14 Alternating Direction Method of Multipliers / Douglas-Rachford

We are still considering optimization problems of the form

$$\min_{x\in\mathbb{R}^n} \quad f(x) + h(Ax) \tag{1}$$

where f and h are convex, the dual of which (after introducing $y = Ax$) is

$$\max_{z\in\mathbb{R}^m} -f^*(-A^T z) - h^*(z). \tag{2}$$

Last lecture we looked at the proximal gradient method applied to (2), and we compared it with the supgradient method and the augmented Lagrangian methods for (2).

One problem with the dual proximal gradient method, is that it requires the function f to be strongly convex. For the augmented Lagrangian method, a problem is that the variables (x, y) are coupled in the update step. To remedy this, we consider the ADMM algorithm which introduces a quadratic penalty in the x-update step, while maintaining the x and y-updates decoupled. It takes the following form:

$$\text{ADMM} \qquad \begin{cases} x_{k+1} = \operatorname*{argmin}_x \left\{ f(x) + \langle z_k, Ax\rangle + \frac{t}{2}\|Ax - y_k\|_2^2 \right\} \\ y_{k+1} = \operatorname*{argmin}_y \left\{ h(y) - \langle z_k, y\rangle + \frac{t}{2}\|Ax_{k+1} - y\|_2^2 \right\} \\ z_{k+1} = z_k + t(Ax_{k+1} - y_{k+1}). \end{cases} \tag{3}$$

Note that the x-update step depends on y_k, while the y-update step depends on x_{k+1}. The ADMM algorithm is a very popular algorithm for optimization, due to its versatility and wide applicability. See [BPC+11] for examples.

Just like its close relatives, the ADMM can also be seen as a particular optimization algorithm applied to the dual. This algorithm is known as the *Douglas-Rachford algorithm*, which we introduce next.

14.1 Douglas-Rachford algorithm

The Douglas-Rachford method is an algorithm to minimize the sum of two convex functions

$$\min_{x\in\mathbb{R}^n} f(x) + h(x)$$

when *both functions* f and h have a simple proximal operator. The Douglas-Rachford algorithm can be expressed in the following way:

$$\begin{cases} x_{k+1} &= \mathbf{prox}_f(y_k - z_k) \\ y_{k+1} &= \mathbf{prox}_h(x_{k+1} + z_k) \\ z_{k+1} &= z_k + (x_{k+1} - y_{k+1}). \end{cases} \tag{4}$$

Fixed points One can check that if $(x_{k+1}, y_{k+1}, z_{k+1}) = (x_k, y_k, z_k) = (\bar{x}, \bar{y}, \bar{z})$ then necessarily we have reached an optimal solution of the problem. Indeed if $z_{k+1} = z_k$ then $x_{k+1} = y_{k+1}$, i.e., $\bar{x} = \bar{y}$. From the first equation we get $0 \in \partial f(x_{k+1}) + (x_{k+1} - y_k + z_k)$ which implies, since $x_{k+1} = y_{k+1} = y_k$, $-\bar{z} \in \partial f(\bar{x})$. From the second equation, we get that $0 \in \partial h(y_{k+1}) + (y_{k+1} - x_{k+1} - z_k)$, and so using the assumptions we get $\bar{z} \in \partial h(\bar{x})$. This implies that $0 \in \partial f(\bar{x}) + \partial h(\bar{x})$ as desired.

The Douglas-Rachford operator One can combine the iterates (4) into a single sequence (w_k) given by $w_{k+1} = x_{k+1} + z_k$. Then, it is not hard to verify that the Douglas-Rachford algorithm is equivalent to

$$w_{k+1} = T(w_k)$$

where T is the Douglas-Rachford operator:

$$T(w) = \mathbf{prox}_f(2\,\mathbf{prox}_h(w) - w) + w - \mathbf{prox}_h(w). \tag{5}$$

To prove the convergence of the DR algorithm, we will prove that T is a *firmly nonexpansive* map.

Definition 14.1. A map $T : \mathbb{R}^n \to \mathbb{R}^n$ is *firmly nonexpansive* if

$$\|T(w) - T(w')\|_2^2 \le \left\langle w - w', T(w) - T(w') \right\rangle \qquad \forall w, w' \in \mathbb{R}^n.$$

We have already seen that proximal operators of convex functions are firmly nonexpansive. This allows us to prove that the Douglas-Rachford operator in (5) is firmly nonexpansive. Indeed, let $w, w' \in \mathbb{R}^n$ and let $y = \mathbf{prox}_h(w)$ and $x = \mathbf{prox}_f(2y - w)$, so that $T(w) = x + w - y$. Since $\mathbf{prox}_f$ and $\mathbf{prox}_h$ are firmly nonexpansive, we have:

$$\begin{cases} \|y - y'\|_2^2 & \le \langle y - y', w - w' \rangle \\ \|x - x'\|_2^2 & \le \langle x - x', 2(y - y') - (w - w') \rangle \,. \end{cases}$$

Now we can write

$$\begin{aligned}
\|T(w) - T(w')\|_2^2 &= \|x - x' + w - w' - (y - y')\|_2^2 \\
&= \|x - x'\|_2^2 + \|w - w'\|_2^2 + \|y - y'\|_2^2 \\
&\qquad + 2\left\langle x - x', w - w' \right\rangle - 2\left\langle x - x', y - y' \right\rangle - 2\left\langle y - y', w - w' \right\rangle \\
&\le \left\langle x - x', w - w' \right\rangle + \|w - w'\|_2^2 - \left\langle y - y', w - w' \right\rangle \\
&= \left\langle (x + w - y) - (x' + w' - y'), w - w' \right\rangle = \left\langle T(w) - T(w'), w - w' \right\rangle .
\end{aligned}$$

EXERCISE: Show that T is firmly nonexpansive, if, and only if, $T = (1/2)(I + U)$ where U is a nonexpansive map.

Now we prove a general convergence result about firmly nonexpansive iterations.

Theorem 14.1. *Assume $T : \mathbb{R}^n \to \mathbb{R}^n$ is a firmly nonexpansive map that has at least one fixed point w^*. Then the iterates $w_{k+1} = T(w_k)$ converge to some fixed point of T, and furthermore*

$$\min_{0 \le j \le k-1} \|w_j - T(w_j)\|_2^2 \le \frac{\|w_0 - w^*\|_2^2}{k}.$$

The following lemma is easy to verify.

Lemma 1. *If T is firmly nonexpansive, then $G = I - T$ is also firmly nonexpansive.*

Proof. We have $\|Gw - Gw'\|_2^2 = \|w - w'\|_2^2 + \|Tw - Tw'\|_2^2 - 2\langle w - w', Tw - Tw'\rangle \leq \|w - w'\|_2^2 - \langle w - w', Tw - Tw'\rangle = \langle w - w', Gw - Gw'\rangle$ as desired. □

We now prove the theorem.

Proof. Let w^* be any fixed point of T. Then for any w, we have

$$\begin{aligned}\|T(w) - w^*\|_2^2 - \|w - w^*\|_2^2 &\leq \langle w - w^*, T(w) - w^*\rangle - \|w - w^*\|_2^2 \\ &= \langle w - w^*, -G(w)\rangle \leq -\|G(w)\|_2^2\end{aligned} \tag{6}$$

where we used the fact that G is firmly nonexpansive, and $G(w^*) = 0$. Thus, summing these inequalities and rearranging we get

$$\sum_{i=0}^{k-1} \|G(w_i)\|_2^2 \leq \|w_0 - w^*\|_2^2.$$

Let $r_{best,k} = \min\{\|G(w_0)\|_2^2, \ldots, G(w_{k-1})\|_2^2\}$, we see that

$$r_{best,k} \leq \frac{1}{k}\sum_{i=0}^{k-1} \|G(w_i)\|_2^2 \leq \frac{\|w_0 - w^*\|_2^2}{k}.$$

and so $r_{best,k} \leq \|w_0 - w^*\|_2^2/k$.

It remains to show that (w_i) converges to a fixed point of T. The inequality (6) shows that $\|w_i - w^*\|_2$ is nonincreasing for any choice of fixed point of w^* of T; in particular (w_i) is bounded and so has a limit point $\bar{w}$. Let's show that $w_i \to \bar{w}$. First note that since $\|G(w_i)\|_2 \to 0$, and that G is continuous, we must have $G(\bar{w}) = 0$, i.e., $\bar{w}$ is a fixed point for T. It follows that the sequence $\|w_i - \bar{w}\|_2^2$ is nonincreasing, and has 0 as a limit point. Thus it must be that $\lim_i \|w_i - \bar{w}\|_2 = 0$, i.e., $w_i \to \bar{w}$. □

Relation between ADMM and Douglas-Rachford If we apply the Douglas-Rachford method to the dual problem (2) we obtain the following iterates:

$$\begin{cases}\tilde{x}_{k+1} = \mathbf{prox}_{f^* \circ -A^T}(\tilde{y}_k - \tilde{z}_k) \\ \tilde{y}_{k+1} = \mathbf{prox}_{h^*}(\tilde{x}_{k+1} + \tilde{z}_k) \\ \tilde{z}_{k+1} = \tilde{z}_k + (\tilde{x}_{k+1} - \tilde{y}_{k+1})\end{cases}$$

These equations can be simplified using Moreau's identity, and its generalization:

$$\mathbf{prox}_{f^* \circ A^T}(x) = x - A \operatorname*{argmin}_u \{f(u) + (1/2)\|Au - x\|_2^2\}. \tag{7}$$

Using this identity, we get:

$$\begin{cases}\tilde{x}_{k+1} &= \tilde{y}_k - \tilde{z}_k + A \operatorname{argmin}_x \left\{f(x) + (1/2)\|Ax - (\tilde{z}_k - \tilde{y}_k)\|_2^2\right\} \\ \tilde{y}_{k+1} &= \tilde{x}_{k+1} + \tilde{z}_k - \operatorname{argmin}_y \left\{h(y) + (1/2)\|y - (\tilde{x}_{k+1} + \tilde{z}_k)\|_2^2\right\} \\ \tilde{z}_{k+1} &= \tilde{z}_k + (\tilde{x}_{k+1} - \tilde{y}_{k+1}).\end{cases}$$

By a suitable change of variables, the iterates can be shown to be equivalent to the ADMM method of (3). Indeed, by calling x_{k+1} the argmin in the first line, y_{k+1} the argmin in the second line, and $z_k = \tilde{y}_k$, we see that the iterations above can be written as (check!)

$$\begin{cases} x_{k+1} &= \operatorname{argmin}_x \left\{ f(x) + (1/2)\|Ax - (y_k - z_k)\|_2^2 \right\} \\ y_{k+1} &= \operatorname{argmin}_y \left\{ h(y) + (1/2)\|y - (Ax_{k+1} + z_k)\|_2^2 \right\} \\ x_{k+1} &= z_k + (Ax_{k+1} - y_{k+1}). \end{cases}$$

It is easy to see that these are the same as (3) with $t = 1$ (the case with general t can be obtained by appropriately scaling the functions f and h).

14.2 Historical note on the Douglas-Rachford algorithm

The Douglas-Rachford algorithm was invented in the 1950s [DR56] as a method to solve the heat equation, i.e.,

$$\frac{\partial u}{\partial t} = \nabla_x^2 u + \nabla_y^2 u.$$

Let $A = -\nabla_x^2$ and $B = -\nabla_y^2$, so that the equation can be written as $u_t = -Au - Bu$. We assume we have discretized along space variables x and y using finite differences; as such, with a suitable ordering of the nodes, A and B are tridiagonal. If we use the backward Euler method to solve this problem we end up with the following scheme:

$$u^{n+1} = u^n + \lambda(-Au^{n+1} - Bu^{n+1})$$

i.e., $u^{n+1} = (I + \lambda(A + B))^{-1}u^n$ where $\lambda > 0$ is the time step. Solving a linear system with $I + \lambda(A + B)$ can be expensive, unlike solving linear systems with $I + \lambda A$ and $I + \lambda B$ which are much easier because the latter are tridiagonal after suitable permutation of the nodes (different for A and B). Splitting schemes have thus been developed to address this need. There are many possible splittings one can do:

- One possibility for splitting is the forward backward splitting where we use forward Euler on B, and backward Euler on A (or vice-versa):

  $$u^{n+1} = u^n + \lambda(-Au^{n+1} - Bu^n)$$

  This only requires solving a linear system involving $I + \lambda A$.

- Another possibility, proposed by Peaceman-Rachford, is to alternate the roles of A and B in forward-backward splitting, i.e.,

  $$\begin{cases} u^{n+1/2} = u^n + \lambda(-Au^{n+1/2} - Bu^n) \\ u^{n+1} = u^{n+1/2} + \lambda(-Au^{n+1/2} - Bu^{n+1}) \end{cases}$$

  This requires solving, at each time step, a linear system with $I + \lambda A$ and a linear system with $I + \lambda B$.

- The third one, proposed by Douglas-Rachford, proceeds as follows. Even though $(I+\lambda A+\lambda B)$ is difficult to invert, we can see that $(I + \lambda A + \lambda B + \lambda^2 AB)$ is actually easy to invert because the latter is simply $(I+\lambda A)(I+\lambda B)$. So this motivates us to consider the following altered backward difference formula:

$$u^{n+1} = u^n + \lambda(-Au^{n+1} - Bu^{n+1}) + \underbrace{\lambda^2 AB(u^n - u^{n+1})}.$$

This reduces to $(I+\lambda A)(I+\lambda B)u^{n+1} = u^n + \lambda^2 ABu^n$, which again only requires $(I+\lambda A)^{-1}$ and $(I+\lambda B)^{-1}$.

Extension to nonlinear operators The above applies to any positive linear operators A and B, and not just to the Laplacian. In fact, these methods were shown to be convergent for nonlinear *maximal monotone* operators A, B by Lions and Mercier in [LM79]. The latter Douglas-Rachford can be written as

$$u^{n+1} = Tu^n$$

where $T = (I+\lambda B)^{-1}(I+\lambda A)^{-1}(I+\lambda^2 AB)$. If we write $I + \lambda^2 AB = (I+\lambda A)\lambda B + I - \lambda B$, we get

$$T = (I+\lambda B)^{-1}\left[(I+\lambda A)^{-1}(I-\lambda B) + \lambda B\right]$$

Call $(I+\lambda B)^{-1}v^n = u^n$, then we get

$$\begin{aligned} v^{n+1} &= ((I+\lambda A)^{-1}(I-\lambda B) + \lambda B)(I+\lambda B)^{-1}v^n \\ &= \left[(I+\lambda A)^{-1}(2(I+\lambda B)^{-1} - I) + I - (I+\lambda B)^{-1}\right] v^n. \end{aligned} \tag{8}$$

where we used the fact that $(I-\lambda B)(I+\lambda B)^{-1} = 2(I+\lambda B)^{-1} - I$, and $\lambda B(I+\lambda B)^{-1} = I - (I+\lambda B)^{-1}$. If $A = \partial f$ and $B = \partial h$, then $(I+\lambda A)^{-1} = \mathbf{prox}_{\lambda f}$ and $(I+\lambda B)^{-1} = \mathbf{prox}_{\lambda h}$ and so the equation above is precisely the Douglas-Rachford iteration (5)! For a nice survey of monotone operator methods in optimization, see [RB16].

References

[BPC+11] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, Jonathan Eckstein, et al. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011. 1

[DR56] Jim Douglas and Henry H Rachford. On the numerical solution of heat conduction problems in two and three space variables. *Transactions of the American mathematical Society*, 82(2):421–439, 1956. 4

[LM79] Pierre-Louis Lions and Bertrand Mercier. Splitting algorithms for the sum of two nonlinear operators. *SIAM Journal on Numerical Analysis*, 16(6):964–979, 1979. 5

[RB16] Ernest K Ryu and Stephen Boyd. Primer on monotone operator methods. *Appl. Comput. Math*, 15(1):3–43, 2016. 5