UNIVERSITÀ DEGLI STUDI DI PADOVA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

Multi-agent distributed optimization and estimation over lossy networks

Nicoletta Bof

22 February 2017

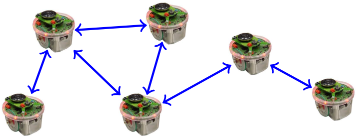
Advisor: Prof. Luca Schenato

Co-advisor: Prof. Ruggero Carli

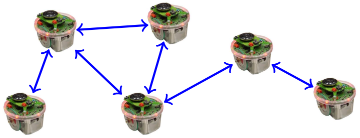
Outline

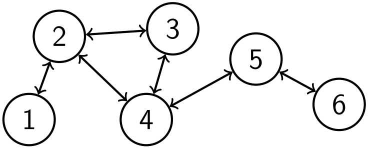
Current section

Multi-agent systems & optimization



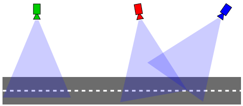
Multi-agent systems & optimization



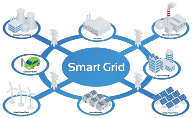


Multi-agent systems & optimization

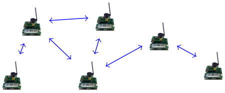
Camera network



Smart grid



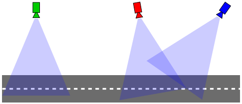
Wireless sensor network



Multi-robot

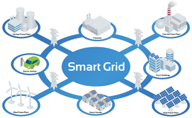
Multi-agent systems & optimization

Camera network



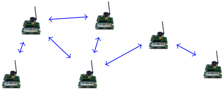
Coordination

Smart grid



State estimation

Wireless sensor network

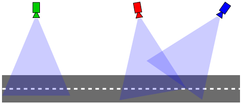
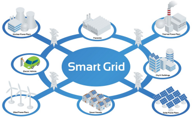
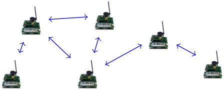


Consensus

Multi-robot

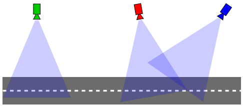
Map building

Multi-agent systems & optimization

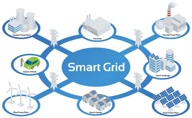
Camera network		Coordination	$\rightarrow f(x)$
Smart grid		State estimation	$\rightarrow f(x)$
Wireless sensor network		Consensus	$\rightarrow f(x)$
Multi-robot		Map building	$\rightarrow f(x)$

Multi-agent systems & optimization

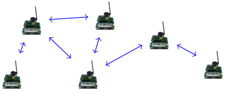
Camera network



Smart grid



Wireless sensor network

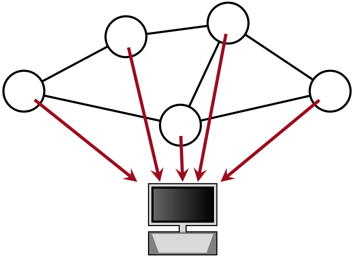


Multi-robot

Solvable using optimization

$\min_{x \in X} f(x)$

Centralized VS distributed

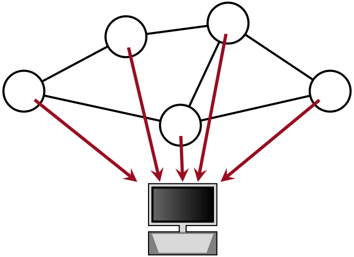


Centralized algorithm

+ No computational or memory constraints
+ No communication issues

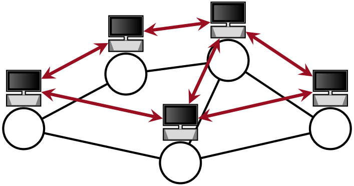
– Does not scale well with N
– Expensive
– No privacy

Centralized VS distributed



Centralized algorithm

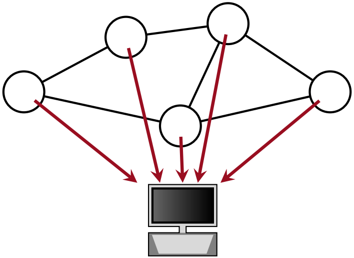
- \+ No computational
  or memory constraints
- \+ No communication issues
- – Does not scale well with N
- – Expensive
- – No privacy



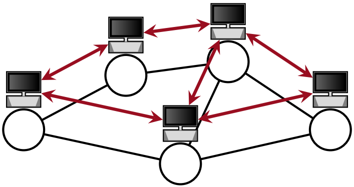
Distributed algorithm

- \+ No need of central unit
- \+ Scale well with N
- \+ Cheaper
- – Computational and
  memory constraints
- – Communication issues

Centralized VS distributed



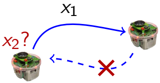
Centralized algorithm



Distributed algorithm

Real-world communication challenges

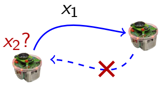
Unreliability



Witsenhausen counterexample (1968)

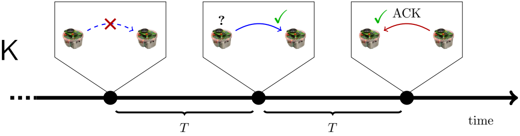
Real-world communication challenges

Unreliability

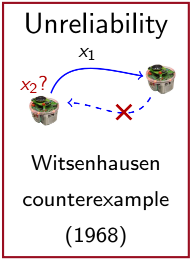


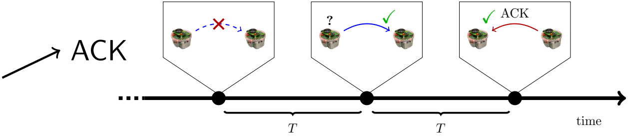
Witsenhausen counterexample (1968)

→ ACK



Real-world communication challenges

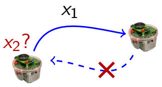




Packet loss resilient algorithm (robust)

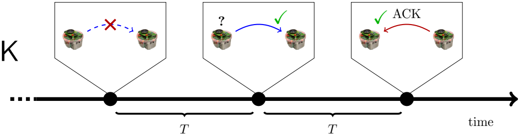
Real-world communication challenges

Unreliability



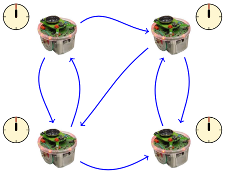
Witsenhausen counterexample (1968)

→ ACK



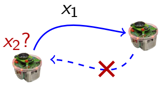
→ Packet loss resilient algorithm (robust)

Synchronicity



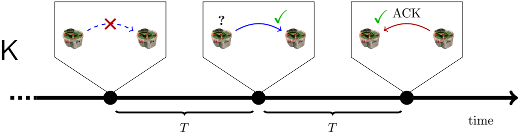
Real-world communication challenges

Unreliability



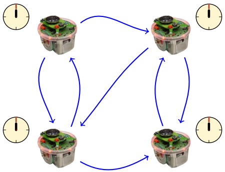
Witsenhausen counterexample (1968)

→ ACK

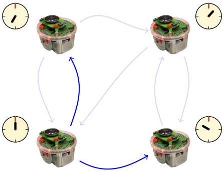


→ Packet loss resilient algorithm (robust)

Synchronicity



→



Asynchronous protocol
(Asymmetric broadcast)

History and state of the art - Distributed algorithms

'80s: Parallel and Distributed Computation [Bertsekas '89]

History and state of the art - Distributed algorithms

'80s: Parallel and Distributed Computation [Bertsekas '89]

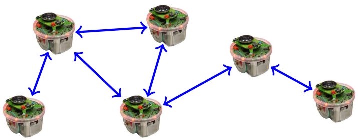
Big data

- Computational parallelization (star like)
- Reliable communication (internet)

History and state of the art - Distributed algorithms

'80s: Parallel and Distributed Computation [Bertsekas '89]

Multi-agent systems



Big data

- intrinsic peer-to-peer architecure
- unrealiable communication (wireless)

- Computational parallelization (star like)
- Reliable communication (internet)

State of the art - Peer to peer architecture

Subgradient based $\longrightarrow$ DSM [Nedic '09, ...]

State of the art - Peer to peer architecture

Subgradient based $\longrightarrow$ DSM [Nedic '09, ...]

Lagrangian based $\longrightarrow$ ADMM [Boyd '11, ...]

State of the art - Peer to peer architecture

Subgradient based $\longrightarrow$ DSM [Nedic '09, ...]

Lagrangian based $\longrightarrow$ ADMM [Boyd '11, ...]

Consensus based $\nearrow$ Newton-Raphson consensus [Zanella '11,...]

Consensus based $\searrow$ Push-DIGing [Nedic '16, ...]

State of the art - Peer to peer architecture

Subgradient based $\longrightarrow$ DSM [Nedic '09, ...]

Lagrangian based $\longrightarrow$ ADMM [Boyd '11, ...]

Consensus based
- $\nearrow$ Newton-Raphson consensus [Zanella '11,...]
- $\searrow$ Push-DIGing [Nedic '16, ...]

Can work with asynchronous protocols but need reliable communication

Only PDMM can deal with packet losses [Sherson July '17]

Current section

Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol

↓ Patrolling

↓ Partition based optimization

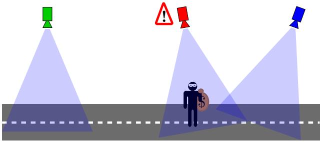
↓ Consensus ↓ Newton-Rapshon Consensus

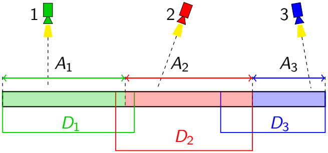
Contributions

Distributed algorithms which are ROBUST and employ an ASYNCHRONOUS communication protocol

↓

Patrolling





D_i: segment which can be patrolled by camera i

A_i: segment assigned to camera i

Asynchronous+non robust algorithm $\xrightarrow{\text{Algorithm modification}}$ Asynchronous+robust algorithm

Non strictly decreasing Lyapunov function + continuity argument

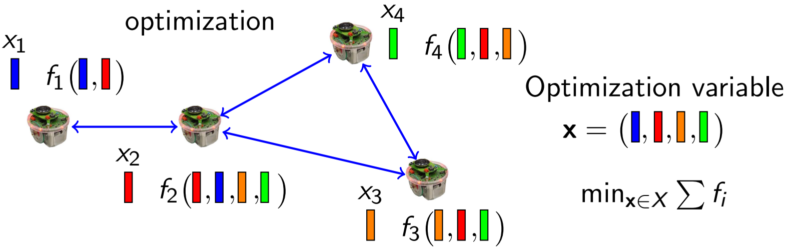
[Bof, Carli, Cenedese & Schenato, IEEE TAC, 2017]

Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol

↓

Partition based optimization



Synchronous+non robust algorithm $\xrightarrow{\text{Use of memory}}$ Asynchronous+robust +only local exchange

Discrete-time separation of time scale

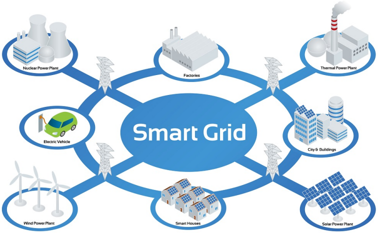
[Todescato, Bof, Cavraro, Carli & Schenato, ArXiv, 2017]

Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol

↓

Partition based optimization

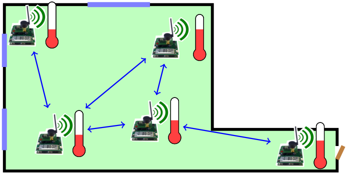


Voltage estimation from voltage and current measurements with outliers

[Todescato, Bof, Cavraro, Carli & Schenato, ArXiv, 2017]

Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol



↓
Consensus

Comparison of convergence rate Consensus based and Lagrangian based algorithms (ADMM and dual ascent)

Root locus analysis+regular graphs

Standard Average Consensus —Additional variables→ Robust & asynchronous Consensus (ra-AC)

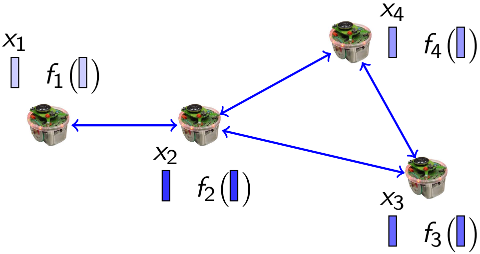
Ergodicity theory

[Bof, Carli & Schenato, Automatica, 2018], [Bof, Carli & Schenato, IFAC, 2017]

Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol

↳ Consensus ⟶ Newton-Rapshon Consensus



Optimization variable

$\mathbf{x} =$

$\min_{\mathbf{x} \in X} \sum f_i$

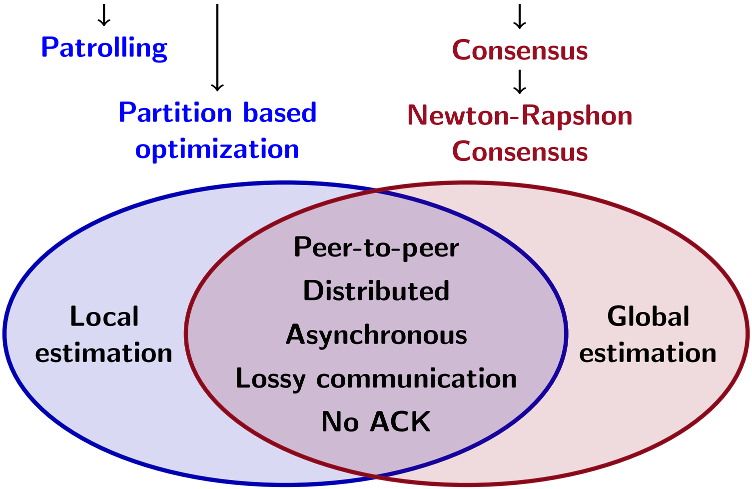
Synchronous+non robust algorithm $\xrightarrow[\text{NRC + ra-AC}]{\text{Merging of algorithms}}$ Asynchronous+robust algorithm

Discrete-time separation of time scale+non convergent variables

[Bof, Carli, Notarstefano, Schenato & Varagnolo, submitted]

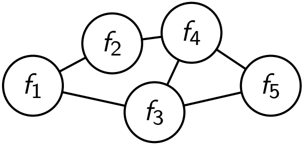
Contributions

Distributed algorithms which are **ROBUST** and employ an **ASYNCHRONOUS** communication protocol



Current section

Robust and asynchronous Newton-Raphson Consensus



Solve in a **distributed** way

$$\min_{x \in X} \sum f_i(x)$$

assuming an **unreliable** communication scenario and using an **asynchronous** algorithm

Newton Raphson (NR) algorithm

Aim: iteratively find optimizer of convex function $f(x)$

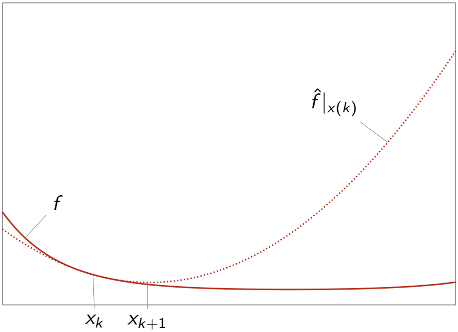
$$x(k+1) = x(k) - \frac{f'(x(k))}{f''(x(k))}$$

Newton Raphson (NR) algorithm

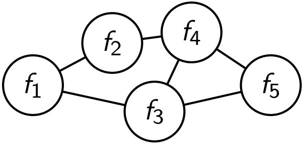
Aim: iteratively find optimizer of convex function $f(x)$

$$x(k+1) = x(k) - \frac{f'(x(k))}{f''(x(k))}$$

Approximate $f(x)$ with $\hat{f}|_{x(k)}$, its quadratic approximation at the current point $x(k)$ and move to its optimizer

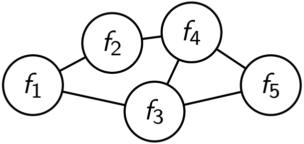


Intuition for Newton Raphson Consensus



$f(x) = \sum_{j=1}^{N} f_j(x)$

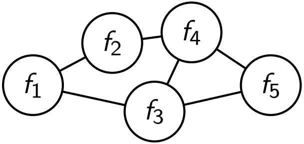
Intuition for Newton Raphson Consensus



$$f(x) = \sum_{j=1}^{N} f_j(x)$$

- Centralized: $x(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))x(k) - f_j'(x(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))}$

Intuition for Newton Raphson Consensus

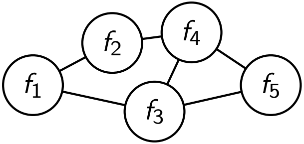


$$f(x) = \sum_{j=1}^{N} f_j(x)$$

- Centralized: $x(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))x(k) - f_j'(x(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))}$
- Distributed: agent i approximates its own $f_i(x)$ at its current $x_i(k)$ and selects $x_i(k+1)$ as the minimizer of $\sum_{i=1}^{N} \hat{f}_i|_{x_i(k)}(x)$

$$x_i(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))x_j(k)f_j'(x_j(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))}$$

Intuition for Newton Raphson Consensus



$$f(x) = \sum_{j=1}^{N} f_j(x)$$

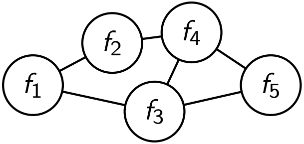
- Centralized: $x(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))x(k) - f_j'(x(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))}$

- Distributed: agent i approximates its own $f_i(x)$ at its current $x_i(k)$ and selects $x_i(k+1)$ as the minimizer of $\sum_{i=1}^{N} \hat{f}_i|_{x_i(k)}(x)$

$$x_i(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))x_j(k) f_j'(x_j(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))}$$

  - If $x_1(k) = \ldots = x_N(k) = x(k)$ $\rightarrow$ traditional NR step
  - If $x_i(k)$s close to each other $\rightarrow$ good approximation
  - If $x_i(k)$s different $\rightarrow$ non correct NR step
  - Step requires exact average $\rightarrow$ time consuming

Intuition for Newton Raphson Consensus



$$f(x) = \sum_{j=1}^{N} f_j(x)$$

- Centralized: $x(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))x(k) - f_j'(x(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x(k))}$

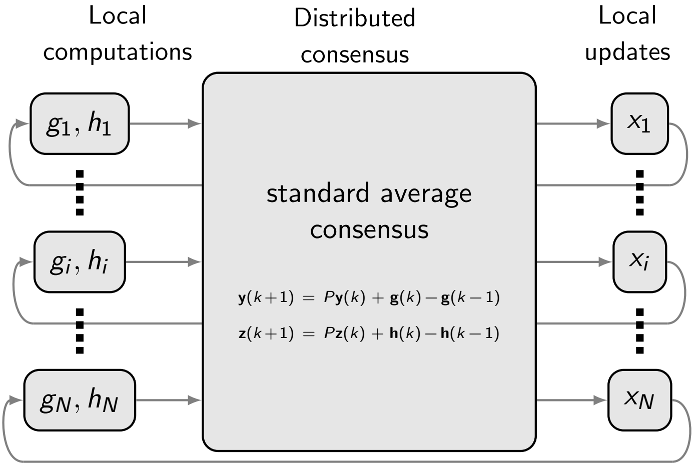
- Distributed: agent i approximates its own $f_i(x)$ at its current $x_i(k)$ and selects $x_i(k+1)$ as the minimizer of $\sum_{i=1}^{N} \hat{f}_i|_{x_i(k)}(x)$

$$x_i(k+1) = \frac{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))x_j(k) f_j'(x_j(k))}{\frac{1}{N}\sum_{j=1}^{N} f_j''(x_j(k))} := \frac{\frac{1}{N}\sum_{j=1}^{N} g_j(k)}{\frac{1}{N}\sum_{j=1}^{N} h_j(k)}$$

Alternate between

- consensus step on g_is and h_is
- smoothed update of x_is

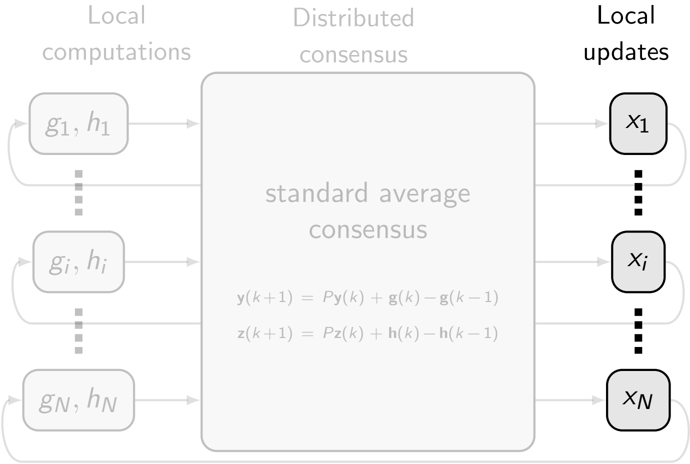
Block schematic representation - NRC



$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

Block schematic representation - NRC

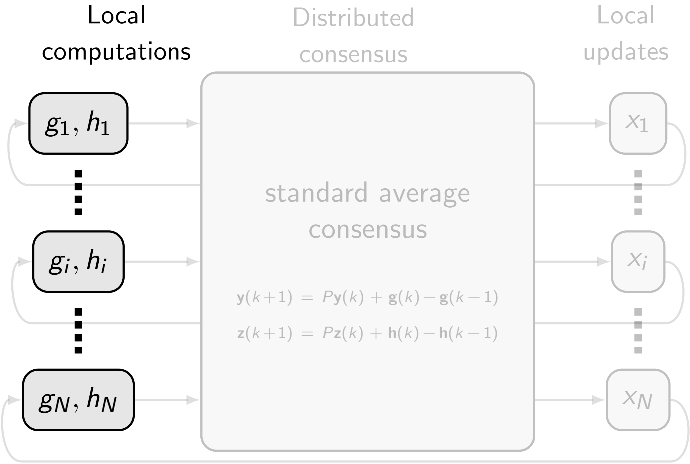


$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{y_i(k+1)}{z_i(k+1)}$$

Direction might not be good

Block schematic representation - NRC

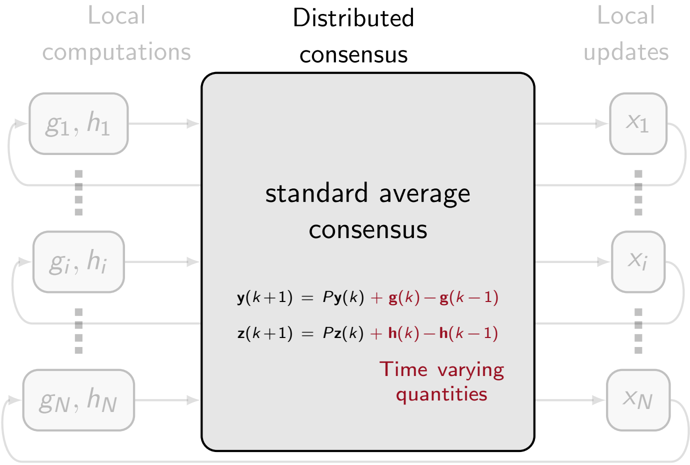


$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

Vary over time

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{y_i(k+1)}{z_i(k+1)}$$

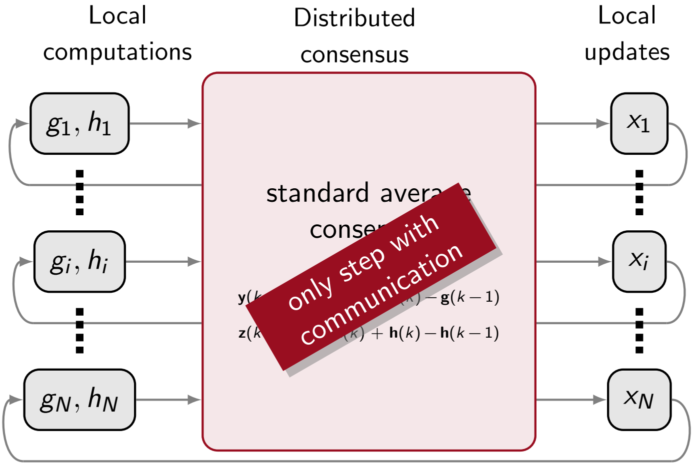
Block schematic representation - NRC



$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

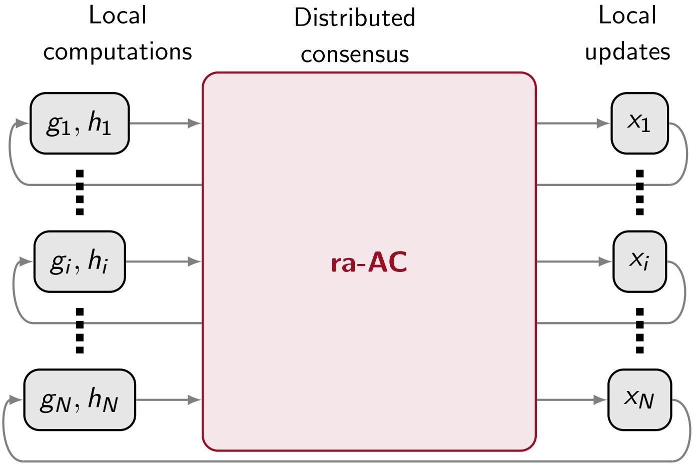
Block schematic representation - NRC



$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

Robust and Asynchronous NRC



$$g_i(k) = f_i''(x_i(k))x_i(k) - f_i'(x_i(k))$$
$$h_i(k) = f_i''(x_i(k))$$

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

Convergence properties of ra-NRC

Assumptions

- strongly convex functions $f_i(x)$
- fixed, strongly connected and directed network
- persistent communications, bounded packet losses

Proposition

The robust and asynchronous Newton-Raphson Consensus is locally exponentially convergent to the minimizer of $f(x) = \sum f_i(x)$.

Convergence properties of ra-NRC

Assumptions

- strongly convex functions $f_i(x)$
- fixed, strongly connected and directed network
- persistent communications, bounded packet losses

Proposition

The robust and asynchronous Newton-Raphson Consensus is locally exponentially convergent to the minimizer of $f(x) = \sum f_i(x)$.

- time scale separation in discrete time with time varying system
- some state variables do not converge

Currently working to extend the result to semi-global convergence

House regression problem

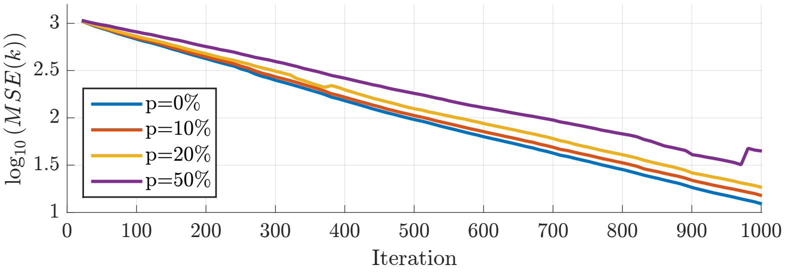
$$f_i(\mathbf{x}) = \sum_{j\in\mathcal{F}_i} \frac{\left(y_j - \chi_j^T \mathbf{x}' - \mathbf{x_0}\right)^2}{\left|y_j - \chi_j^T \mathbf{x}' - \mathbf{x_0}\right| + \beta} + \gamma \left\|\mathbf{x}'\right\|_2^2$$

506 (house features,house value) examples divided among N=10 agents

`http://archive.ics.uci.edu/ml/datasets/Housing`

$$MSE(k) = \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{x}_i(k) - \mathbf{x}^*\right\|^2$$

ra-NRC for different packet loss probability



House regression problem

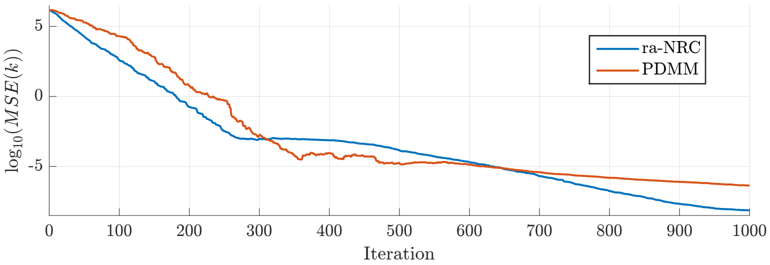
$$f_i(\mathbf{x}) = \sum_{j\in\mathcal{F}_i} \frac{\left(y_j - \chi_j^T \mathbf{x}' - \mathbf{x_0}\right)^2}{\left|y_j - \chi_j^T \mathbf{x}' - \mathbf{x_0}\right| + \beta} + \gamma \left\|\mathbf{x}'\right\|_2^2$$

506 (house features,house value) examples divided among N=10 agents

http://archive.ics.uci.edu/ml/datasets/Housing

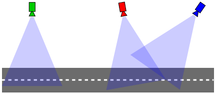
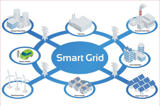
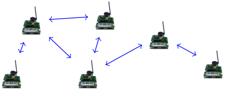
$$MSE(k) = \frac{1}{N}\sum_{i=1}^{N} \left\|\mathbf{x}_i(k) - \mathbf{x}^*\right\|^2$$

Comparison ra-NRC and PDMM for $p = 20\%$



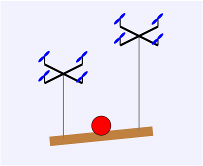
Current section

Conclusions

	Approach	Theory used
	algorithm modification	Lyapunov Continuity argument
	Use of memory	Discrete-time separation of time scale
	Additional variables	Ergodicity theory
	Merging of algorithms	Discrete-time separation of time scale

Future research

- Step size selection
- Density estimation → positivity constraints
- Dynamic optimization: minimize $f(x, t)$
- Optimal control of dynamic systems (MPC)



References I

- N. Bof, R. Carli, and L. Schenato, "On the performance of consensus based versus lagrangian based algorithms for quadratic cost functions," in *Control Conference (ECC), 2016 European*. IEEE, 2016, pp. 160–165.
- N. Bof, M. Todescato, R. Carli, and L. Schenato, "Robust distributed estimation for localization in lossy sensor networks," *IFAC-PapersOnLine*, vol. 49, no. 22, pp. 250–255, 2016.
- N. Bof, R. Carli, A. Cenedese, and L. Schenato, "Asynchronous distributed camera network patrolling under unreliable communication," *IEEE Transactions on Automatic Control*, 2017.
- N. Bof, R. Carli, and L. Schenato, "Average consensus with asynchronous updates and unreliable communication," 2017.
- M. Todescato, N. Bof, G. Cavraro, R. Carli, and L. Schenato, "Generalized gradient optimization over lossy networks for partition-based estimation," *ArXiv*.

References II

N. Bof, R. Carli, G. Notarstefano, L. Schenato, and D. Varagnolo, "Newton-raphson consensus under asynchronous and lossy communications for peer-to-peer networks," *Submitted*.

N. Bof, R. Carli, and L. Schenato, "Is ADMM always faster than average consensus?" *Automatica*, 2018.

Thanks for your attention

Current section

Distributed algorithms: motivations and challenges

Contributions

Distributed optimization with RA-NRC

Conclusions

Appendix

Consensus algorithm for asynchronous protocols

$y_1, \ldots, y_N$ private quantities ($\mathbf{y}$) $\rightarrow x^* = \frac{y_1 + \cdots + y_N}{N}$

Asymmetric broadcast $\rightarrow$ at time k one agent transmits

Consensus algorithm for asynchronous protocols

$y_1, \ldots, y_N$ private quantities $(\mathbf{y}) \rightarrow x^* = \frac{y_1 + \cdots + y_N}{N}$

Asymmetric broadcast $\rightarrow$ at time k one agent transmits

Ratio Consensus [Bénézit et al. 2010]

$$x_i(k) = \frac{s_i(k)}{w_i(k)}, \qquad \begin{array}{ll} \mathbf{s}(k+1) = P(k)\mathbf{s}(k), & \mathbf{s}(0) = \mathbf{y} \\ \mathbf{w}(k+1) = P(k)\mathbf{w}(k), & \mathbf{w}(0) = \mathbb{1}_N \end{array}$$

i active agent $\Rightarrow P(k) = \begin{bmatrix} 1 & 0 & & P_{1i} & & 0 \\ 0 & 1 & & P_{2i} & & 0 \\ \vdots & \vdots & \ldots & \vdots & \ldots & \vdots \\ 0 & 0 & & P_{(N-1)i} & & 0 \\ 0 & 0 & & P_{Ni} & & 1 \end{bmatrix}, \; P(k)\mathbb{1}_N = 1_N$

Consensus algorithm for asynchronous protocols

$y_1, \dots, y_N$ private quantities $(\mathbf{y}) \to x^* = \frac{y_1+\dots+y_N}{N}$

Asymmetric broadcast $\to$ at time k one agent transmits

Ratio Consensus [Bénézit et al. 2010]

$$x_i(k) = \frac{s_i(k)}{w_i(k)}, \qquad \begin{array}{ll} \mathbf{s}(k+1) = P(k)\mathbf{s}(k), & \mathbf{s}(0) = \mathbf{y} \\ \mathbf{w}(k+1) = P(k)\mathbf{w}(k), & \mathbf{w}(0) = \mathbb{1}_N \end{array}$$

i active agent $\Rightarrow P(k) = \begin{bmatrix} 1 & 0 & & P_{1i} & & 0 \\ 0 & 1 & & P_{2i} & & 0 \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ 0 & 0 & & P_{(N-1)i} & & 0 \\ 0 & 0 & & P_{Ni} & & 1 \end{bmatrix}, \quad P(k)\mathbb{1}_N = 1_N$

Ergodicity theory $\to$ $\mathbf{x}(k) \to x^* \mathbb{1}_N$ as $k \to \infty$

Consensus algorithm for asynchronous protocols

$y_1, \ldots, y_N$ private quantities $(\mathbf{y}) \rightarrow x^* = \frac{y_1 + \cdots + y_N}{N}$

Asymmetric broadcast $\rightarrow$ at time k one agent transmits

Ratio Consensus [Bénézit et al. 2010]

$$x_i(k) = \frac{s_i(k)}{w_i(k)}, \qquad \begin{array}{ll} \mathbf{s}(k+1) = P(k)\mathbf{s}(k), & \mathbf{s}(0) = \mathbf{y} \\ \mathbf{w}(k+1) = P(k)\mathbf{w}(k), & \mathbf{w}(0) = \mathbb{1}_N \end{array}$$

$$i \text{ active agent} \Rightarrow P(k) = \begin{bmatrix} 1 & 0 & & P_{1i} & & 0 \\ 0 & 1 & & P_{2i} & & 0 \\ \vdots & \vdots & \ldots & \vdots & \ldots & \vdots \\ 0 & 0 & & P_{(N-1)i} & & 0 \\ 0 & 0 & & P_{Ni} & & 1 \end{bmatrix}, \quad P(k)\mathbb{1}_N = 1_N$$

n_i: numbers of out neighbours of agent i

$P_{ji} = \frac{1}{n_i}$ if i communicates with $j \rightarrow i$ sends $r_i(k) = \frac{1}{n_i} s_i(k)$

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with “mass” counters (synch.)

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with "mass" counters (synch.)

$$\mathbf{x}(k) = \frac{\mathbf{s}(k)}{\mathbf{w}(k)}, \quad \mathbf{s}(0) = \mathbf{y}, \quad \mathbf{w}(0) = \mathbb{1}_N$$

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with “mass” counters (synch.)

$$\mathbf{x}(k) = \frac{\mathbf{s}(k)}{\mathbf{w}(k)}, \quad \mathbf{s}(0) = \mathbf{y}, \quad \mathbf{w}(0) = \mathbb{1}_N$$

message actually sent by i: $\sigma_i(k) \;\; = \;\; \sum_{t=0}^{k} r_i(t)$

$\rho_j^{(i)}(k)$ last message received by j from i before time k

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with “mass” counters (synch.)

$$\mathbf{x}(k) = \frac{\mathbf{s}(k)}{\mathbf{w}(k)}, \quad \mathbf{s}(0) = \mathbf{y}, \quad \mathbf{w}(0) = \mathbb{1}_N$$

message actually sent by i: $\sigma_i(k) \;=\; \sum_{t=0}^{k} r_i(t)$

$\rho_j^{(i)}(k)$ last message received by j from i before time k

Agent j receives a message from agent i

$$s_j(k+1) \;=\; s_j(k) + \sigma_i(k) - \rho_j^{(i)}(k)$$

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with "mass" counters (synch.)

$$\mathbf{x}(k) = \frac{\mathbf{s}(k)}{\mathbf{w}(k)}, \quad \mathbf{s}(0) = \mathbf{y}, \quad \mathbf{w}(0) = \mathbb{1}_N$$

message actually sent by i: $\sigma_i(k) \ = \ \sum_{t=0}^{k} r_i(t)$

$\rho_j^{(i)}(k)$ last message received by j from i before time k

Agent j receives a message from agent i

$$s_j(k+1) \ = \ s_j(k) + \sigma_i(k) - \rho_j^{(i)}(k)$$

Information sent and not received

Consensus algorithm with packet losses (ra-AC)

[Vaidya et al. 2012] ratio consensus with "mass" counters (synch.)

$$\mathbf{x}(k) = \frac{\mathbf{s}(k)}{\mathbf{w}(k)}, \quad \mathbf{s}(0) = \mathbf{y}, \quad \mathbf{w}(0) = \mathbb{1}_N$$

message actually sent by i: $\sigma_i(k) \;=\; \sum_{t=0}^{k} r_i(t)$

$\rho_j^{(i)}(k)$ last message received by j from i before time k

Agent j receives a message from agent i

$$s_j(k+1) \;=\; s_j(k) + \sigma_i(k) - \rho_j^{(i)}(k)$$

- communications are persistent
- bounded packet losses

Ergodicity theory $\rightarrow$ $\mathbf{x}(k) \rightarrow x^* \mathbb{1}_N$ exponentially as $k \rightarrow \infty$

Time scale separation

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

- ε is small
- slow dynamics on x_is
- fast dynamics on the ratios $\frac{y_i}{z_i}$

Time scale separation

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

- ε is small
- slow dynamics on x_is
- fast dynamics on the ratios $\frac{y_i}{z_i}$

Fast dynamics

- $\varepsilon \approx 0 \implies \mathbf{x}(k+1) \approx \mathbf{x}(k) = \mathbf{x}$ (constant)
- $\frac{y_i(k)}{z_i(k)} \to \frac{\frac{1}{N}\sum_{i=1}^N g_i(x_i)}{\frac{1}{N}\sum_{i=1}^N h_i(x_i)} = \frac{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)x_i - f_i'(x_i)}{\frac{1}{N}\sum_{i=1}^N f_i''(x_i)} = \frac{\bar{g}(\mathbf{x})}{\bar{h}(\mathbf{x})}$

Time scale separation

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{y_i(k+1)}{z_i(k+1)}$$

- ε is small
- slow dynamics on x_is
- fast dynamics on the ratios $\frac{y_i}{z_i}$

Slow dynamics

- $\dfrac{y_i(k)}{z_i(k)} = \dfrac{\bar{g}(\mathbf{x})}{\bar{h}(\mathbf{x})}$
- $x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{\bar{g}(\mathbf{x}(k))}{\bar{h}(\mathbf{x}(k))}$

Time scale separation

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{y_i(k+1)}{z_i(k+1)}$$

- ε is small
- slow dynamics on x_is
- fast dynamics on the ratios $\frac{y_i}{z_i}$

Slow dynamics

- $\frac{y_i(k)}{z_i(k)} = \frac{\bar{g}(\mathbf{x})}{\bar{h}(\mathbf{x})}$
- $x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon \frac{\bar{g}(\mathbf{x}(k))}{\bar{h}(\mathbf{x}(k))}$
- same forcing term $\Rightarrow \lim_{k\to\infty} x_i(k) - x_j(k) = 0 \Rightarrow \lim_{k\to\infty} x_i(k) = \bar{x}$

Time scale separation

$$x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{y_i(k+1)}{z_i(k+1)}$$

- ε is small
- slow dynamics on x_is
- fast dynamics on the ratios $\frac{y_i}{z_i}$

Slow dynamics

- $\frac{y_i(k)}{z_i(k)} = \frac{\bar{g}(\mathbf{x})}{\bar{h}(\mathbf{x})}$
- $x_i(k+1) = (1-\varepsilon)x_i(k) + \varepsilon\frac{\bar{g}(\mathbf{x}(k))}{\bar{h}(\mathbf{x}(k))}$
- same forcing term $\Rightarrow \lim_{k\to\infty} x_i(k) - x_j(k) = 0 \Rightarrow \lim_{k\to\infty} x_i(k) = \bar{x}$
- $\bar{x}^+ = \bar{x} - \varepsilon\frac{f'(\bar{x})}{f''(\bar{x})}$ $\Rightarrow$ centralized Newton Raphson step