凸优化的一些典型问题及其求解方法

一. 凸优化和变分不等式框架下的邻近点算法

何 炳 生

南方科技大学数学系　　南京大学数学系

Homepage: maths.nju.edu.cn/~hebma

统计与数据科学前沿理论及应用 教育部重点实验室

华东师范大学计算机科学与技术学院

2020 年 11 月 28 – 29 日

中学的数理基础　必要的社会实践

普通的大学数学　一般的优化常识

两大基本概念：变分不等式　和　邻近点 (PPA) 算法

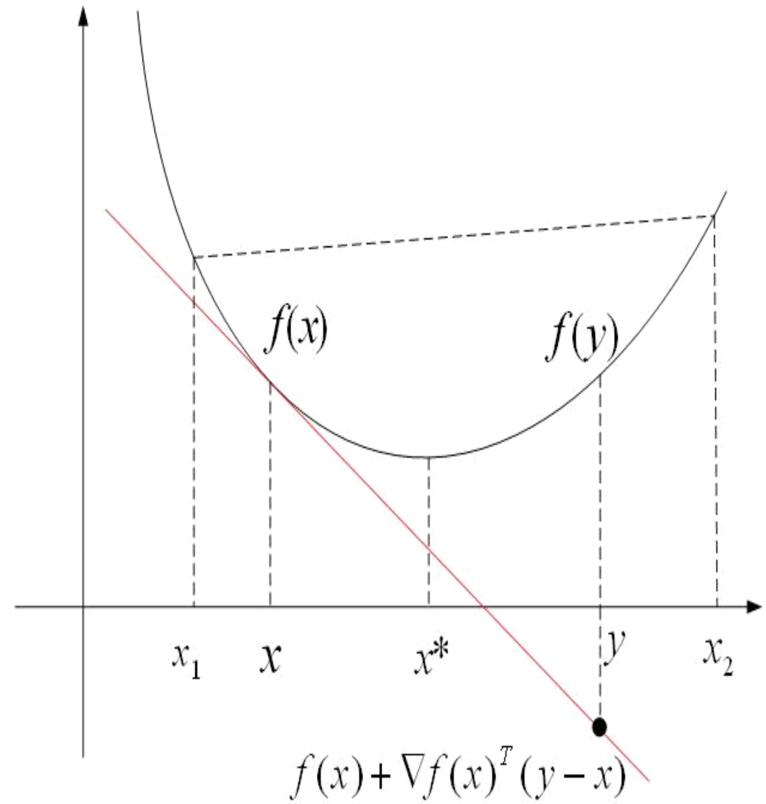
变分不等式(VI) 是瞎子爬山的数学表达形式

邻近点算法(PPA) 是步步为营 稳扎稳打的求解方法.

A function $f(x)$ is convex iff

$$f((1-\theta)x+\theta y) \leq (1-\theta)f(x)+\theta f(y)$$

$$\forall \theta \in [0,1].$$



Convex function

Properties of convex function

- $f \in \mathcal{C}^1$. f is convex iff

  $$f(y) - f(x) \geq \nabla f(x)^T (y-x).$$

  Thus, we have also

  $$f(x) - f(y) \geq \nabla f(y)^T (x-y).$$

- Adding above two inequalities, we get

  $$(y-x)^T (\nabla f(y) - \nabla f(x)) \geq 0.$$

- $f \in \mathcal{C}^1$, ∇f is monotone. $f \in \mathcal{C}^2$, $\nabla^2 f(x)$ is positive semi-definite.
- Any local minimum of a convex function is a global minimum.

1 一些线性约束凸优化问题

$$
\begin{aligned}
\min \quad & \theta(x) \\
\text{s.t} \quad & Ax = b \\
& x \in \mathcal{X}
\end{aligned}
\tag{1.1}
$$

1.1 图像处理中的凸优化问题

Image deblurring

Blurry can be produced by

defocus the camera's lens, the moving object, turbulence in the air, $\cdots$

Notations: $\mathbf{g}$ — observation, $\mathbf{f}$ — ideal image;

$\mathcal{U}$ — restriction on pixels, e.g., $\mathcal{U} = \{u \mid 0 \le u \le 255\}$

$\mathbf{g} = H\mathbf{f}$, $\quad H$ — blur matrix .

Model

$$\min \ \{\|\nabla \mathbf{f}\|_1 \ \mid \ H\mathbf{f} = \mathbf{g}, \ \ \mathbf{f} \in \mathcal{U}\}$$

original image

blurred image

restored image

Image inpainting

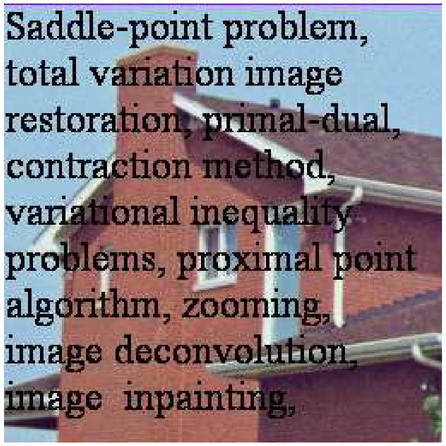
Some pixels are missing in image. Partial information of image is available

$$\mathbf{g} = S\,\mathbf{f}, \quad S \text{ — mask (missing pixels)}$$

Model

$$\min \ \{\|\nabla \mathbf{f}\|_1 \ \mid \ S\,\mathbf{f} = \mathbf{g}, \ \ \mathbf{f} \in \mathcal{U}\}$$

original image



missing pixel image

restored image

Image zooming and super-resolution

Produce a high-resolution (HR) image by its low-resolution (LR) image(s)

$$\mathbf{g} = D\,\mathbf{f}, \quad \mathbf{f} \text{ — HR image, } \mathbf{g} \text{ — LR image, } D \text{ — down-sampling}$$

Model $$\min \ \{\|\nabla \mathbf{f}\|_1 \ | \ D\mathbf{f} = \mathbf{g}, \ \ \mathbf{f} \in \mathcal{U}\}$$

LR image

HR image

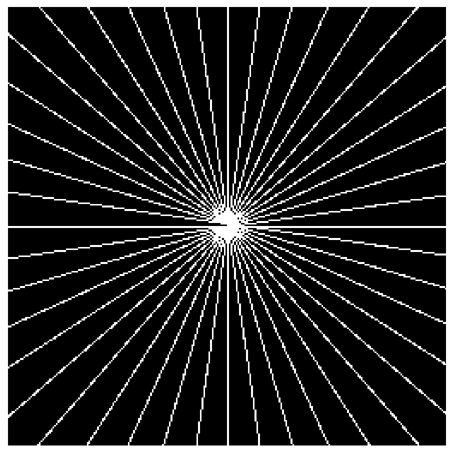
Magnetic resonance imaging (MRI)

Reconstruct a medical image by sampling its Fourier coefficients partially

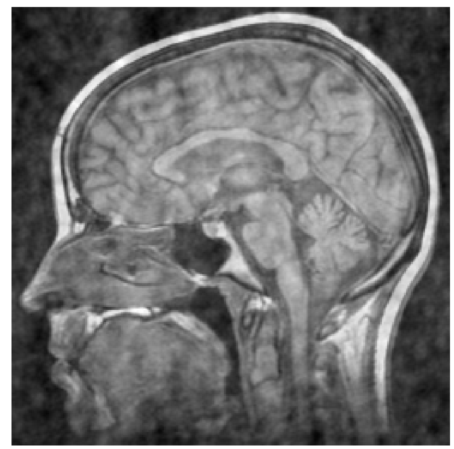
$$\mathcal{F}\mathbf{g} = P\mathcal{F}\mathbf{f}, \quad P \text{ — sampling mask, } \mathcal{F} \text{ — Fourier transform}$$

Model $\min \; \{\|\nabla \mathbf{f}\|_1 \;\mid\; P\mathcal{F}\mathbf{f} = \mathcal{F}\mathbf{g}\}$

medical image



sampling mask



reconstruction

1.2 Problems in matrix optimization

Example 1 of the problem (1.1): Finding the nearest correlation matrix

A positive semi-definite matrix, whose each diagonal element is equal 1, is called the correlation matrix.

For given symmetric $n \times n$ matrix C, the mathematical form of finding the nearest correlation matrix X is

$$\min\{\tfrac{1}{2}\|X - C\|_F^2 \,|\, \text{diag}(X) = e,\ X \in S_+^n\}, \tag{1.2}$$

where S_+^n is the positive semi-definite cone and e is a n-vector whose each element is equal 1.

The problem (1.2) is a concrete problem of type (1.1).

Example 2 of the problem (1.1): The matrix completion problem

Let M be a given $m \times n$ matrix, Π is the elements indices set of M,

$$\Pi \subset \{(ij)\,|\,i \in \{1,2,\ldots,m\},\ j \in \{1,2,\ldots,n\}\}.$$

The mathematical form of the matrix completion problem is relaxed to

$$\min\{\|X\|_* \mid X_{ij} = M_{ij},\ (ij) \in \Pi\}, \tag{1.3}$$

where $\|\cdot\|_*$ is the nuclear norm–the sum of the singular values of a given matrix. The problem (1.3) is a convex optimization of form (1.1).

The matrix A in (1.1) for the linear constraints

$$X_{ij} = M_{ij},\ (ij) \in \Pi,$$

is a projection matrix, and thus $\|A^T A\| = 1$.

2 Optimization problem and VI

2.1 Differential convex optimization in Form of VI

Let $\Omega \subset \Re^n$, we consider the convex minimization problem

$$\min\{f(x) \mid x \in \Omega\}. \tag{2.1}$$

What is the first-order optimal condition ?

$x^* \in \Omega^* \quad \Leftrightarrow \quad x \in \Omega$ and any feasible direction is not descent direction.

Optimal condition in variational inequality form

- $S_d(x^*) = \{s \in \Re^n \mid s^T \nabla f(x^*) < 0\} \;=\;$ Set of the descent directions.
- $S_f(x^*) = \{s \in \Re^n \mid s = x - x^*,\ x \in \Omega\} \;=\;$ Set of feasible directions.

$$\boxed{x^* \in \Omega^* \quad \Leftrightarrow \quad x^* \in \Omega \quad \text{and} \quad S_f(x^*) \cap S_d(x^*) = \emptyset.}$$

瞎子爬山判定山顶的准则是: 所有可行方向都不再是上升方向

The optimal condition can be presented in a variational inequality (VI) form:

$$x^* \in \Omega, \quad (x - x^*)^T F(x^*) \geq 0, \quad \forall x \in \Omega, \tag{2.2}$$

where $F(x) = \nabla f(x)$.

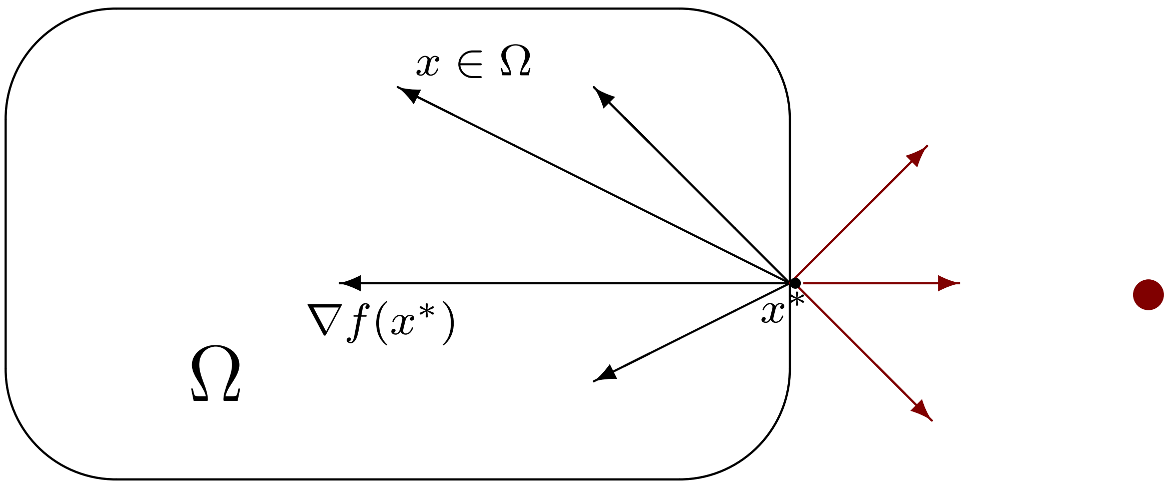


Fig. 1.1 Differential Convex Optimization and VI

Since $f(x)$ is a convex function, we have

$f(y) \geq f(x) + \nabla f(x)^T (y - x)$ and thus $(x - y)^T (\nabla f(x) - \nabla f(y)) \geq 0.$

We say the gradient ∇f of the convex function f is a monotone operator.

通篇我们需要用到的大学数学　主要是基于微积分学的一个引理

$$\min\{\theta(x)|x\in\mathcal{X}\},\quad x^*\in\mathcal{X},\qquad \theta(x)-\theta(x^*)\ge 0,\qquad \forall\, x\in\mathcal{X};$$

$$\min\{f(x)|x\in\mathcal{X}\},\quad x^*\in\mathcal{X},\quad (x-x^*)^T\nabla f(x^*)\ge 0,\quad \forall\, x\in\mathcal{X}.$$

上面的凸优化最优性条件是最基本的, 合在一起就是下面的引理:

Lemma 1 *Let $\mathcal{X}\subset\Re^n$ be a closed convex set, $\theta(x)$ and $f(x)$ be convex functions and $f(x)$ is differentiable. Assume that the solution set of the minimization problem $\min\{\theta(x)+f(x)\,|\,x\in\mathcal{X}\}$ is nonempty. Then,*

$$x^*\in\arg\min\{\theta(x)+f(x)\,|\,x\in\mathcal{X}\} \tag{2.3a}$$

if and only if

$$x^*\in\mathcal{X},\ \ \theta(x)-\theta(x^*)+(x-x^*)^T\nabla f(x^*)\ge 0,\ \ \forall\, x\in\mathcal{X}. \tag{2.3b}$$

2.2 Min-Max Problem

The min-max problem has the following mathematical form

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \ \Phi(x,y) := \theta_1(x) - y^T Ax - \theta_2(y), \tag{2.4}$$

where $A \in \Re^{m\times n}$, $\theta_1(x) : \Re^n \to \Re$, $\theta_2(y) : \Re^m \to \Re$ are convex functions. Let (x^*, y^*) be the solution of (2.4), then we have

$$\left\{ \begin{array}{ll} x^* \in \mathcal{X}, \ \ \Phi(x,y^*) - \Phi(x^*,y^*) \geq 0, \ \ \forall\, x \in \mathcal{X}, & \text{(2.5a)} \\ y^* \in \mathcal{Y}, \ \ \Phi(x^*,y^*) - \Phi(x^*,y) \geq 0, \ \ \forall\, y \in \mathcal{Y}. & \text{(2.5b)} \end{array} \right.$$

Using the notation of $\Phi(x,y)$, it can be written as

$$\left\{ \begin{array}{lll} x^* \in \mathcal{X}, & \theta_1(x) - \theta_1(x^*) + (x - x^*)^T(-A^Ty^*) \geq 0, & \forall\, x \in \mathcal{X}, \\ y^* \in \mathcal{Y}, & \theta_2(y) - \theta_2(y^*) \ + \ (y - y^*)^T(\ Ax^*) \ \ \geq 0, & \forall\, y \in \mathcal{Y}. \end{array} \right.$$

It can be written as a variational inequality: $u \in \Omega$,

$$\theta(u) - \theta(u^*) + (u - u^*)^T F(u^*) \geq 0, \ \forall\, u \in \Omega, \tag{2.6}$$

where

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \theta(u) = \theta_1(x) + \theta_2(y), \quad F(u) = \begin{pmatrix} -A^T y \\ Ax \end{pmatrix}$$

and $\Omega = \mathcal{X} \times \mathcal{Y}$.

If $\theta_1(x)$ and $\theta_2(y)$ are differentiable, by setting $\nabla\theta_1(x) = f(x)$, $\nabla\theta_2(y) = g(y)$, the solution of (2.4) should satisfy

$$\begin{cases} x^* \in \mathcal{X}, \quad (x - x^*)^T (f(x^*) - A^T y^*) \geq 0, \quad \forall\, x \in \mathcal{X}, \\ y^* \in \mathcal{Y}, \quad (y - y^*)^T (g(y^*) + Ax^*) \quad \geq 0, \quad \forall\, y \in \mathcal{Y}. \end{cases}$$

The compact form of the above variational inequality can be written as

$$u^* \in \Omega, \quad (u - u^*)^T F(u^*) \geq 0, \quad \forall u \in \Omega$$

where

$$F(u) = \begin{pmatrix} f(x) - A^T y \\ g(y) + Ax \end{pmatrix} = \begin{pmatrix} f(x) \\ g(y) \end{pmatrix} + \begin{pmatrix} 0 & -A^T \\ A & 0 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}.$$

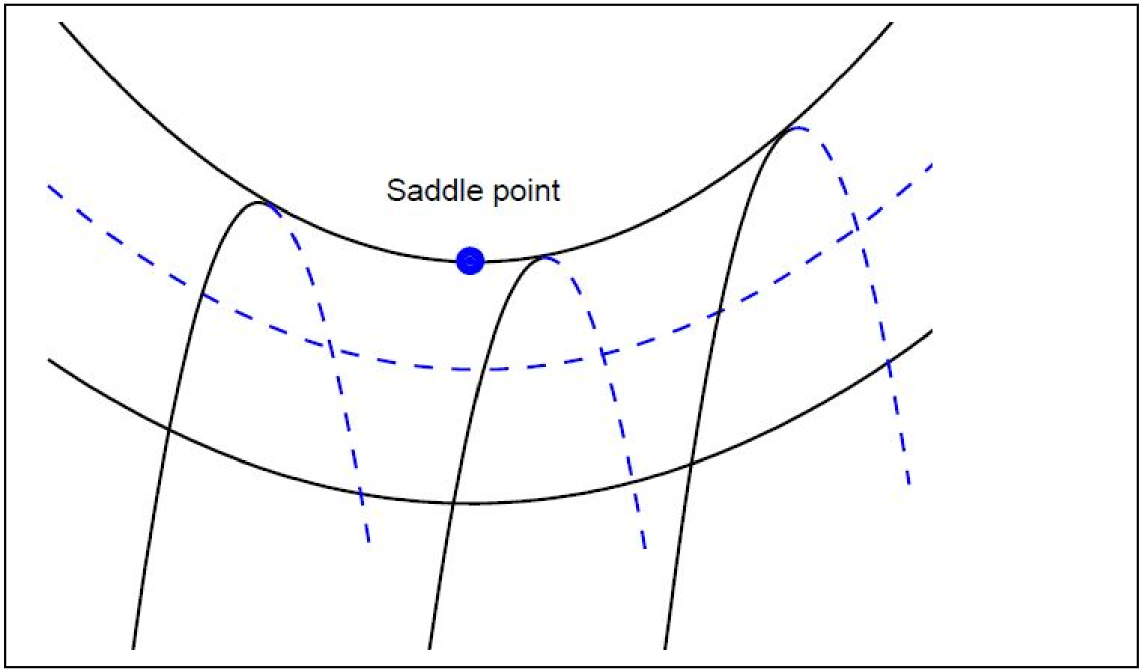
2.3 Linearly constrained Optimization in form of VI

We consider the linearly constrained convex optimization problem

$$\min\{\theta(u) \mid \mathcal{A}u = b,\ u \in \mathcal{U}\}. \tag{2.7}$$

The Lagrange function of (2.7) is

$$L(u,\lambda) = \theta(u) - \lambda^T(\mathcal{A}u - b), \qquad (u,\lambda) \in \mathcal{U} \times \Re^m. \tag{2.8}$$



A pair of (u^*, λ^*) is called a saddle point if

$$L_{\lambda\in\Re^m}(u^*,\lambda) \le L(u^*,\lambda^*) \le L_{u\in\mathcal{U}}(u,\lambda^*).$$

The above inequalities can be written as

$$\left\{\begin{array}{ll} u^* \in \mathcal{U}, \;\; L(u,\lambda^*) - L(u^*,\lambda^*) \ge 0, \;\; \forall\, u \in \mathcal{U}, & \text{(2.9a)} \\ \lambda^* \in \Lambda, \;\; L(u^*,\lambda^*) - L(u^*,\lambda) \ge 0, \;\; \forall\, \lambda \in \Lambda. & \text{(2.9b)} \end{array}\right.$$

According to the definition of $L(u,\lambda)$ (see(2.8)),

$$\begin{aligned} & L(u,\lambda^*) - L(u^*,\lambda^*) \\ & = \;\; [\theta(u) - (\lambda^*)^T(\mathcal{A}u - b)] - [\theta(u^*) - (\lambda^*)^T(\mathcal{A}u^* - b)] \\ & = \;\; \theta(u) - \theta(u^*) + (u-u^*)^T(-\mathcal{A}^T\lambda^*) \end{aligned}$$

it follows from (2.9a) that

$$u^* \in \mathcal{U}, \;\; \theta(u) - \theta(u^*) + (u-u^*)^T(-\mathcal{A}^T\lambda^*) \ge 0, \;\; \forall\, u \in \mathcal{U}. \qquad \text{(2.10)}$$

Similarly, for (2.9b), since

$$\begin{aligned} & L(u^*,\lambda^*) - L(u^*,\lambda) \\ = \quad & [\theta(u^*) - (\lambda^*)^T(\mathcal{A}u^* - b)] - [\theta(u^*) - (\lambda)^T(\mathcal{A}u^* - b)] \\ = \quad & (\lambda - \lambda^*)^T(\mathcal{A}u^* - b), \end{aligned}$$

we have

$$\lambda^* \in \Re^m, \quad (\lambda - \lambda^*)^T(\mathcal{A}u^* - b) \ge 0, \quad \forall\, \lambda \in \Re^m. \qquad (2.11)$$

Notice that the above expression is equivalent to

$$\mathcal{A}u^* = b.$$

Writing (2.10) and (2.11) together, we get the following variational inequality:

$$\begin{cases} u^* \in \mathcal{U}, & \theta(u) - \theta(u^*) + (u - u^*)^T(-\mathcal{A}^T\lambda^*) \ge 0, \quad \forall\, u \in \mathcal{U}, \\ \lambda^* \in \Re^m, & (\lambda - \lambda^*)^T(\mathcal{A}u^* - b) \ge 0, \quad \forall\, \lambda \in \Re^m. \end{cases}$$

Using a more compact form, the saddle-point can be characterized as the solution of the following VI:

$$w^* \in \Omega, \quad \theta(u) - \theta(u^*) + (w - w^*)^T F(w^*) \ge 0, \quad \forall\, w \in \Omega. \tag{2.12}$$

where

$$w = \begin{pmatrix} u \\ \lambda \end{pmatrix}, \quad F(w) = \begin{pmatrix} -\mathcal{A}^T\lambda \\ \mathcal{A}u - b \end{pmatrix} \text{ and } \Omega = \mathcal{U} \times \Re^m. \tag{2.13}$$

Because F is a affine operator and

$$F(w) = \begin{pmatrix} 0 & -\mathcal{A}^T \\ \mathcal{A} & 0 \end{pmatrix}\begin{pmatrix} u \\ \lambda \end{pmatrix} - \begin{pmatrix} 0 \\ b \end{pmatrix}.$$

The matrix is skew-symmetric, we have

$$(w - \tilde{w})^T (F(w) - F(\tilde{w})) \equiv 0.$$

Convex optimization problem with two separable functions

$$\min\{\theta_1(x) + \theta_2(y) \mid Ax + By = b,\ x \in \mathcal{X}, y \in \mathcal{Y}\}. \tag{2.14}$$

This is a special problem of (2.7) with

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \mathcal{U} = \mathcal{X} \times \mathcal{Y}, \quad \mathcal{A} = (A, B).$$

The Lagrangian function of the problem (2.14) is

$$L^2(x, y, \lambda) = \theta_1(x) + \theta_2(y) - \lambda^T(Ax + By - b).$$

The same analysis tells us that the saddle point is a solution of the following VI:

$$w^* \in \Omega, \quad \theta(u) - \theta(u^*) + (w - w^*)^T F(w^*) \ge 0, \quad \forall\, w \in \Omega. \tag{2.15}$$

where

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \qquad \theta(u) = \theta_1(x) + \theta_2(y), \tag{2.16a}$$

$$w = \begin{pmatrix} x \\ y \\ \lambda \end{pmatrix}, \qquad F(w) = \begin{pmatrix} -A^T\lambda \\ -B^T\lambda \\ Ax + By - b \end{pmatrix}, \tag{2.16b}$$

and

$$\Omega = \mathcal{X} \times \mathcal{Y} \times \Re^m. \tag{2.16c}$$

The affine operator $F(w)$ has the form

$$F(w) = \begin{pmatrix} 0 & 0 & -A^T \\ 0 & 0 & -B^T \\ A & B & 0 \end{pmatrix} \begin{pmatrix} x \\ y \\ \lambda \end{pmatrix} - \begin{pmatrix} 0 \\ 0 \\ b \end{pmatrix}.$$

Again, we have

$$(w - \tilde{w})^T (F(w) - F(\tilde{w})) \equiv 0.$$

Convex optimization problem with three separable functions

$$\min\{\theta_1(x)+\theta_2(y)+\theta_3(z)\,|\,Ax+By+Cz=b,\; x\in\mathcal{X}, y\in\mathcal{Y}, z\in\mathcal{Z}\},$$

which is a special problem of (2.7). The Lagrangian function is

$$L^3(x,y,z,\lambda)=\theta_1(x)+\theta_2(y)+\theta_3(z)-\lambda^T(Ax+By+Cz-b).$$

The same analysis tells us that the saddle point is a solution of the following VI:

$$w^*\in\Omega,\quad \theta(u)-\theta(u^*)+(w-w^*)^TF(w^*)\ge 0,\quad \forall\, w\in\Omega.$$

where

$$w=\begin{pmatrix} x\\ y\\ z\\ \lambda\end{pmatrix},\quad u=\begin{pmatrix} x\\ y\\ z\end{pmatrix},\quad F(w)=\begin{pmatrix} -A^T\lambda\\ -B^T\lambda\\ -C^T\lambda\\ Ax+By+Cz-b\end{pmatrix},$$

$$\theta(u)=\theta_1(x)+\theta_2(y)+\theta_3(z),\qquad\qquad \Omega=\mathcal{X}\times\mathcal{Y}\times\mathcal{Z}\times\Re^m.$$

3 Proximal point algorithms and its Beyond

Lemma 2 *Let the vectors* $a, b \in \Re^n$, $H \in \Re^{n\times n}$ *be a positive definite matrix. If* $b^T H(a-b) \geq 0$, *then we have*

$$\|b\|_H^2 \leq \|a\|_H^2 - \|a-b\|_H^2. \tag{3.1}$$

The assertion follows from $\|a\|^2 = \|b + (a-b)\|^2 \;\geq\; \|b\|^2 + \|a-b\|^2$.

3.1 Proximal point algorithms for convex optimization

Convex Optimization Now, let us consider the *simple* convex optimization

$$\min\{\theta(x) + f(x) \mid x \in \mathcal{X}\}, \tag{3.2}$$

where $\theta(x)$ and $f(x)$ are convex but $\theta(x)$ is not necessary smooth, $\mathcal{X}$ is a closed convex set.

For solving (3.2), the k-th iteration of the proximal point algorithm (abbreviated to

PPA) [16, 19] begins with a given x^k, offers the new iterate x^{k+1} via the recursion

$$x^{k+1} = \text{Argmin}\{\theta(x) + f(x) + \frac{r}{2}\|x - x^k\|^2 \mid x \in \mathcal{X}\}. \tag{3.3}$$

Since x^{k+1} is the optimal solution of (3.3), it follows from Lemma 1 that

$$\theta(x) - \theta(x^{k+1}) + (x - x^{k+1})^T\{\nabla f(x^{k+1}) + r(x^{k+1} - x^k)\} \ge 0, \ \forall x \in \mathcal{X}. \tag{3.4}$$

Setting $x = x^*$ in the above inequality, it follows that

$$(x^{k+1} - x^*)^T(x^k - x^{k+1}) \ge \theta(x^{k+1}) - \theta(x^*) + (x^{k+1} - x^*)^T\nabla f(x^{k+1}).$$

Since $(x^{k+1} - x^*)^T\nabla f(x^{k+1}) \ge (x^{k+1} - x^*)^T\nabla f(x^*) \ge 0$, it follows that

$$(x^{k+1} - x^*)^T(x^k - x^{k+1}) \ge 0. \tag{3.5}$$

Let $a = x^k - x^*$ and $b = x^{k+1} - x^*$ and using Lemma 2, we obtain

$$\|\boldsymbol{x^{k+1} - x^*}\|^{\boldsymbol{2}} \le \|\boldsymbol{x^k - x^*}\|^{\boldsymbol{2}} - \|\boldsymbol{x^k - x^{k+1}}\|^{\boldsymbol{2}}, \tag{3.6}$$

which is the nice convergence property of Proximal Point Algorithm.

We write the problem (3.2) and its PPA (3.3) in VI form

For the optimization problem (3.2) , namely,

$$\min\{\theta(x)+f(x)\,|\,x\in\mathcal{X}\},$$

the equivalent variational inequality form is

$$x^*\in\mathcal{X},\quad \theta(x)-\theta(x^*)+(x-x^*)^T\nabla f(x^*)\ge 0,\quad \forall\, x\in\mathcal{X}. \tag{3.7a}$$

For solving the problem (3.2), the variational inequality form of the k-th iteration of the PPA (see (3.4)) is:

$$\begin{aligned} x^{k+1}\in\mathcal{X},\quad \theta(x)-\theta(x^{k+1})+(x-x^{k+1})^T\nabla f(x^{k+1}) &\\ \ge (x-x^{k+1})^T r(x^k-x^{k+1}),\quad \forall\, x\in\mathcal{X}. & \end{aligned} \tag{3.7b}$$

PPA 通过求解一系列的 (3.3), 求得 (3.2) 的解, 采用的是步步为营的策略.

According to (3.7), we consider the PPA for the variational inequality (2.12)

3.2 Preliminaries of PPA for Variational Inequalities

The optimal condition of the linearly constrained convex optimization is characterized as a mixed monotone variational inequality:

$$w^* \in \Omega, \quad \theta(u) - \theta(u^*) + (w - w^*)^T F(w^*) \geq 0, \quad \forall\, w \in \Omega. \tag{3.8}$$

PPA for VI (3.8) in Euclidean-norm For given w^k and $r > 0$, find w^{k+1},

$$\begin{aligned} w^{k+1} \in \Omega, \quad \theta(u) - \theta(u^{k+1}) + (w - w^{k+1})^T F(w^{k+1}) \\ \geq (w - w^{k+1})^T r(w^k - w^{k+1}), \quad \forall\, w \in \Omega. \end{aligned} \tag{3.9}$$

w^{k+1} is called the proximal point of the k-th iteration for the problem (3.8).

✠ **w^k is the solution of (3.8) if and only if $w^k = w^{k+1}$** ✠

Setting $w = w^*$ in (3.9), we obtain

$$(w^{k+1} - w^*)^T r(w^k - w^{k+1}) \geq \theta(u^{k+1}) - \theta(u^*) + (w^{k+1} - w^*)^T F(w^{k+1})$$

Note that (see the structure of $F(w)$ in (2.13))

$$(w^{k+1} - w^*)^T F(w^{k+1}) = (w^{k+1} - w^*)^T F(w^*),$$

and consequently (by using (3.8)) we obtain

$$(w^{k+1} - w^*)^T r(w^k - w^{k+1}) \ge \theta(u^{k+1}) - \theta(u^*) + (w^{k+1} - w^*)^T F(w^*) \ge 0.$$

Thus, we have

$$(w^{k+1} - w^*)^T (w^k - w^{k+1}) \ge 0. \tag{3.10}$$

By setting $\boldsymbol{a = w^k - w^*}$ and $\boldsymbol{b = w^{k+1} - w^*}$, the inequality (3.10) means that $\boldsymbol{b^T(a - b) \ge 0}$. By using Lemma 2, we obtain

$$\|\boldsymbol{w^{k+1} - w^*}\|^2 \le \|\boldsymbol{w^k - w^*}\|^2 - \|\boldsymbol{w^k - w^{k+1}}\|^2. \tag{3.11}$$

We get the nice convergence property of Proximal Point Algorithm.

PPA for monotone mixed VI in H-norm

For given w^k, find the proximal point w^{k+1} in H-norm which satisfies

$$w^{k+1} \in \Omega, \quad \theta(u) - \theta(u^{k+1}) + (w - w^{k+1})^T F(w^{k+1}) \\ \geq (w - w^{k+1})^T H(w^k - w^{k+1}), \ \forall\, w \in \Omega, \tag{3.12}$$

where H is a symmetric positive definite matrix.

✠ Again, w^k is the solution of (3.8) if and only if $w^k = w^{k+1}$ ✠

Convergence Property of Proximal Point Algorithm in H-norm

$$\|w^{k+1} - w^*\|_H^2 \leq \|w^k - w^*\|_H^2 - \|w^k - w^{k+1}\|_H^2. \tag{3.13}$$

The sequence $\{w^k\}$ is Fejér monotone in H-norm. In primal-dual algorithm [7], via choosing a proper positive definite matrix H, the solution of the subproblem (3.12) has a closed form. **In addition, for the residue sequence, we have**

$$\|w^k - w^{k+1}\|_H^2 \leq \|w^{k-1} - w^k\|_H^2 - \|(w^{k-1} - w^k) - (w^k - w^{k+1})\|_H^2.$$

4 From augmented Lagrangian method to C-PPA

We consider the convex optimization (1.1), namely

$$\min\{\theta(x) \mid Ax = b,\ x \in \mathcal{X}\}.$$

4.1 From C-P method to Customized PPA

The Lagrange function is

$$L(x,\lambda) = \theta(x) - \lambda^T(Ax-b), \qquad (x,\lambda) \in \mathcal{X} \times \Re^m.$$

For the primal-dual methods and customized PPA in this subsection, we assume that the subproblem

$$\min\{\theta(x) + \frac{r}{2}\|x-a\|^2 \mid x \in \mathcal{X}\} \quad \text{is simple}.$$

4.1.1 Original primal-dual hybrid gradient algorithm [20]

For given (x^k, λ^k), produce a pair of (x^{k+1}, λ^{k+1}). First,

$$x^{k+1} = \text{Argmin}\{L(x, \lambda^k) + \frac{r}{2}\|x - x^k\|^2 \mid x \in \mathcal{X}\}, \tag{4.1a}$$

and then we obtain λ^{k+1} via

$$\lambda^{k+1} = \text{Argmax}\{L(x^{k+1}, \lambda) - \frac{s}{2}\|\lambda - \lambda^k\|^2 \mid \lambda \in \Re^m\}. \tag{4.1b}$$

Note that the optimality condition of (4.1a) is

$$\theta(x) - \theta(x^{k+1}) + (x - x^{k+1})^T\{-A^T\lambda^k + r(x^{k+1} - x^k)\} \geq 0, \ \forall\, x \in \mathcal{X}. \tag{4.2}$$

The problem (4.1b) is an unconstrained optimization, thus we have

$$(Ax^{k+1} - b) + s(\lambda^{k+1} - \lambda^k) = 0, \tag{4.3}$$

and it can be written as

$$\lambda^{k+1} \in \Re^m, \ (\lambda - \lambda^{k+1})^T\{(Ax^{k+1} - b) + s(\lambda^{k+1} - \lambda^k)\} \geq 0, \ \forall \lambda \in \Re^m.$$

Combining (4.2) and (4.3), we get

$$\theta(x)-\theta(x^{k+1})+\begin{pmatrix} x-x^{k+1}\\ \lambda-\lambda^{k+1}\end{pmatrix}^T\left\{\begin{pmatrix} -A^T\lambda^{k+1}\\ Ax^{k+1}-b\end{pmatrix}\right.$$

$$\left.+\begin{pmatrix} r(x^{k+1}-x^k)+A^T(\lambda^{k+1}-\lambda^k)\\ s(\lambda^{k+1}-\lambda^k)\end{pmatrix}\right\}\geq 0,\quad \forall(x,\lambda)\in\Omega,$$

where

$$\Omega=\mathcal{X}\times\Re^m.$$

The compact form is

$$\theta(x)-\theta(x^{k+1})+(w-w^{k+1})^T\{F(w^{k+1})+Q(w^{k+1}-w^k)\}\geq 0,\ \forall w\in\Omega, \tag{4.4}$$

where

$$Q=\begin{pmatrix} rI_n & A^T\\ 0 & sI_m\end{pmatrix}\qquad \text{is not symmetric.}$$

It does not result in the PPA form (3.12), and we can not expect its convergence.

The following example of linear programming indicates the original PDHG (4.1) is not necessary convergent.

Consider the following pair of primal-dual linear programming:

$$\text{(Primal)} \quad \begin{array}{ll} \min & x \\ \text{s. t.} & x = 1 \\ & x \geq 0. \end{array} \qquad\qquad \text{(Dual)} \quad \begin{array}{ll} \max & y \\ \text{s. t.} & y \leq 1 \end{array}$$

The optimal solutions of this pair of linear programming are $x^* = 1$ and $y^* = 1$. Note that its Lagrange function is

$$L(x, y) = x - y(x - 1) \tag{4.5}$$

which defined on $R_+ \times R$. $(x^*, y^*) = (1, 1)$ is the unique saddle point of the Lagrange function.

For solving the $\min\text{-}\max$ problem (4.5), by using (4.1), the iterative formula is

$$\begin{cases} x^{k+1} = \max\{(x^k + \frac{1}{r}(y^k - 1)), 0\}, \\ y^{k+1} = y^k - \frac{1}{s}(x^{k+1} - 1). \end{cases}$$

We use $(x^0, y^0) = (0,0)$ as the start point. For this example, the method is not convergent.

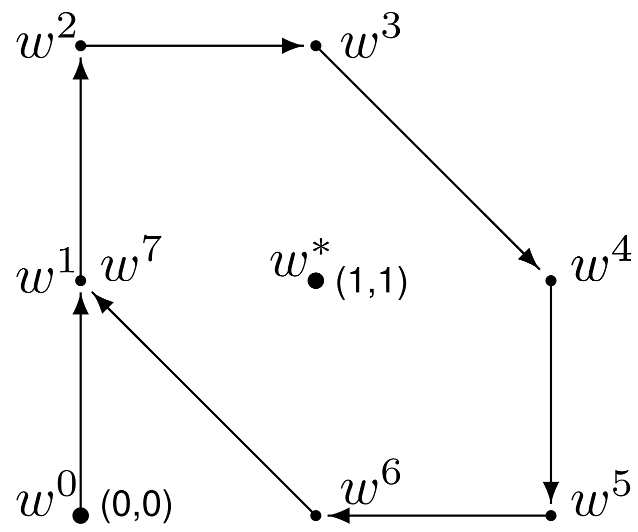


Fig. 4.1 The sequence generated by PDHG Method with $r = s = 1$

4.1.2 Customized Proximal Point Algorithm-Classical Version

If we change the non-symmetric matrix Q to a symmetric matrix H such that

$$Q = \begin{pmatrix} rI_n & A^T \\ 0 & sI_m \end{pmatrix} \quad \Rightarrow \quad H = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix},$$

then the variational inequality (4.4) will become the following desirable form:

$$\theta(x) - \theta(x^{k+1}) + (w - w^{k+1})^T \{F(w^{k+1}) + H(w^{k+1} - w^k)\} \geq 0, \ \forall w \in \Omega. \tag{4.6}$$

For this purpose, we need only to change (4.3), namely,

$$(Ax^{k+1} - b) + s(\lambda^{k+1} - \lambda^k) = 0,$$

to

$$(Ax^{k+1} - b) + A(x^{k+1} - x^k) + s(\lambda^{k+1} - \lambda^k) = 0. \tag{4.7}$$

Because x^{k+1} is known, with the given x^k and λ^k, λ^{k+1} in (4.7) is given by

$$\lambda^{k+1} = \lambda^k - \frac{1}{s}[A(2x^{k+1} - x^k) - b].$$

Thus, for given (x^k, λ^k), produce a proximal point (x^{k+1}, λ^{k+1}) via (4.1a) and (4.7) can be summarized as:

$$x^{k+1} = \text{argmin}\big\{L(x, \lambda^k) + \frac{r}{2}\|x - x^k\|^2 \,\big|\, x \in \mathcal{X}\big\}. \tag{4.8a}$$

$$\lambda^{k+1} = \text{argmax}\big\{L\big([2x^{k+1} - x^k], \lambda\big) - \frac{s}{2}\|\lambda - \lambda^k\|^2\big\} \tag{4.8b}$$

By ignoring the constant term in the objective function, getting x^{k+1} from (4.8a) is equivalent to obtaining x^{k+1} from

$$x^{k+1} = \text{argmin}\Big\{\theta(x) + \frac{r}{2}\big\|x - \big[x^k + \frac{1}{r}A^T\lambda^k\big]\big\|^2 \,\big|\, x \in \mathcal{X}\Big\}.$$

The solution of (4.8b) is given by

$$\lambda^{k+1} = \lambda^k - \frac{1}{s}[A(2x^{k+1} - x^k) - b].$$

Assumption: $\min\left\{\theta(x) + \frac{r}{2}\|x - a\|^2 \,|\, x \in \mathcal{X}\right\}$ **is simple**

Indeed, under the assumption, the sub-problem (4.8a) is simple.

In the case that $rs > \|A^T A\|$, the matrix

$$H = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix} \quad \text{is positive definite.}$$

Theorem 1 *The sequence* $\{w^k = (x^k, \lambda^k)\}$ *generated by the customized PPA* *satisfies*

$$\|w^{k+1} - w^*\|_H^2 \le \|w^k - w^*\|_H^2 - \|w^k - w^{k+1}\|_H^2. \tag{4.9}$$

For solving the $\min$-$\max$ problem (4.5), by using (4.8b), the iterative formula is

$$\begin{cases} x^{k+1} = \max\{(x^k + \frac{1}{r}(y^k - 1)), 0\}, \\ y^{k+1} = y^k - \frac{1}{s}[2(x^{k+1} - x^k) - 1]. \end{cases}$$

We use $(x^0, y^0) = (0, 0)$ as the start point. For this example, the method is convergent in 3 steps.

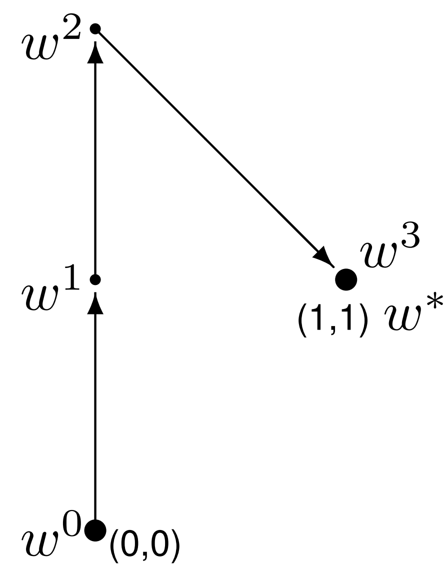


Fig. 4.2 The sequence generated by C-PPA Method with $r = s = 1$

The VI approach greatly simplifies the convergence analysis of the CP (Chambolle-Pock) method which can be viewed as a classical version of the customized PPA.

◇ A. Chambolle, T. Pock, A first-order primal-dual algorithms for convex problem with applications to imaging, J. Math. Imaging Vison, 40, 120-145, 2011.

◇ B.S. He, X.M. Yuan and W.X. Zhang, A customized proximal point algorithm for convex minimization with linear constraints, Comput. Optim. Appl., 56: 559-572, 2013.

◇ G.Y. Gu, B.S. He and X.M. Yuan, Customized proximal point algorithms for linearly constrained convex minimization and saddle-point problems: a unified approach, Comput. Optim. Appl., 59(2014), 135-161.

◇ B.S. He and X.M. Yuan, Convergence analysis of primal-dual algorithms for a saddle -point problem: From contraction perspective, SIAM J. Imag. Sci., 5, 119-149, 2012.

Produce (x^{k+1}, λ^{k+1}) by using the dual-primal order:

$$\lambda^{k+1} = \text{argmax}\Big\{L\big(x^k, \lambda\big) - \frac{s}{2}\big\|\lambda - \lambda^k\big\|^2\Big\} \tag{4.10a}$$

$$x^{k+1} = \text{argmin}\Big\{L(x, (2\lambda^{k+1} - \lambda^k)) + \frac{r}{2}\big\|x - x^k\big\|^2 \;\big|\; x \in \mathcal{X}\Big\}. \tag{4.10b}$$

By using the notation of w, $F(w)$ and Ω in (2.13), we get $w^{k+1} \in \Omega$ and

$$\theta(x)-\theta(x^{k+1})+(w-w^{k+1})^T\{F(w^{k+1})+H(w^{k+1}-w^k)\} \geq 0, \;\; \forall w \in \Omega,$$

where

$$H = \begin{pmatrix} rI_n & -A^T \\ -A & sI_m \end{pmatrix}$$

is symmetric and in the case $rs > \|A^TA\|$, the matrix H is positive definite.

Note that in the primal-dual order,

$$H = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix}.$$

Remark Let the linear constraints become to a system of inequalities.

$$\boxed{\min\{\theta(x) \mid Ax = b,\ x \in \mathcal{X}\}} \quad \Rightarrow \quad \boxed{\min\{\theta(x) \mid Ax \geq b,\ x \in \mathcal{X}\}}$$

In this case, the Lagrange multiplier λ should be nonnegative. $\Omega = \mathcal{X} \times \Re^m_+$.

We need only to make a slight change in the prediction procedure:

In the primal-dual order:

$$\boxed{\lambda^{k+1} = \lambda^k - \tfrac{1}{s}\big(A(2x^{k+1} - x^k) - b\big) \ \Rightarrow\ \lambda^{k+1} = \big[\lambda^k - \tfrac{1}{s}\big(A(2x^{k+1} - x^k\big) - b]\big]_+}$$

In the dual-primal order:

$$\boxed{\lambda^{k+1} = \lambda^k - \tfrac{1}{s}(Ax^k - b) \quad \Rightarrow \quad \lambda^{k+1} = \big[\lambda^k - \tfrac{1}{s}(Ax^k - b)\big]_+}$$

4.2 Simplicity recognition

Frame of VI is recognized by some Researcher in Image Science

Diagonal preconditioning for first order primal-dual algorithms in convex optimization*

Thomas Pock
Institute for Computer Graphics and Vision
Graz University of Technology
pock@icg.tugraz.at

Antonin Chambolle
CMAP & CNRS
École Polytechnique
antonin.chambolle@cmap.polytechnique.fr

- T. Pock and A. Chambolle, IEEE ICCV, 1762-1769, 2011
- A. Chambolle, T. Pock, A first-order primal-dual algorithms for convex problem with applications to imaging, J. Math. Imaging Vison, 40, 120-145, 2011.

preconditioned algorithm. In very recent work [10], it has been shown that the iterates (2) can be written in form of a proximal point algorithm [14], which greatly simplifies the convergence analysis.

From the optimality conditions of the iterates (4) and the convexity of G and F^* it follows that for any $(x, y) \in X \times Y$ the iterates x^{k+1} and y^{k+1} satisfy

$$\left\langle \begin{pmatrix} x - x^{k+1} \\ y - y^{k+1} \end{pmatrix}, F\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} + M\begin{pmatrix} x^{k+1} - x^{k} \\ y^{k+1} - y^{k} \end{pmatrix} \right\rangle \geq 0 \, , \tag{5}$$

where

$$F\begin{pmatrix} x^{k+1} \\ y^{k+1} \end{pmatrix} = \begin{pmatrix} \partial G(x^{k+1}) + K^T y^{k+1} \\ \partial F^*(y^{k+1}) - K x^{k+1} \end{pmatrix} ,$$

and

$$M = \begin{bmatrix} \mathrm{T}^{-1} & -K^T \\ -\theta K & \Sigma^{-1} \end{bmatrix} . \tag{6}$$

It is easy to check, that the variational inequality (5) now takes the form of a proximal point algorithm [10, 14, 16].

[9] L. Ford and D. Fulkerson. *Flows in Networks*. Princeton University Press, Princeton, New Jersey, 1962.

[10] B. He and X. Yuan. Convergence analysis of primal-dual algorithms for total variation image restoration. Technical report, Nanjing University, China, 2010.

Math. Program., Ser. A
DOI 10.1007/s10107-015-0957-3

CrossMark

FULL LENGTH PAPER

On the ergodic convergence rates of a first-order primal–dual algorithm

Antonin Chambolle[1] · **Thomas Pock**[2,3]

The paper published by Chambolle and Pock in Math. Progr. uses the VI framework

In this work we revisit a first-order primal–dual algorithm which was introduced in [15, 26] and its accelerated variants which were studied in [5]. We derive new estimates for the rate of convergence. In particular, exploiting a proximal-point interpretation due to [16], we are able to give a very elementary proof of an ergodic $O(1/N)$ rate of convergence (where N is the number of iterations), which also generalizes to non-

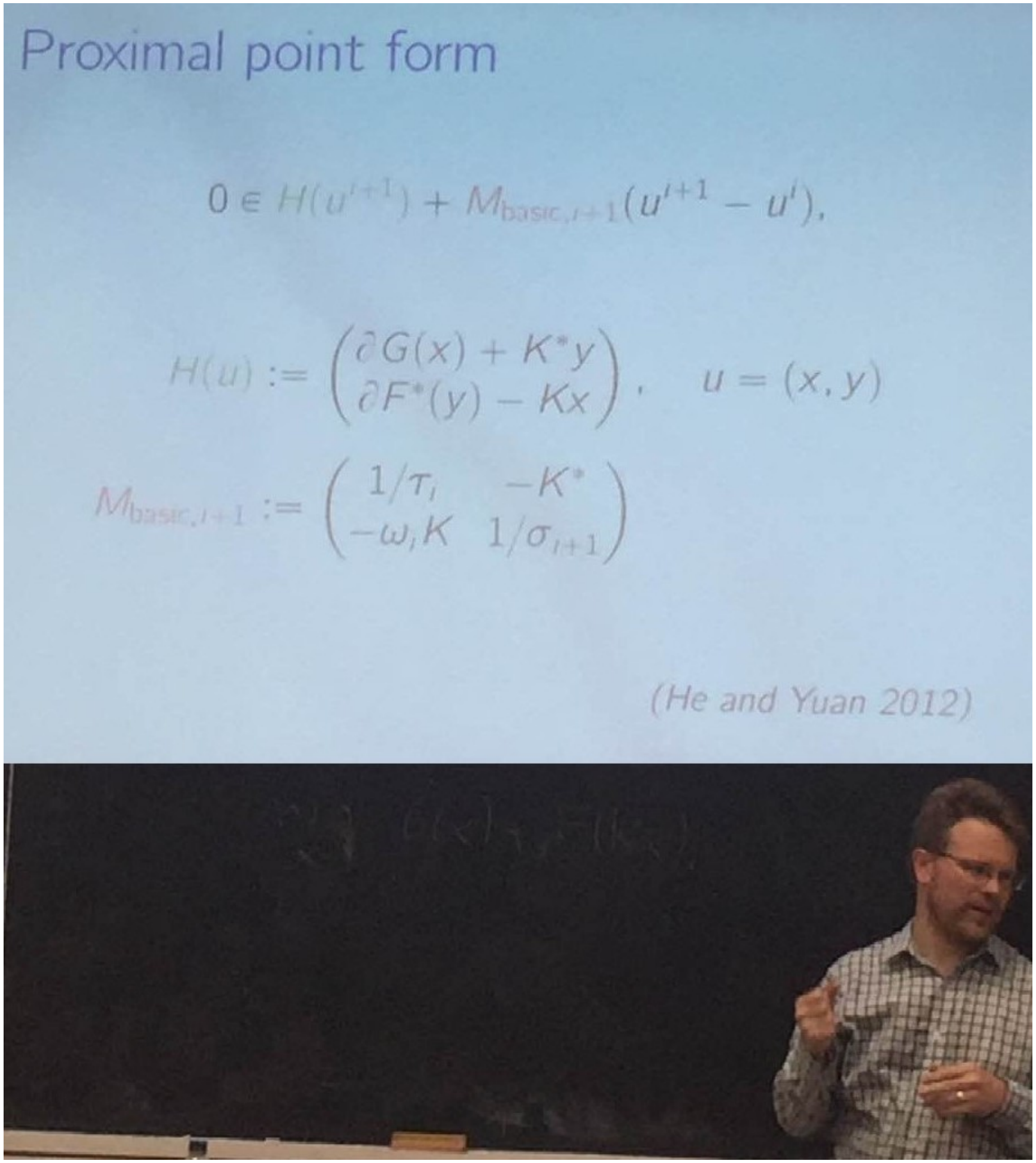
Algorithm 1: $O(1/N)$ Non-linear primal–dual algorithm

- Input: Operator norm $L := \|K\|$, Lipschitz constant L_f of ∇f, and Bregman distance functions D_x and D_y.
- Initialization: Choose $(x^0, y^0) \in \mathcal{X} \times \mathcal{Y}$, $\tau, \sigma > 0$
- Iterations: For each $n \geq 0$ let

$$(x^{n+1}, y^{n+1}) = \mathcal{PD}_{\tau,\sigma}(x^n, y^n, 2x^{n+1} - x^n, y^n) \tag{11}$$

The elegant interpretation in [16] shows that by writing the algorithm in this form

♣ 该文的文献 [16] 是我们发表在 SIAM J. Imaging Science 上的文章.
B.S. He and X.M. Yuan, Convergence analysis of primal-dual algorithms for a saddle -point problem: From contraction perspective, *SIAM J. Imag. Science* **5**(2012), 119-149.



2017 年 7 月, 南方科技大学数学系的一位副主任去英国访问. 在他参加的一个学术会议上, 首位报告人讲到, 用 He and Yuan 提出的邻近点形式 (PPF), 处理图像问题。

见到一幅幻灯片介绍我们的工作, 我的同事抢拍了一张照片发给我。

这也说明, 只有简单的思想才容易得到传播, 被人接受。

5 Applications in scientific computation

5.1 Finding the nearest correlation matrix

$$\min\{\frac{1}{2}\|X - C\|_F^2 \,|\, \text{diag}(X) = e,\ X \in S_+^n\}, \tag{5.1}$$

where e is a n-vector whose each element is equal 1.

The problem has the mathematical form (1.1) with $\|A^T A\| = 1$.

We use $z \in \Re^n$ as the Lagrange multiplier for the linear equality constraint.

For given (X^k, z^k), produce the predictor (X^{k+1}, z^{k+1}) by using (4.10):

1. Producing z^{k+1} by

$$z^{k+1} = z^k - \frac{1}{s}(\text{diag}(X^k) - e).$$

2. Finding X^{k+1} which is the solution of the following minimization problem

$$\min\{\frac{1}{2}\|X-C\|_F^2+\frac{r}{2}\|X-[X^k+\frac{1}{r}\text{diag}(2z^{k+1}-z^k)]\|_F^2|X\in S_+^n\}. \tag{5.2}$$

How to solve the subproblem (5.2) The problem (5.2) is equivalent to

$$\min\{\frac{1}{2}\|X-\frac{1}{1+r}[rX^k+\text{diag}(2z^{k+1}-z^k)+C]\|_F^2|X\in S_+^n\}.$$

The main computational load of each iteration is a SVD decomposition.

Numerical Tests To construct the test examples, we give the matrix C via:

C=rand(n,n); C=(C'+C)-ones(n,n) + eye(n)

In this way, C is symmetric, $C_{jj}\in(0,2)$, and $C_{ij}\in(-1,1)$, for $i\neq j$.

Matlab code for Creating the test examples

```
clear;  close all; n = 1000;     tol=1e-5;     r=2.0;   s=1.01/r;
gamma=1.5; rand('state',0);  C=rand(n,n);   C=(C'+C)-ones(n,n) +
eye(n);
```

Matlab code of the classical Customized PPA

```
%%%     Classical PPA for calibrating correlation matrix         %(1)
function PPAC(n,C,r,s,tol)                                       %(2)
X=eye(n);      y=zeros(n,1);    tic;      %%  The initial iterate %(3)
stopc=1;                         k=0;                            %(4)
while (stopc>tol && k<=100)          %% Beginning of an Iteration %(5)
 if mod(k,20)==0 fprintf(' k=%4d   epsm=%9.3e  \n',k,stopc); end; %(6)
    X0=X;    y0=y;    k=k+1;                                     %(7)
    yt=y0 - (diag(X0)-ones(n,1))/s;              EY=y0-yt;       %(8)
    A=(X0*r + C + diag(yt*2-y0))/(1+r);                          %(9)
    [V,D]=eig(A);   D=max(0,D);  XT=(V*D)*V';     EX=X0-XT;      %(10)
    ex=max(max(abs(EX)));  ey=max(abs(EY));   stopc=max(ex,ey);  %(11)
    X=XT;                      y=yt;                             %(12)
end;                                        % End of an Iteration %(13)
toc;                       TB = max(abs(diag(X-eye(n))));        %(14)
fprintf(' k=%4d  epsm=%9.3e   max|X_jj - 1|=%8.5f \n',k,stopc,TB); %%
```

The SVD decomposition is performed by [V,D]=eig(A) in the line (10) of the above code.

The computational load of each decomposition [V,D]=eig(A) is about $9n^3$ flops.

Modifying the Classical PPA to Extended PPA, it needs only change the line (12)

Matlab Code of the Extended Customized PPA

```
%%%     Extended PPA for calibrating correlation matrix           %(1)
function PPAE(n,C,r,s,tol,gamma)                                  %(2)
X=eye(n);      y=zeros(n,1);    tic;     %%  The initial iterate  %(3)
stopc=1;                         k=0;                             %(4)
while (stopc>tol && k<=100)          %% Beginning of an Iteration %(5)
 if mod(k,20)==0 fprintf(' k=%4d   epsm=%9.3e  \n',k,stopc); end; %(6)
    X0=X;    y0=y;    k=k+1;                                      %(7)
    yt=y0 - (diag(X0)-ones(n,1))/s;               EY=y0-yt;       %(8)
    A=(X0*r + C + diag(yt*2-y0))/(1+r);                           %(9)
    [V,D]=eig(A);   D=max(0,D);  XT=(V*D)*V';     EX=X0-XT;      %(10)
    ex=max(max(abs(EX)));  ey=max(abs(EY));   stopc=max(ex,ey);  %(11)
    X=X0-EX*gamma;         y=y0-EY*gamma;                        %(12)
end;                                        % End of an Iteration %(13)
toc;                   TB = max(abs(diag(X-eye(n))));            %(14)
fprintf(' k=%4d  epsm=%9.3e   max|X_jj - 1|=%8.5f \n',k,stopc,TB); %%
```

The difference of the above mentioned codes only in the line 12, the method is much efficient by taking the relaxed factor $\gamma = 1.5$.

Numerical results of (5.1)–SVD by using Mexeig

$n \times n$ Matrix	Classical PPA		Extended PPA	
$n =$	No. It	CPU Sec.	No. It	CPU Sec.
100	30	0.12	23	0.10
200	33	0.54	25	0.40
500	38	7.99	26	6.25
800	38	37.44	28	27.04
1000	45	94.32	30	55.32
2000	62	723.40	38	482.18

The extended PPA converges faster than the classical PPA.

$$\frac{\text{It. No. of Extended PPA}}{\text{It. No. of Classical PPA}} \approx 65\%.$$

5.2 Application in matrix completion problem

$$(\textbf{Problem}) \qquad \min\{\|X\|_* \mid X_\Omega = M_\Omega\}. \tag{5.3}$$

We let $Z \in \Re^{n\times n}$ as the Lagrangian multiplier to the constraints $X_\Omega = M_\Omega$.

For given (X^k, Z^k), applying (4.8) to produce (X^{k+1}, Z^{k+1}):

1. Producing Z^{k+1} by

$$z_\Omega^{k+1} = Z_\Omega^k - \frac{1}{s}(X_\Omega^k - M_\Omega). \tag{5.4}$$

2. Finding X^{k+1} by

$$X^{k+1} = \arg\min\ \Big\{\|X\|_* + \frac{r}{2}\Big\|X - \big[X^k + \frac{1}{r}(2\tilde{Z}_\Omega^k - Z_\Omega^k)\big]\Big\|_F^2\Big\}. \tag{5.5}$$

Then, the new iterate is given by

$$X^{k+1} := X^k - \gamma(X^k - X^{k+1}), \qquad Z^{k+1} := Z^k - \gamma(Z^k - z^{k+1}).$$

Test examples

The test examples is taken from

◇ J. F. Cai, E. J. Candès and Z. W. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. **20**, 1956-1982, 2010.

Code for Creating the test examples of Matrix Completion

```
%%  Creating the test examples of the matrix Completion problem     %(1)
clear all;  clc                                                      %(2)
maxIt=100;        tol = 1e-4;                                        %(3)
r=0.005;          s=1.01/r;        gamma=1.5;                        %(4)
  n=200;            ra = 10;         oversampling = 5;               %(5)
% n=1000;   ra=100;     oversampling = 3; %%  Iteration No. 31       %(6)
% n=1000;   ra=50;      oversampling = 4; %%  Iteration No. 36       %(7)
% n=1000;   ra=10;      oversampling = 6; %%  Iteration No. 78       %(8)
%% Generating the test problem                                       %(9)
rs = randseed;          randn('state',rs);                          %(10)
M=randn(n,ra)*randn(ra,n);           %% The matrix will be completed %(11)
df =ra*(n*2-ra);                     %% The freedom of the matrix   %(12)
mo=oversampling;                                                    %(13)
m =min(mo*df,round(.99*n*n));         %% No. of the known elements  %(14)
Omega= randsample(n^2,m);             %% Define the subset Omega    %(15)
fprintf('Matrix: n=%4d  Rank(M)=%3d  Oversampling=%2d \n',n,ra,mo);%(16)
```

Code: Extended Customized PPA for Matrix Completion Problem

```
function PPAE(n,r,s,M,Omega,maxIt,tol,gamma)    %  Ititial Process %%(1)
X=zeros(n);      Y=zeros(n);      YT=zeros(n);                      %(2)
  nM0=norm(M(Omega),’fro’);    eps=1;  VioKKT=1;  k=0;   tic;       %(3)
%% Minimum nuclear norm solution by PPA method                      %(4)
while  (eps > tol && k<= maxIt)                                     %(5)
 if mod(k,5)==0                                                     %(6)
 fprintf(’It=%3d |X-M|/|M|=%9.2e VioKKT=%9.2e\n’,k,eps,VioKKT); end;%(7)
  k=k+1;       X0=X;       Y0=Y;                                    %(8)
  YT(Omega)=Y0(Omega)-(X0(Omega)-M(Omega))/s;              EY=Y-YT;  %(9)
  A = X0 + (YT*2-Y0)/r;                         [U,D,V]=svd(A,0);   %(10)
  D=D-eye(n)/r;     D=max(D,0);      XT=(U*D)*V’;          EX=X-XT;  %(11)
  DXM=XT(Omega)-M(Omega);              eps = norm(DXM,’fro’)/nM0;   %(12)
  VioKKT = max( max(max(abs(EX)))*r, max(max(abs(EY))) );           %(13)
   if (eps <= tol)   gamma=1;   end;                                %(14)
  X = X0 - EX*gamma;                                                %(15)
  Y(Omega) = Y0(Omega) - EY(Omega)*gamma;                           %(16)
end;                                                                %(17)
 fprintf(’It=%3d  |X-M|/|M|=%9.2e  Vi0KKT=%9.2e \n’,k,eps,VioKKT); %(18)
   RelEr=norm((X-M),’fro’)/norm(M,’fro’);   toc;                    %(19)
 fprintf(’ Relative error = %9.2e  Rank(X)=%3d \n’,RelEr,rank(X)); %(20)
 fprintf(’ Violation of KKT Condition = %9.2e  \n’,VioKKT);         %(21)
```

Numerical Results: using SVD in Matlab

Unknown $n \times n$ matrix M				Computational Results		
n	rank(ra)	m/d_{ra}	m/n^2	#iters	times(Sec.)	relative error
1000	10	6	0.12	76	841.59	9.38E-5
1000	50	4	0.39	37	406.24	1.21E-4
1000	100	3	0.58	31	362.58	1.50E-4

Numerical Results: Using SVD in PROPACK

Unknown $n \times n$ matrix M				Computational Results		
n	rank(ra)	m/d_{ra}	m/n^2	#iters	times(Sec.)	relative error
1000	10	6	0.12	**76**	30.99	9.30E-5
1000	50	4	0.39	**36**	40.25	1.29E-4
1000	100	3	0.58	**30**	42.45	1.50E-4

♣ The paper by Cai *et. al* is the first publication in SIAM J. Opti. for matrix completion problem. For the same accuracy, the iteration numbers are listed in the last column of the above table (See the first 3 examples in Table 5.1 of [2], Page. 1974).

♣ The reader may find, for the two examples in in §2.4, the constrained matrix A is a projection matrix and thus $\|A^TA\| = 1$, thus we take $rs = 1.01$. However, we take $r = 2$ an $r = 1/200$ in §2.4.1 and §2.4.2, respectively. r is problems-dependent.

References

[1] X.J. Cai, G.Y. Gu, B.S. He and X.M. Yuan, A proximal point algorithms revisit on the alternating direction method of multipliers, Science China Mathematics, 56 (2013), 2179-2186.

[2] J. F. Cai, E. J. Candès and Z. W. Shen, A singular value thresholding algorithm for matrix completion, SIAM J. Optim. **20**, 1956-1982, 2010.

[3] A. Chambolle, T. Pock, A first-order primal-dual algorithms for convex problem with applications to imaging, J. Math. Imaging Vison, 40, 120-145, 2011.

[4] C. H. Chen, B. S. He, Y. Y. Ye and X. M. Yuan, *The direct extension of ADMM for multi-block convex minimization problems is not necessarily convergent*, to appear in *Mathematical Programming, Series A*.

[5] D. Gabay, Applications of the method of multipliers to variational inequalities, *Augmented Lagrange Methods: Applications to the Solution of Boundary-valued Problems*, edited by M. Fortin and R. Glowinski, North Holland, Amsterdam, The Netherlands, 1983, pp. 299–331.

[6] R. Glowinski, *Numerical Methods for Nonlinear Variational Problems*, Springer-Verlag, New York, Berlin, Heidelberg, Tokyo, 1984.

[7] G.Y. Gu, B.S. He and X.M. Yuan, Customized proximal point algorithms for linearly constrained convex minimization and saddle-point problems: a unified approach, Comput. Optim. Appl., 59(2014), 135-161.

[8] D. Hallac, Ch. Wong, S. Diamond, A. Sharang, R. Sosič, S. Boyd and J. Leskovec, SnapVX: A Network-Based Convex Optimization Solver, Journal of Machine Learning Research 18 (2017) 1-5.

[9] B. S. He, H. Liu, Z.R. Wang and X.M. Yuan, A strictly contractive Peaceman-Rachford splitting method for convex programming, *SIAM Journal on Optimization* **24**(2014), 1011-1040.

[10] B. S. He, M. Tao and X.M. Yuan, Alternating direction method with Gaussian back substitution for separable convex programming, *SIAM Journal on Optimization* **22**(2012), 313-340.

[11] B.S. He, M. Tao and X.M. Yuan, A splitting method for separable convex programming, *IMA J. Numerical Analysis* **31**(2015), 394-426.

[12] B.S. He, H. Yang, and S.L. Wang, Alternating directions method with self-adaptive penalty parameters for monotone variational inequalities, *JOTA* **23**(2000), 349–368.

[13] B. S. He and X. M. Yuan, On the $O(1/t)$ convergence rate of the alternating direction method, *SIAM J. Numerical Analysis* **50**(2012), 700-709.

[14] B.S. He and X.M. Yuan, Convergence analysis of primal-dual algorithms for a saddle-point problem: From contraction perspective, *SIAM J. Imag. Science* **5**(2012), 119-149.

[15] M. R. Hestenes, Multiplier and gradient methods, *JOTA* **4**, 303-320, 1969.

[16] B. Martinet, Regularisation, d'inéquations variationelles par approximations succesives, *Rev. Francaise d'Inform. Recherche Oper.*, **4**, 154-159, 1970.

[17] J. Nocedal and S. J. Wright, Numerical Optimization, Springer Verlag, New York, 1999.

[18] M. J. D. Powell, A method for nonlinear constraints in minimization problems, in Optimization, R. Fletcher, ed., Academic Press, New York, NY, pp. 283-298, 1969.

[19] R.T. Rockafellar, Monotone operators and the proximal point algorithm, *SIAM J. Cont. Optim.*, **14**, 877-898, 1976.

[20] M. Zhu and T. F. Chan, An Efficient Primal-Dual Hybrid Gradient Algorithm for Total Variation Image Restoration, CAM Report 08-34, UCLA, Los Angeles, CA, 2008.

Thank you very much for your attention ！

Thank you very much for reading ！