ADMM

alternating direction method of multipliers

Jingchang Liu

December 28, 2016

Table of Contents

Overview of ADMM

Problem

$$\begin{aligned} min \quad & f(x) + g(z) \\ s.t. \quad & Ax + Bz = c \end{aligned}$$

History

- First proposed in the later 1960s.
- Reintroduced by Boyd et.al. in 2011.

Iteration

Augmented Lagrangian

$$L_\mu\left(x, z, y\right) = f(x) + g(z) + y^T\left(Ax + Bz - c\right) + \frac{\mu}{2}\left\|Ax + Bz - c\right\|_2^2$$

Iteration

Augmented Lagrangian

$$L_\mu(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\mu}{2}\|Ax + Bz - c\|_2^2$$

Interation

$$\begin{aligned}
x^{k+1} &:= \underset{x}{argmin}\, L_\mu(x, z^k, y^k) && (1)\\
z^{k+1} &:= \underset{z}{argmin}\, L_\mu(x^{k+1}, z, y^k) && (2)\\
y^{k+1} &:= y^k + \mu\left(Ax^{k+1} + Bz^{k+1} - c\right) && (3)
\end{aligned}$$

Table of Contents

Problem

Convex Opt problem

$$
\begin{aligned}
min \quad & f(x) && (4)\\
s.t. \quad & Ax = b && (5)
\end{aligned}
$$

Dual

- Lagrangian

$$L(x, y) = f(x) + y^T(Ax - b)$$

- Dual function

$$g(y) = \inf_x L(x, y) = -f^*(-A^T y) - b^T y$$

- Dual problem

$$max\ g(y)$$

- Relationship

$$x^* = \underset{x}{argmin}\, L(x, y^*)$$

Dual ascent

1. $x^+ = argmin_x L(x, y)$
2. $\bigtriangledown g(y) = Ax^+ - b$

Iteration

$$
\begin{aligned}
x^{k+1} &:= \underset{x}{argmin}\, L(x, y^k) &(6)\\
y^{k+1} &:= y^k + \alpha^k (Ax^{k+1} - b) &(7)
\end{aligned}
$$

Table of Contents

Equivalent opt

Origin

$$\begin{aligned} min \quad & f(x) && (8) \\ s.t. \quad & Ax = b && (9) \end{aligned}$$

Equal

$$\begin{aligned} min \quad & f(x) + \frac{\mu}{2} \|Ax - b\|_2^2 && (10) \\ s.t. \quad & Ax = b && (11) \end{aligned}$$

Augmented Lagrangians

- $L_\mu(x, y) = f(x) + y^T(Ax - b) + \frac{\mu}{2}\|Ax - b\|_2^2$
- Dual function: $g_\mu(y) = inf_x L_\mu(x, y)$
- Dual ascent

$$x^{k+1} := \underset{x}{argmin}\, L_\mu(x, y^k) \tag{12}$$

$$y^{k+1} := y^k + \alpha^k(Ax^{k+1} - b) \tag{13}$$

Table of Contents

Problem format

Problem

$$\begin{aligned} min \quad & f(x) + g(z) \\ s.t. \quad & Ax + Bz = c \end{aligned}$$

Augmented Lagrangian

$$L_\mu(x, z, y) = f(x) + g(z) + y^T(Ax + Bz - c) + \frac{\mu}{2}\|Ax + Bz - c\|_2^2$$

Methods

Dual ascent

$$(x^{k+1}, z^{k+1}) := \underset{x,z}{argmin}\, L_\mu(x, z, y^k) \tag{14}$$
$$y^{k+1} := y^k + \mu(Ax^{k+1} + Bz^{k+1} - c) \tag{15}$$

ADMM

$$x^{k+1} := \underset{x}{argmin}\, L_\mu(x, z^k, y^k) \tag{16}$$
$$z^{k+1} := \underset{z}{argmin}\, L_\mu(x^{k+1}, z, y^k) \tag{17}$$
$$y^{k+1} := y^k + \mu\left(Ax^{k+1} + Bz^{k+1} - c\right) \tag{18}$$

Scaled form

Rule

$$
\begin{aligned}
y^T r + \frac{\mu}{2}\|r\|_2^2 &= \frac{\mu}{2}\|r + \frac{1}{\mu} y\|_2^2 - \frac{1}{2\mu}\|y\|_2^2 && (19)\\
&= \frac{\mu}{2}\|r + u\|_2^2 - \frac{1}{2\mu}\|y\|_2^2 && (20)
\end{aligned}
$$

Scaled form

Rule

$$y^T r + \frac{\mu}{2}\|r\|_2^2 = \frac{\mu}{2}\|r + \frac{1}{\mu}y\|_2^2 - \frac{1}{2\mu}\|y\|_2^2 \quad (19)$$

$$= \frac{\mu}{2}\|r + u\|_2^2 - \frac{1}{2\mu}\|y\|_2^2 \quad (20)$$

Scaled form $u = (1/\mu)y$

$$x^{k+1} := \underset{x}{argmin}\left(f(x) + \frac{\mu}{2}\|Ax + Bz^k - c + u^k\|_2^2\right) \quad (21)$$

$$z^{k+1} := \underset{z}{argmin}\left(g(z) + \frac{\mu}{2}\|Ax^{k+1} + Bz - c + u^k\|_2^2\right) \quad (22)$$

$$u^{k+1} := u^k + Ax^{k+1} + Bz^{k+1} - c \quad (23)$$

Table of Contents

Problem and equal problem

Origin

$$min \quad f(x) \tag{24}$$
$$s.t. \quad x \in C \tag{25}$$

Equal problem

$$min \quad f(x) + g(z) \tag{26}$$
$$s.t. \quad x - z = 0 \tag{27}$$

g is the indicator function of C

ADMM

$$x^{k+1} := \underset{x}{argmin}\left(f(x) + \frac{\mu}{2}\|Ax + Bz^k - c + u^k\|_2^2\right) \tag{28}$$

$$z^{k+1} := \Pi_C\left(x^{k+1} + u^k\right) \tag{29}$$

$$u^{k+1} := u^k + x^{k+1} - z^{k+1} \tag{30}$$

Table of Contents

Problem and equal problem

Problem

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

Equal problem

$$\begin{aligned} min \quad & \sum_{i=1}^{N} f_i(x_i) && (31) \\ s.t. \quad & x_i - z = 0, i = 1, 2, \cdots, N && (32) \end{aligned}$$

Method

Augmented Lagrangian

$$L_\mu(x_1, \cdots, x_N, z, y) = \sum_{i=1}^{N} \left(f_i(x_i) + y_i^T(x_i - z) + (\mu/2)\|x_i - z\|_2^2 \right)$$

ADMM

$$\begin{aligned} x_i^{k+1} &:= \underset{x_i}{argmin} \left(f_i(x_i) + y_i^{kT}(x_i - z^k) + (\mu/2)\|x_i - z^k\|_2^2 \right) && (33) \\ z^{k+1} &:= \frac{1}{N} \sum_{i=1}^{N} \left(x_i^{k+1} + (1/\mu) y_i^k \right) && (34) \\ y_i^{k+1} &:= y_i^k + \mu(x_i^{k+1} - z^{k+1}) && (35) \end{aligned}$$

Sync-ADMM

ADMM

$$x_i^{k+1} := \underset{x_i}{argmin}\left(f_i(x_i) + y_i^{kT}(x_i - z^k) + (\mu/2)\|x_i - z^k\|_2^2\right) \quad (36)$$

$$z^{k+1} := \frac{1}{N}\sum_{i=1}^{N}\left(x_i^{k+1} + (1/\mu)y_i^k\right) \quad (37)$$

$$y_i^{k+1} := y_i^k + \mu(x_i^{k+1} - z^{k+1}) \quad (38)$$

Sync-ADMM

- Each worker i is responsible for updating its (x_i, y_i) using (36) and (38).
- the master has to wait for the x_i updates from all the N workers.

Async-ADMM(Ruiliang Zhang & James T.Kwok ICML14)

Update x by Worker

Clocks

master: k Worker i: k_i

Updating x_i

$$x_i^{k+1} := \underset{x_i}{argmin} \left(f_i(x_i) + y_i^{kT}(x_i - \tilde{z}_i) + (\mu/2)\|x_i - \tilde{z}_i\|_2^2 \right)$$

$\tilde{z}_i$: the most recent z received by i from the master.

Updating z by the Master

Two rules

- Partial barrier: The master only needs to wait for S updates
- Bounded delay: every worker has to be serviced by the master at least once every τ iterations

Update

$$z^{k+1} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{x}_i + \frac{1}{N}\hat{\mu}_i\right)$$

$\hat{\mu}_i$(resp. $\hat{\mu}_i$) is the most x_i(res. μ_i) received from worker i by master.

Example

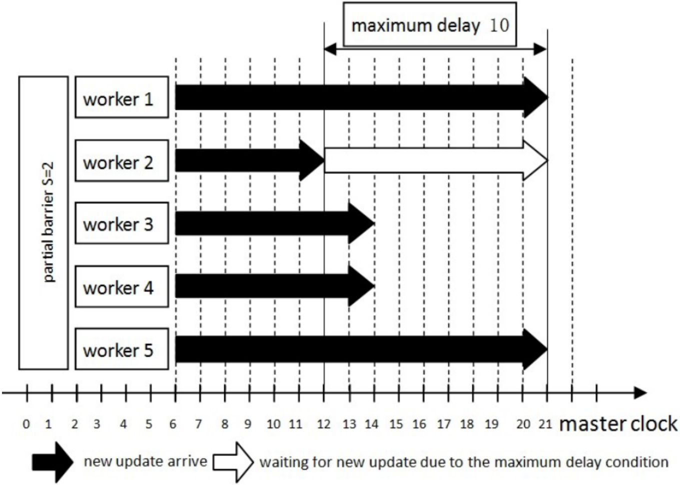

Figure: An example showing the operation of the partial barrier and bounded delay

Algorithm: master

Algorithm 3 Asynchronous ADMM (async-ADMM): Processing by the master.

initialize: $k = 0, \hat{x}_i = 0, \hat{\lambda}_i = 0, i = 1, 2, \ldots, N$.
repeat
    repeat
        wait;
    until receive a minimum of S updates from the workers **and** $\max(\tau_1, \tau_2, \ldots, \tau_N) \le \tau$;
    for worker $i \in \Phi^k$ **do**
        $\tau_i \leftarrow 1$;
        $\hat{x}_i \leftarrow$ newly received x_i from worker i;
        $\hat{\lambda}_i \leftarrow$ newly received λ_i from worker i;
    end for
    for worker $i \notin \Phi^k$ **do**
        $\tau_i \leftarrow \tau_i + 1$;
    end for
    update z^{k+1} by (7);
    broadcast z^{k+1} to all the workers in Φ^k;
    $k \leftarrow k + 1$;
until termination;
output z^k.

Table of Contents

Sharing

Problem

$$min \sum_{i=1}^{N} f_i(x_i) + g(\sum_{i=1}^{N} x_i)$$

f_i is a local cost function for subsystem i and g is the shared objective.

Equal problem

$$\begin{aligned} min \quad & \sum_{i=1}^{N} f_i(x_i) + g(\sum_{i=1}^{N} z_i) && (39) \\ s.t. \quad & x_i - z_i = 0, i = 1, 2, \cdots, N && (40) \end{aligned}$$

Scaled form of ADMM

$$x_i^{k+1} := \underset{x_i}{argmin}\left(f_i(x_i) + (\mu/2)\|x_i - z_i^k + u_i^k\|_2^2\right) \tag{41}$$

$$z^{k+1} := \underset{z}{argmin}\left(g(\sum_{i=1}^N z_i) + (\mu/2)\sum_{i=1}^N \|z_i - u_i^k - x_i^{k+1}\|_2^2\right) \tag{42}$$

$$u_i^{k+1} := u_i^k + x_i^{k+1} - z_i^{k+1} \tag{43}$$

Optimal Exchange

Problem

$$min \quad \sum_{i=1}^{N} f_i(x_i) \tag{44}$$

$$s.t. \quad \sum_{i=1}^{N} x_i = 0 \tag{45}$$

- f_i represents the cost function for subsystem i
- x_i represent quantities of commodities that are exchanged among N agent or subsystem.
- Treating it a generic constrained convex problem:

$$C = \{x \in R^{nN} | x_1 + x_2 + \cdots + x_N = 0\}$$

Conclusion

- Many opt problems can be transformed to be solved by ADMM.
- ADMM is designed for parallel or distribution.
- Write augmented Lagrangian, then ADMM formulation is easy to build.

Q & A