

$$L_\rho := \sum_i \Big[f_i(x_i) + y_i^{(\ell)}(x_i - z_{i-1}) + y_i^{(c)}(x_i - z_i) + y_i^{(r)}(x_i - z_{i+1}) + \tfrac{\delta}{2}|x_i - z_{i-1}|^2 + \tfrac{\delta}{2}|x_i - z_i|^2 + \tfrac{\delta}{2}|x_i - z_{i+1}|^2 \Big]$$

Numerical comparison



Conclusions and bibliography

Conclusions

- combines Newton-Raphson-like behaviors with *average-consensi*
- converges to global optimum (convexity and smoothness assumptions)
- does not require network topology knowledge
- minimal agents synchronization (symmetric gossip like)
- extremely simple to be implemented
- numerically faster than Subgradient methods but slower than Alternating Direction Method of multipliers

Bibliography

Zanella et al., *Newton-Raphson consensus for distributed convex optimization*, CDC 2011

Zanella et al., *Multidimensional Newton-Raphson consensus for distributed convex optimization*, ACCC 2012 (submitted)