Bregman Method from PEP Perspective

Zhenghao Xu

Zhejiang University

August 8, 2021

Table of Contents

Progress

Finished:

- BPG's PEP formulation (Gram matrix form)
- Numerical experiment for BPG (modified from [Dragomir et al., 2019])

In progress:

- Dual relaxation
- Analytical dual solution
- General first-order Bregman (proximal) gradient formulation

Idea

Two ways of dealing with PEP:

- Kim & Fessler: relax, dualize, analytical solution
- Taylor: Gram matrix, interpolation, tight, numerical induction

Idea

Two ways of dealing with PEP:

- Kim & Fessler: relax, dualize, analytical solution
- Taylor: Gram matrix, interpolation, tight, numerical induction

Kim & Fessler actually use some sort of descent lemma to perform relaxation, thus for simple method the dual solution can be guessed through ordinary convergence proof.
Their explicit relaxation are more valuable for general methods, where feasible solution is not easy to see through.

Idea

Two ways of dealing with PEP:

- Kim & Fessler: relax, dualize, analytical solution
- Taylor: Gram matrix, interpolation, tight, numerical induction

Kim & Fessler actually use some sort of descent lemma to perform relaxation, thus for simple method the dual solution can be guessed through ordinary convergence proof.
Their explicit relaxation are more valuable for general methods, where feasible solution is not easy to see through.

To induce analytical bound, Taylor et al. also perform relaxation.
In most cases only the inequalities between nearby steps, each step and the minimizer are considered.
No explanation in their paper.

Idea

We want to

- avoid using existing descent lemma (otherwise the proof is somehow limited)
- formulate general Bregman proximal method
- (if possible) accelerate or prove optimality

Idea (current)

In PG we have

$$p_L(x) = \arg\min_u \{f(x) + \langle \nabla f(x), u - x \rangle + \frac{L}{2}\|u - x\|^2 + \phi(u)\}$$
$$= \arg\min_u \left\{ \phi(u) + \frac{L}{2}\left\| u - (x - \frac{1}{L}\nabla f(x)) \right\|^2 \right\}$$

Algorithm Class FSFOM

Input: $f \in \mathcal{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_0 \in \mathbb{R}^d$, $\boldsymbol{y}_0 = \boldsymbol{x}_0$.

For $i = 0, \dots, N-1$

$$\boldsymbol{x}_{i+1} = \mathbf{p}_L(\boldsymbol{y}_i) = \boldsymbol{y}_i - \frac{1}{L}\tilde{\nabla}_L F(\boldsymbol{y}_i)$$

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + \sum_{k=0}^{i} h_{i+1,k}\,(\boldsymbol{x}_{k+1} - \boldsymbol{y}_k) = \boldsymbol{y}_i - \frac{1}{L}\sum_{k=0}^{i} h_{i+1,k}\,\tilde{\nabla}_L F(\boldsymbol{y}_k).$$

Idea (current)

Algorithm Class FSFOM

Input: $f \in \mathcal{F}_L(\mathbb{R}^d)$, $\boldsymbol{x}_0 \in \mathbb{R}^d$, $\boldsymbol{y}_0 = \boldsymbol{x}_0$.

For $i = 0, \ldots, N-1$

$$\boldsymbol{x}_{i+1} = \mathbf{p}_L(\boldsymbol{y}_i) = \boldsymbol{y}_i - \frac{1}{L}\tilde{\nabla}_L F(\boldsymbol{y}_i)$$

$$\boldsymbol{y}_{i+1} = \boldsymbol{y}_i + \sum_{k=0}^{i} h_{i+1,k}\,(\boldsymbol{x}_{k+1} - \boldsymbol{y}_k) = \boldsymbol{y}_i - \frac{1}{L}\sum_{k=0}^{i} h_{i+1,k}\,\tilde{\nabla}_L F(\boldsymbol{y}_k).$$

Composite gradient mapping:

$$\tilde{\nabla}_L F(x) = -L(p_L(x) - x)$$

We might need an analogous definition for Bregman case

$$p_L(x) = \arg\min_u \{f(x) + \langle \nabla f(x), u - x\rangle + LD_h(u, x) + \phi(u)\}$$

Idea (current)

In the Bregman case, it is easy to induce a definition with mirror points $\nabla h(x)$ from

$$-L(\nabla h(\mathrm{p}_L(x)) - \nabla h(x)) = \phi'(\mathrm{p}_L(x)) + \nabla f(x)$$

In PEP, it is easy to track. But not for method formulation.

Idea (current)

Definition 4 An algorithm $\mathcal{A}$ is called a Bregman first-order algorithm if, for a given problem instance $(f, h) \in \mathcal{B}_L$ and number of iterations $T \in \mathbb{N}$, it generates at each time step $t \in \{0, \dots, T\}$, a set of primal points $\mathcal{X}_t$ and dual points $\mathcal{Y}_t$ from the following process:

1. Set $\mathcal{X}_0 = \{x_0\}$, where $x_0 \in \operatorname{int} \operatorname{dom} h$ is some initialization point, and $\mathcal{Y}_0 = \{\nabla f(x_0), \nabla h(x_0)\}$.
2. For each $t = 1, \dots T$, perform one of the two following operations:
   - either call the **primal oracle** $(\nabla f, \nabla h)$ at some point x_t chosen such as
   $$x_t \in \operatorname{Span}(\mathcal{X}_{t-1}) \cap \operatorname{dom} \nabla h$$
   and update the dual set as
   $$\mathcal{Y}_t = \mathcal{Y}_{t-1} \cup \{\nabla f(x_t), \nabla h(x_t)\}.$$
   - Or call the **mirror oracle** ∇h^* at some dual point y_t chosen such as
   $$y_t \in \operatorname{Span}(\mathcal{Y}_{t-1})$$
   with
   $$\nabla h^*(y_t) = \operatorname*{argmin}_{u \in C} h(u) - \langle y_t, u \rangle$$
   and update the primal set as
   $$\mathcal{X}_t = \mathcal{X}_{t-1} \cup \{\nabla h^*(y_t)\}.$$
3. Output some point $x_T \in \operatorname{Span}(\mathcal{X}_T)$.

Need to translate into explicit formulation.

Bibliography I

Dragomir, R., Taylor, A. B., d'Aspremont, A., and Bolte, J. (2019). Optimal complexity and certification of bregman first-order methods. *ArXiv*, abs/1911.08510.