Averaging algorithms and distributed optimization

John N. Tsitsiklis
MIT

NIPS 2010 Workshop on Learning
on Cores, Clusters and Clouds
December 2010

Outline

- Motivation and applications
- Consensus/averaging in distributed optimization
- Convergence times of consensus/averaging
  - time-invariant case
  - time-varying case

The Setting

- n agents
  - starting values $x_i(0)$
- reach consensus on some x^*, with either:
  - $\min_i x_i(0) \leq x^* \leq \max_i x_i(0)$ (consensus)
  - $x^* = \dfrac{x_1(0) + \cdots + x_n(0)}{n}$ (averaging)
  - averaging when $x_i \in \{-1, +1\}$ (voting)
- interested in:
  - genuinely distributed algorithm
  - no synchronization
  - no "infrastructure" such as spanning trees
- simple updates, such as: $x_i := \dfrac{x_i + x_j}{2}$

Social sciences

- Merging of "expert" opinions
- Evolution of public opinion
- Evolution of reputation
- Modeling of jurors
- Language evolution

- Preference for "simple" models
  - behavior described by "rules of thumb"
  - less complex than Bayesian updating
- interested in modeling, analysis (descriptive theory)
  - ... and narratives

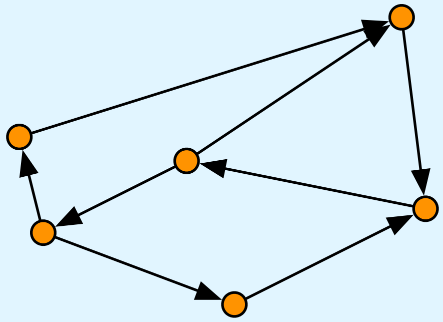
Engineering

- Distributed computation and sensor networks
  - Fusion of individual estimates
  - Distributed Kalman filtering
  - Distributed optimization
  - Distributed reinforcement learning
- Networking
  - Load balancing and resource allocation
  - Clock synchronization
  - Reputation management in ad hoc networks
  - Network monitoring
- Multiagent coordination and control
  - Coverage control
  - Monitoring
  - Creating virtual coordinates for geographic routing
  - Decentralized task assignment
  - Flocking

The DeGroot opinion pooling model (1974)

$x_i(t+1) = \sum_j a_{ij}\, x_j(t)$ $\qquad$ $a_{ij} \geq 0, \quad \sum_j a_{ij} = 1$

$x(t+1) = Ax(t)$ $\qquad$ A: stochastic matrix



- Markov chain theory + "mixing conditions"
  - $\longrightarrow$ convergence of A^t, to matrix with equal rows
  - $\longrightarrow$ convergence of x_i to $\sum_j \pi_j x_j$
  - $\longrightarrow$ convergence rate estimates

- Averaging algorithms
  - A doubly stochastic: $\mathbf{1}' A\, x = \mathbf{1}'\, x$, $\quad$ where $\mathbf{1}' = [1\ 1\ \ldots\ 1]$
  - $x_1 + \cdots + x_n$ is conserved
  - convergence to $x^* = \dfrac{x_1(0) + \cdots + x_n(0)}{n}$

Part I: Distributed Optimization

Gradient-like methods

- $\min_x f(x)$ special case: $f(x) = \sum_i f_i(x)$
  - $f,\ f_i$ convex
- f smooth; work with $\nabla f(x)$
  - update: $x := x - \gamma \nabla f(x)$
  - with noise: $x := x - \gamma(\nabla f(x) + w)$
    (stochastic approximation, $\gamma_t \to 0$)
- f nonsmooth, work with subgradient $\partial f(x)$
  - update: $x := x - \gamma \partial f(x)$ ($\gamma_t \to 0$)
  - with noise: $x := x - \gamma(\partial f(x) + w)$
- More sophisticated variants: Dual averaging methods

Smooth f; compentwise decentralization

- x^i_j: agent i, component j
  - update: $x^i_i := x^i_i - \gamma \dfrac{\partial f}{\partial x_i}(x^i)$
  - reconcile: $x^i_j := x^j_j$ (occasionally; upper bound B)
- Analysis: track $y = (x^1_1, \ldots, x^n_n)$

$$\|y - x^i\| = O(B\gamma)$$

$$y := y - \gamma \nabla f(y) + O(B\gamma^2)$$

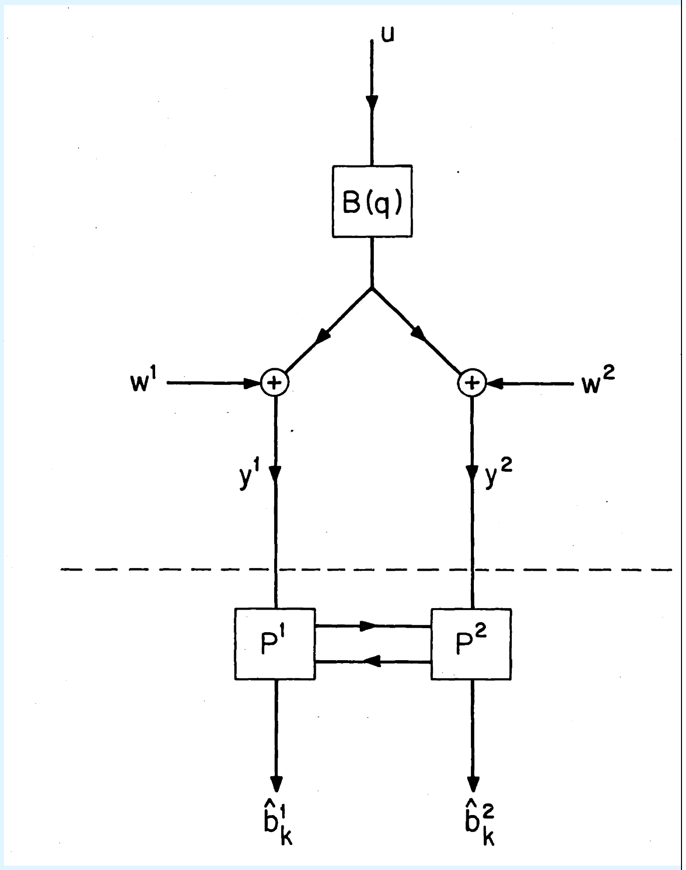
- Convergence theorem for centralized gradient method remains valid: [Bertsekas, JNT, Athans, 86]
  - need $\gamma \sim 1/B$
  - also for stochastic approximation variant

$$x^i_i := x^i_i - \gamma \left(\frac{\partial f}{\partial x_i}(x^i) + w_i \right)$$

Smooth f; overlap and cooperate

- Assume (for simplicity) scalar x
  - subscript denotes agent's value of x
  - $x_i := x_i - \gamma f(x_i)$ redundant/useless

- useful in the presence of noise:
  - update: $x_i := x_i - \gamma\left(\nabla f(x_i) + w_i\right)$
  - reconcile: $x := x - \gamma \cdot \frac{1}{n}\sum_i (\nabla f(x_i) + w_i)$



Smooth f; overlap and cooperate (ctd.)

- **Two-phase version**
  - update: $x_i := x_i - \gamma\,(\nabla f(x_i) + w_i)$
  - reconcile: run consensus algorithm $x := Ax$

  converges: $x_i \to y,\ \forall i$ $\qquad y = \sum_j \pi_j x_j \qquad \pi_j \geq 0$

$$y := y - \gamma \sum_j \pi_j (\nabla f(x_j) + w_j)$$

- expected update direction is still descent direction
- classical convergence results for centralized stochastic gradient method, with $\gamma_t \to 0$, remain valid

Smooth f; overlap and cooperate (ctd.)

- **Interleaved version**

$$\boxed{x_i := \sum_j a_{ij} x_j - \gamma\,(\nabla f(x_i) + w_i)}$$

  - define $y = \sum_i \pi_i x_i$
  - note: $\sum_i \pi_i \sum_j a_{ij} x_j = \sum_i \pi_i x_i$

$$y := y - \gamma \sum_i \pi_i (\nabla f(x_i) + w_i)$$

- $|x_i - y| = O(\gamma T \cdot |\nabla f(y)|)$

  T: convergence time (time constant) of consensus algorithm

$$y := y - \gamma \sum_i \pi_i (\nabla f(y) + w_j) + O(\gamma^2 T \cdot |\nabla f(y)|)$$

- convergence theorem for centralized stochastic gradient method, with $\gamma_t \to 0$, remains valid [Bertsekas, JNT, Athans, 86]

Smooth, additive f; overlap and cooperate

- $f(x) = \frac{1}{n}\sum_i f_i(x)$ optimality $\iff$ $\sum_i \nabla f_i(x) = 0$
- **Two-phase version**
  - update: $x_i := x_i - \gamma \nabla f_i(x_i)$
  - reconcile: run consensus algorithm $x := Ax$

    converges: $x_i \to y$, $\forall i$ $\quad y = \sum_i \pi_i x_i \quad \pi_i \geq 0$

$$y := y - \gamma \sum_i \pi_i \nabla f_i(x_i)$$

- correctness requires $\pi_i = 1/n$
  - Use **averaging** algorithm (A: doubly stochastic)

Additive f; overlap and cooperate (ctd.)

- **Interleaved version**

$$\boxed{x_i := \sum_j a_{ij} x_j - \gamma \nabla f_i(x_i) + w_i}$$

  - define $y = \frac{1}{n} \sum_i x_i$

$$y := y - \gamma \frac{1}{n} \sum_i \nabla f_i(x_i)$$

- $|x_i - y| = O\Big(\gamma T \cdot \sum_i |\nabla f_i(y)|\Big)$

  T: convergence time (time constant) of averaging algorithm
  - for constant γ, error does not vanish at optimum
  - optimality possible only with $\gamma_t \to 0$
    (even in the absence of noise)
  - hence studied for nonsmooth f or stochastic case
    [Nedic & Ozdaglar, 09; Duchi, Agarwal, & Wainright, 10]

Convergence times — the big picture

- $T_{\text{con}}(n, \epsilon)$: time for consensus/averaging algorithm to reduce disagreement from unity to ϵ
  - generically $O(1/\log(1/\epsilon))$
  - focus on $T_{\text{con}}(n)$

- $T_{\text{opt}}(n, \epsilon)$: time for centralized (sub)gradient algorithm to bring cost gap to ϵ
  - hide dependence on other constants (bounds on first, second derivatives, stepsize details)

- **Two-phase version:** $O\Big(T_{\text{con}}(n) \cdot T_{\text{opt}}(n, \epsilon)\Big)$

- **Interleaved version:** Results have the same flavor [Nedic & Ozdaglar, 09; Duchi, Agarwal, & Wainright, 10]
  - is interleaving faster or "better" than two-phase version?

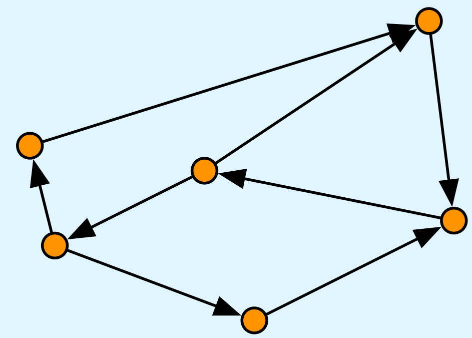
- **Our mission:** study and reduce $T_{\text{con}}(n)$ automatically better overall convergence time e.g., [Nedic, Olshevsky, Ozdaglar & JNT, 08]

Part II: Consensus and averaging

Convergence time of consensus algorithms

$$x_i(t+1) = \sum_j a_{ij}\, x_j(t) \qquad a_{ij} \geq 0, \quad \sum_j a_{ij} = 1$$

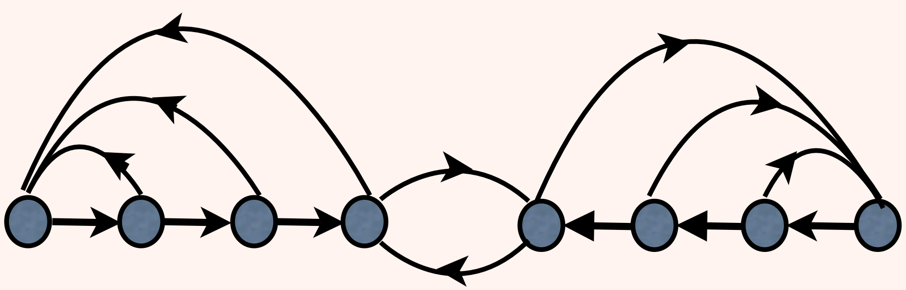
$$x(t+1) = Ax(t) \qquad A\text{: stochastic matrix}$$



Convergence time (time to get close to "steady-state")

Equal weight to all neighbors

Directed graphs: exponential(n)

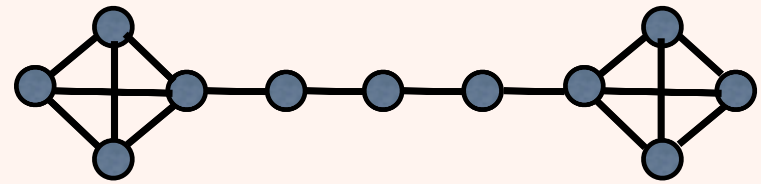


Better results for special graphs

(Erdös-Rényi, geometric, small world)

Undirected graphs: $O(n^3)$, tight

(Landau and Odlyzko, 1981)



$\Theta(n^2)$ for line graphs

Averaging algorithms

- A doubly stochastic: $\mathbf{1}'Ax = \mathbf{1}'x$
  - positive diagonal
  - nonzero entries are $\geq \alpha > 0$
  - convergence to $x^* = \frac{x_1(0)+\cdots+x_n(0)}{n}$
  - convergence time $= O(n^2/\alpha)$

  $V(x) = \sum_i (x_i - x^*)^2$ is a Lyapunov function

  (Nedic, Olshevsky, Ozdaglar & JNT, 09)

- bidirectional graph, natural algorithm:

$$x_i := x_i + \frac{1}{2n} \sum_{\text{neighbors } j} (x_j - x_i)$$

$$\alpha \sim \frac{1}{n} \qquad \text{convergence time} = O(n^3)$$

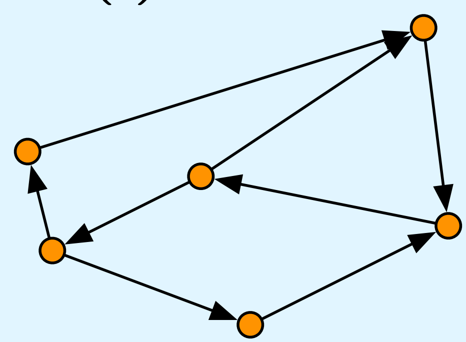
A critique

- The consensus/averaging algorithm $x := Ax$ assumes constant a_{ij} $\Longrightarrow$ **fixed graph**
  - elect a leader, form a spanning tree, accumulate on tree
- Want simplicity and robustness in dealing with changing topologies, failures, etc.

Time-Varying/Chaotic Environments

- i.i.d. random graphs: same (in expectation) as fixed graphs; convergence rate $\longleftrightarrow$ "mixing times" (Boyd et al., 2005)

- Fairly arbitrary sequence of graphs/matrices $A(t)$: worst-case analysis



$$x_i(t+1) = \sum_j a_{ij}(t) x_j(t)$$

$a_{ij}(t)$: nonzero whenever i receives message from j

$x(t+1) = A(t)x(t)$ (inhomogeneous Markov chain)

Consensus convergence

$$x_i(t+1) = \sum_j a_{ij}(t) x_j\big(t\big)$$

- $a_{ii}(t) > 0$; $\quad a_{ij}(t) > 0 \implies a_{ij}(t) \geq \alpha > 0$
- "strong connectivity in bounded time":
  over B time steps "communication graph"
  is strongly connected
- Convergence to consensus:
  $\forall\ i: \quad x_i(t) \to x^* =$ convex combination of initial values
  (JNT, Bertsekas, Athans, 86; Jadbabaie et al., 03)
- "convergence time": exponential in n and B
  - even with:
    symmetric graph at each time
    equal weight to each neighbor
    (Cao, Spielman, Morse, 05)

Averaging in Time-Varying Setting

- $x(t+1) = A(t)x(t)$ (Nedic, Olshevsky, Ozdaglar & JNT, 09)
  - $A(t)$ doubly stochastic, for all t
  - $O(n^2/\alpha)$ bound remains valid!

- Improved convergence rate
  - exchange "load" with up to two neighbors at a time
  - can use $\alpha = O(1)$
  - convergence time: $O(n^2)$

- Averaging in time-varying bidirectional graphs: no harder than consensus on fixed graphs $O(n^2)$

- Various convergence proofs of optimization algs. remain valid
  - Improves the convergence time estimate for subgradient methods [Nedic, Olshevsky, Ozdaglar, JNT, 09]

Can we beat $O(n^2)$?

- The program: Understand the question for static graphs
- Yes, for special static graphs
- No, in general, if we restrict to (possibly nonlinear) update functions

$$x_i := f(x_j;\ j \in \text{neighbors of } i)$$

  that are smooth [Olshevsky & JNT, 10]
  - Nonlinearity cannot help
  - Playing with the coefficients of random walks on a line does not help

- Yes, if we allow building a spanning tree
- We want to rule this out by picking a precise model of computation

A model of computation; static graphs

- To have a hope for strong lower bounds,
  rule out fancy encoding of information in real numbers
  - work with discrete messages
  - can only solve discrete problems

- **The majority problem**
  - $x_i \in \{-1, 1\}$; Is the average > 0?

- **Model:**
  - Fixed but unknown bidirectional graph
  - No randomization
  - Anonymous nodes, all running same code
  - Bounded message alphabet

Majority problem under our model

- Is $O(n^2)$ possible, in the first place?
- Yes! (nontrivial)
  (Hendrickx, Olshevsky & JNT, 10)
- **Idea:** move -1s and $+1$s around
  - cancel them when they meet
  - see what is left

- Open questions
  - Can we get a $\Omega(n^2)$ lower bound? (may be hard)
  - Can we get $O(n^2)$ on directed static graphs?
  - Can we get $O(n^2)$ method for time-varying graphs?
    (under what connectivity assumptions?)

Thank you!