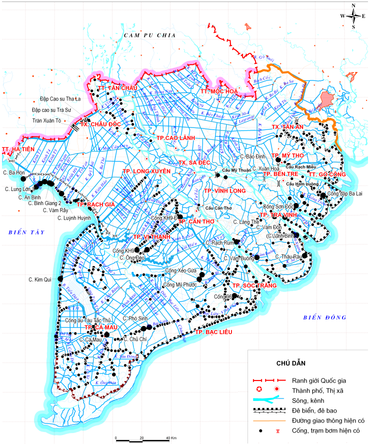
Distributed embedded optimization and applications on distributed sensing over water systems

Dang Doan

Syscop, IMTEK, University of Freiburg

Syscop retreat September $5^{\text{th}} - 7^{\text{th}}$, 2016

Motivation



- Map of water control structures in Mekong Delta of Vietnam, 2009
  - Blue lines: rivers and canals
  - Connected squares: dikes
  - Round black dots: sluices
  - Red dots: major pumping stations
- Problems to address:
  - Lacking of sensing facilities
  - Burdensome investment cost for measuring water flow rates
  - Lacking of control strategies for water-related problems
- This regional water system is a potential application for distributed estimation and optimal control

Problem description

- Solve convex problem with mixed quadratic and 1-norm cost:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{y}} \quad & \frac{1}{2}\mathbf{x}^T H \mathbf{x} + g^T \mathbf{x} + \gamma \|\mathbf{y}\|_1 \\ \text{s.t.} \quad & A_1 \mathbf{x} = b_1 \\ & A_2 \mathbf{x} \le b_2 \\ & \mathbf{y} = P\mathbf{x} - p \end{aligned}$$

  with $\mathbf{x} = [x_1^T, \cdots, x_M^T]^T$, separable Hessian $H = blkdiag(H_1, \cdots, H_M)$ with H_i pos. def., $\gamma > 0$, and A_1, A_2, P have sparsity structure.
- Objective: design a distributed algorithm with fast convergence rate
- Key ideas:
  - Use accelerated proximal gradient (APG) algorithm (first order method)
  - Solve the dual problem
  - Distributing computations wrt. sparsity structure

Proximal gradient method

- Solve unconstrained problem:

$$\min_{\mathbf{x}} \quad f(\mathbf{x}) = g(\mathbf{x}) + h(\mathbf{x})$$

  - $g(\cdot)$ convex, differentiable
  - h closed, convex, possible non-differentiable; with cheap **proximal mapping**: $\mathbf{prox}_h(x) \triangleq \arg\min_u (h(u) + \frac{1}{2}\|u - x\|_2^2)$
- Iterative algorithm:

$$\mathbf{x}_{k+1} = \mathbf{prox}_{t_k h}\left(\mathbf{x}_k - t_k \nabla g(\mathbf{x}_k)\right)$$

  with step size $t_k > 0$, fixed or determined by line search.
- Intepretation: minimize a quadratic upper bound of the mixed objective function at the current iterate $y = \mathbf{x}_k$: $g(x) + h(x) \leq f(y) + \nabla g(y)^T (x - y) + \frac{L}{2}\|x - y\|_2 + h(x)$
- Convergence rate when ∇g is Lipschitz continous: $\|\nabla g(x) - \nabla g(y)\|_2^2 \leq L\|x - y\|_2, \forall x, y$ and step size $t_k = \frac{1}{L}$: need $O(\frac{1}{\epsilon})$ iterations for ϵ accuracy.

Accelerated proximal gradient method

- Iterative algorithm: start from $\mathbf{x}_0 \in \mathbf{dom} h$ and $\mathbf{y}_0 = \mathbf{x}_0$

$$\mathbf{x}_{k+1} = \mathbf{prox}_{t_k h}(\mathbf{y}_k - t_k \nabla g(\mathbf{y}_k))$$
$$\mathbf{y}_{k+1} = \mathbf{x}_{k+1} + \frac{k-1}{k+2}(\mathbf{x}_{k+1} - \mathbf{x}_k)$$

- Some properties:
  - Faster convergence than proximal gradient method
  - Compare to proximal gradient: need to store $\mathbf{y}_k$, and one more simple matrix calculation
  - Convergence rate when ∇g is Lipschitz continous with constant L and $t_k = \frac{1}{L}$: need $O(\frac{1}{\sqrt{\epsilon}})$ iterations for ϵ accuracy.
  - **Difficulty for application**: this method only applies to **unconstrained** problems.

Applying APG to the dual problem

- Primal problem:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{y}} \quad & \frac{1}{2}\mathbf{x}^T H\mathbf{x} + g^T\mathbf{x} + \gamma\|\mathbf{y}\|_1 \\ \text{s.t.} \quad & A_1\mathbf{x} = b_1 \quad \mid \quad z_1 \\ & A_2\mathbf{x} \le b_2 \quad \mid \quad z_2 \\ & \mathbf{y} = P\mathbf{x} - p \quad \mid \quad z_3 \end{aligned}$$

- Dual problem:

$$f(z) := \frac{1}{2}(\mathcal{A}^T z + g)^T H^{-1}(\mathcal{A}^T z + g) + \mathcal{B}^T z$$

  - with $z = [z_1^T, z_2^T, z_3^T]^T$
  - $\nabla f(z) = \mathcal{A}H^{-1}(\mathcal{A}^T z + g) + \mathcal{B}$, is Lipschitz continous with $L = \|\mathcal{A}H^{-1}\mathcal{A}^T\|_2$.
  - Function $h(\cdot)$ in the APG method is indicator function $I_Z \triangleq z \in Z \Leftrightarrow z_2 \ge 0, |z_3| \le \gamma * 1$, and $\mathbf{prox}_{I_Z}(z) = \arg\min_{u\in Z} \|u - z\|_2^2 = \mathcal{P}_Z(z)$.

Applying APG to the dual problem

Iteration:

$$\begin{aligned} v^k &= z^k + \frac{k-1}{k+2}(z^k - z^{k-1}) \\ z^{k+1} &= \mathcal{P}_Z\left(v^k - \frac{1}{L}\nabla f(v^k)\right) \\ \mathbf{x}^{k+1} &= H^{-1} z^{k+1} \end{aligned}$$

where $\mathcal{P}_Z$ is the Euclidean projection onto the set Z. Thus, the new iterate, z^{k+1}, is the previous iterate plus a step in the negative gradient direction projected onto the feasible set.

Distributed dual accelerated proximal gradient

Table : Comparison for 100 random DMPC problems. DDAPG with different step size L is implemented in MATLAB while CPLEX and MOSEK are implemented in C.

Alg.	vars./constr.	tol.	# iters		exec (ms)	
			mean	max	mean	max
DDAPG (L)	4320/3231	0.005	69.8	160	253	609
DDAPG (L_1)	4320/3231	0.005	160	420	594	1532
DDAPG (L_F)	4320/3231	0.005	248	640	934	2444
MOSEK	4320/3231	-	-	-	1945	2674
CPLEX	4320/3231	0.005	-	-	1663	2832
DDAPG (L)	2160/1647	0.005	63.8	100	94	200
DDAPG (L_1)	2160/1647	0.005	75.8	180	115	368
DDAPG (L_F)	2160/1647	0.005	121	320	185	488
MOSEK	2160/1647	-	-	-	334	399
CPLEX	2160/1647	0.005	-	-	282	522

Distributed dual accelerated proximal gradient

- Applicability and properties of DDAPG method:
  - Mixed convex quadratic and 1-norm costs
  - Linear constraints
  - Inversion of H is pre-calculated (invariant)
  - Fast convergence rate (among first-order methods), with a suboptimality of $O(\frac{1}{k^2})$
- Application of DDAPG on distributed MPC of a Hydro Power Valley:
  - Distributed MPC can handle the power reference tracking
  - Depending on the availability of the communication structure, distributed MPC achieve different sub-optimality
  - When the problem is sparse, distributed optimization is well suited and can outperform centralized optimization algorithms

Future research on distributed sensing over water systems

- Implement DDAPG method into embedded systems:
  - Mostly only need simple linear algebra operations
  - For prototyping: Linux-enabled box (BeagleBone, RPi, Udoo, Odroid...)
  - Promising to be implemented only with C code on non-OS microcontrollers
- Distributed sensing over water systems:
  - Modelling of the water system as a network of cooperative subsystems
  - Formulate the moving-horizon estimation problem over the water system
  - Implement DDAPG method for the MHE problem with the network of microcontrollers

Intended tasks in implementation - DISCUSSIONS

- Implement DDAPG solver on microcontrollers:
  - Develop with C code, on ARM / MIPS architecture
  - Construct basic linear algebra library (matrix addition, multiplication, assign, get sub-matrix)
  - Pre-condition for general starting optimization problems (may be implemented on PC)
  - Build communication procedure (choose transmission protocol, define frame of data, method of synchronization)
  - Power supply (battery, solar cell)
- Formulate the distributed MHE over water systems:
  - Modelling: from physical model, make mathematical model (discretization, reduction)
  - Identification: if detailed physical model not available, do system identification with low-order models
  - Need sensors for measurement of water levels, water flows
  - Formulate the MHE for estimating water flows on where the sensors are not available

Discussions...

Thank you for your attention and discussions!