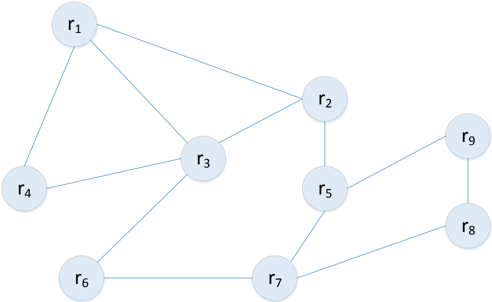
Distributed Average Consensus with Deterministic Quantization: An ADMM Approach

Shengyu Zhu and Biao Chen

Electrical Engineering and Computer Science
Syracuse University

Dec 2015

Distributed Average Consensus



$r_i \in \mathbb{R}$: data at agent i

N: number of agents

E: number of edges

(i, j): the edge connecting agents i and j

$\mathcal{N}_i$: the set of neighboring agents of agent i

- Goal: using only local computation and communication to reach the consensus at

$$x_{\text{avg}} = \frac{1}{N} \sum_{i=1}^{N} r_i$$

Distributed Average Consensus (cont.)

- Motivation and Applications
  - distributed agreement and synchronization
  - distributed coordination of mobile autonomous agents
  - distributed data fusion in sensor networks
  - load balancing for parallel computers
  - ...
- Chanllenges
  - no fusion center in large scale networks ⇒ distributed algorithms
  - limited channel capacity, agent power, etc. ⇒ quantization

⇒ Quantized Consensus *[Kashyap'2007]*

Distributed Average Consensus (cont.)

- Motivation and Applications
  - distributed agreement and synchronization
  - distributed coordination of mobile autonomous agents
  - distributed data fusion in sensor networks
  - load balancing for parallel computers
  - ...

- Chanllenges
  - no fusion center in large scale networks ⇒ distributed algorithms
  - limited channel capacity, agent power, etc. ⇒ quantization

⇒ Quantized Consensus *[Kashyap'2007]*

Distributed Average Consensus (cont.)

- Motivation and Applications
  - distributed agreement and synchronization
  - distributed coordination of mobile autonomous agents
  - distributed data fusion in sensor networks
  - load balancing for parallel computers
  - $\cdots$

- Chanllenges
  - no fusion center in large scale networks $\Rightarrow$ distributed algorithms
  - limited channel capacity, agent power, etc. $\Rightarrow$ quantization

$\Rightarrow$ Quantized Consensus *[Kashyap'2007]*

Existing Approaches

Notation: $\boldsymbol{x}^k = [x_1^k; x_2^k; \cdots ; x_N^k]$ and $\boldsymbol{x}^0 = [r_1; r_2; \cdots ; r_N]$

- Classical approach: $\boldsymbol{x}^{k+1} = \boldsymbol{W}\boldsymbol{x}^k$
  - without quantization: $x_i^k \to x_{\text{avg}}$ when $\boldsymbol{W}$ is doubly stochastic *[Elsner'1990, Xiao'2004, Jakovetic'2010, Nedić'2009]*
  - dithered quantization: $\mathbb{E}[x_i^k] \to x_{\text{avg}}$ *[Aysal'2008]*
  - deterministic quantization: x_i^k converges to a consensus in finite time with an error from x_{avg} or cycles in a small neighborhood around x_{avg} *[Nedić'2009, Chamie'2014]*
- Gossip based approach: $x_i^{k+1} = x_j^{k+1} = \frac{x_i^k + x_j^k}{2}$ for a randomly selected (i, j)
  - Similar results *[Tsitsiklis'1984, Ashyap'2007, Kar'2010, Carli'2010]*

Existing Approaches (cont.)

Fact: $x_{\mathsf{avg}} = \arg\min_x \frac{1}{2}\sum_{i=1}^{N}(x - r_i)^2$

Equivalent ADMM formulation for connected networks:

$$\begin{aligned} \underset{\{x_i\},\{z_{ij}\}}{\text{minimize}} \quad & \sum_{i=1}^{N} \frac{1}{2}(x_i - r_i)^2 \\ \text{subject to} \quad & x_i = z_{ij}, x_j = z_{ij}, \forall (i,j) \end{aligned}$$

Existing Approaches (cont.)

ADMM based Approach: Distributed Consensus ADMM (DC-ADMM)

[Schizas'2008,Zhu'2009]

$$x_i^{k+1} = \frac{1}{1 + 2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|x_i^k + \rho\sum_{j\in\mathcal{N}_i} x_j^k - \alpha_i^k + r_i\right),$$

$$\alpha_i^{k+1} = \alpha_i^k + \rho\left(|\mathcal{N}_i|x_i^{k+1} - \sum_{j\in\mathcal{N}_i} x_j^{k+1}\right).$$

- ρ: any fixed positive number
- without quantization: $x_i^k \to x_{\text{avg}}$ *[Schizas'2008,Zhu'2009]*; linear rate under mild conditions *[Shi'2014]*
- dithered quantization: $\mathbb{E}[x_i^k] \to x_{\text{avg}}$ *[Zhu'2009]*
- deterministic quantization: no results and a hard problem

Existing Approaches (cont.)

With quantization, existing approaches

- do not converge to a consensus in finite time, or
- converge to a consensus in polynomial time with an error increasing in the range of agents data, quantization resolution and the number of agents

  Big Problem: large data range and network size in today's applications, e.g., large scale networks or big data settings

DC-ADMM: resilient to noise; fast convergence (linear rate)

Our goal: Can we use the ADMM to achieve better results?

Existing Approaches (cont.)

With quantization, existing approaches

- do not converge to a consensus in finite time, or
- converge to a consensus in polynomial time with an error increasing in the range of agents data, quantization resolution and the number of agents

  Big Problem: large data range and network size in today's applications, e.g., large scale networks or big data settings

DC-ADMM: resilient to noise; fast convergence (linear rate)

Our goal: Can we use the ADMM to achieve better results?

Modified DC-ADMM Algorithm

Rounding Quantization

$$Q_d(y) = t\Delta, \text{ if } \left(t - \frac{1}{2}\right)\Delta \leq y < \left(t + \frac{1}{2}\right)\Delta, t \in \mathbb{Z}$$

Deterministically Quantized DC-ADMM (DQ-DC-ADMM)

$$x_i^{k+1} = \frac{1}{1 + 2\rho|\mathcal{N}_i|}\left(\rho|\mathcal{N}_i|x_{i[Q_d]}^k + \rho\sum_{j\in\mathcal{N}_i} x_{j[Q_d]}^k - \alpha_i^k + r_i\right),$$

$$\alpha_i^{k+1} = \alpha_i^k + \rho\left(|\mathcal{N}_i|x_{i[Q_d]}^{k+1} - \sum_{j\in\mathcal{N}_i} x_{j[Q_d]}^{k+1}\right).$$

Convergence Results

If $\boldsymbol{\alpha}^0 = [\alpha_1^0; \alpha_2^0; \cdots; \alpha_N^0]$ lies in the column space of the Laplacian matrix of the graph (e.g., $\alpha_i^0 = 0$), then the DQ-DC-ADMM has

- *Convergence:* the sequence $(\boldsymbol{x}_{[Q_d]}^k, \boldsymbol{\alpha}^k)$ converges to a finite value $(\mathbf{1}x_{[Q_d]}^*, \boldsymbol{\alpha}^*)$ where $\mathbf{1}$ is a N-dimensional vector with all entries being 1.
- *Consensus error:* a tight upper bound for the consensus error is given by

$$|x_{[Q_d]}^* - x_{\mathsf{avg}}| \leq \left(\frac{1}{2} + \rho \frac{2E}{N}\right) \Delta.$$

- *Number of iterations:* DQ-DC-ADMM converges within $\lceil \log_{1+\delta} \Omega \rceil + 3$ iterations, where $\delta > 0$ depends on ρ and the network topology and Ω is a polynomial fraction depending on ρ, Δ, agents' data, and the network topology.

ADMM Based Algorithm for Quantized Consensus

- Note: the DQ-DC-ADMM does not converge to global optima; a good starting point usually helps

- Probability Quantizer

$$Q_p(y) = \begin{cases} t\Delta, & \text{with probability } \frac{y}{\Delta} - t, \\ (t+1)\Delta, & \text{with probability } t + 1 - \frac{y}{\Delta}. \end{cases}$$

- PQDQ-DC-ADMM: First run the PQ-DC-ADMM to obtain good estimates; then run the DQ-DC-ADMM to reach a consensus

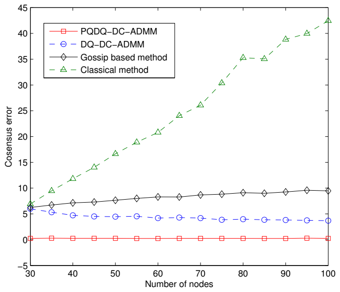
ADMM Based Algorithm for Quantized Consensus

- Note: the DQ-DC-ADMM does not converge to global optima; a good starting point usually helps
- Probability Quantizer

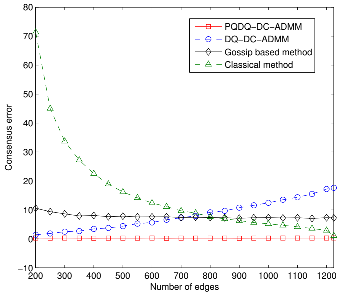
$$Q_p(y) = \begin{cases} t\Delta, & \text{with probability } \frac{y}{\Delta} - t, \\ (t+1)\Delta, & \text{with probability } t + 1 - \frac{y}{\Delta}. \end{cases}$$

- PQDQ-DC-ADMM: First run the PQ-DC-ADMM to obtain good estimates; then run the DQ-DC-ADMM to reach a consensus

Simulation: Consensus Error



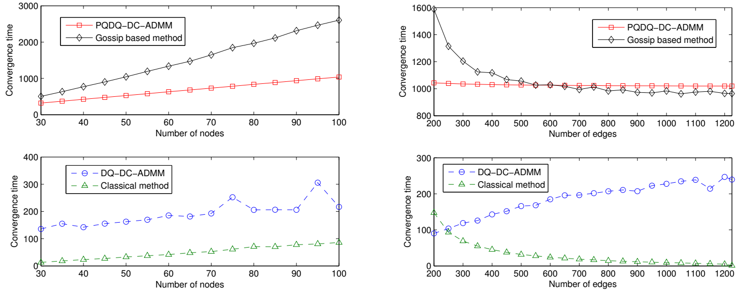
(a) fixing $\frac{2E}{N} = 20$



(b) fixing $N = 50$

- Setting: $\Delta = 1$, $r_i \sim$ unif$[-N, N]$; $\rho = 1$
- DQ-DC-ADMM: much smaller than the upper bound
- PQDQ-DC-ADMM: typically less than one Δ for all connected networks with agents' data of arbitrary magnitudes

Simulations: Convergence time



(a) fixing $\frac{2E}{N} = 20$ (b) fixing $N = 50$

- Setting: $\Delta = 1$, $r_i \sim \text{unif}[-N, N]$; $\rho = 1$
- DQ-DC-ADMM: increases as the data range and graph density becomes larger
- PQDQ-DC-ADMM: converges almost immediately after the PQ-DC-ADMM stage

Conclusion

DQ-DC-ADMM

- converges relatively fast (within $\lceil \log_{1+\delta} \Omega \rceil + 3$ iterations)
- consensus error does not depend on agents' data or the network size

PQDQ-DC-ADMM

- consensus error is typically within one quantization resolution regardless of network topology and agents' data
- need more iterations