一些典型凸优化问题的分裂收缩算法

基于变分不等式和邻近点算法的统一框架

南师大数科院系列讲座 2022 年元月 4 – 9 日

何 炳 生

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

五. 凸优化分裂收缩算法的统一框架

在预备知识一章已经把线性约束的凸优化问题和单调变分不等式

$$w^* \in \Omega, \quad \theta(u) - \theta(u^*) + (w - w^*)^T F(w^*) \geq 0, \quad \forall\, w \in \Omega \tag{0.1}$$

建立了对应关系. 凸优化分裂收缩算法的统一框架聚焦于求解上述变分不等式. 建立这个框架, 是受我们求解经典的单调变分不等式的投影收缩算法框架[6, 7, 11] 的影响与启发.

1 分裂收缩算法的统一框架

在统一框架中, 我们把算法的每步迭代理解成(有时是故意分拆成)预测和校正两部. 我们只需描述第 k-步迭代的预测和校正. 迭代方法中把 v 称为核心变量(essential variables), 是指 k-次迭代可以从给定的 v^k 开始. v 可以是 w 本身, 也可以是 w 的部分分量. 我们把 w 中除了 v 以外的分量(如果有)称为中间变量(intermediate variables).

统一框架中的预测

[预测] 统一框架算法的第k-步迭代从给定的核心变量v^k 开始, 通过求解一些子问题, 产生预测点$\tilde{w}^k$, 使得

$$\tilde{w}^k \in \Omega, \;\; \theta(u) - \theta(\tilde{u}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k) \geq (v - \tilde{v}^k)^T Q(v^k - \tilde{v}^k), \;\; \forall\, w \in \Omega, \tag{1.2}$$

成立. 其中矩阵 $Q^T + Q$ 是原则上(in principal)正定的.

这里需要解释一下预测公式中什么叫原则上正定. 在为求解与线性约束的凸优化等价的变分不等式设计的分裂算法中, 矩阵 $\mathcal{A}$ 常常需要被分块成 $\mathcal{A} = (A, B)$ 或者 $\mathcal{A} = (A, B, C)$. 如果当这些分块矩阵列满秩, 矩阵 $Q^T + Q$ 就正定, 这时我们就说 $Q^T + Q$ 是原则上正定的. 当那些列满秩条件不满足时, 原则上正定的 $Q^T + Q$ 只是半正定. 在往后的叙述中, 为了方便, 我们就把原则上正定就说成正定.

统一框架中的校正

[固定步长的校正公式]. 根据预测得到的$\tilde{v}^k$, 核心变量v的新的迭代点v^{k+1}由

$$v^{k+1} = v^k - \alpha M(v^k - \tilde{v}^k) \tag{1.3}$$

给出. 其中M是非奇异矩阵, $\alpha > 0$是确定的常数.

[计算步长的校正公式] 根据预测得到的$\tilde{v}^k$, 核心变量v的新迭代点v^{k+1}由

$$v^{k+1} = v^k - \alpha_k M(v^k - \tilde{v}^k), \quad \alpha_k = \gamma\alpha_k^*, \quad \gamma \in (0, 2) \tag{1.4}$$

给出, 其中α_k^* 和矩阵 M 要满足下面的收敛性条件.

收敛性条件(分别对固定步长和计算步长)

采用固定步长校正的收敛性条件 对预测公式(1.2) 中的矩阵Q 和校正公式(1.3) 中的矩阵M, 有正定矩阵

$$H \succ 0 \qquad \text{使得} \qquad HM = Q. \tag{1.5a}$$

此外, 校正公式(1.3) 中所取步长 $\alpha > 0$ 能够保证

$$G = Q^T + Q - \alpha M^T HM \succ 0. \tag{1.5b}$$

定义 1.1 当预测 (1.2) 中的矩阵Q本身就是一个对称正定矩阵时, 我们直接把它记为H. 在 (1.3) 中取$M = I$, $\alpha \in (0, 2)$,

$$v^{k+1} = v^k - \alpha(v^k - \tilde{v}^k), \qquad \alpha \in (0, 2). \tag{1.6}$$

这样的校正我们称之为**平凡的松弛校正**. 当矩阵Q非对称时, 校正公式(1.3)中的矩阵$M \neq I$, 这样的矩阵都称之为**必要的非平凡校正**.

在平凡的松弛校正中, $H = Q$和$G = (2 - \alpha)H$ 都是正定矩阵. 这时, 如果在(1.6) 式中取$\alpha = 1$, 就是H-模下的邻近点(PPA) 算法. 拿ALM的预测 (2.6) 来说,

$$Q = \frac{1}{\beta}I_m, \qquad H = Q, \qquad M = I_m, \qquad G = \frac{2-\alpha}{\beta}I_m.$$

然而, 计算实践发现, 采用$\alpha \in [1.2, 1.8]$ 的超松弛校正, 方法的效率一般都有30-50% 的提高. 通常, 我们把预测中的矩阵Q设计成对称正定矩阵的方法称为*按需定制的邻近点算法* (Customized PPA) [4].

对确定的Q, H和M, 由

$$\alpha_{\max} := \operatorname{argmax}\{\alpha | Q^T + Q - \alpha M^T HM \succeq 0\}. \tag{1.7}$$

定义的$\alpha_{\max}$是一个确定的大于零的常数. 对满足条件的(1.3)的α, $(\alpha_{\max} - \alpha)$同样是一个确定的大于零的常数. 对符合条件(1.5b)的矩阵G, 有

$$G \succeq \big(\alpha_{\max} - \alpha\big) M^T HM. \tag{1.8}$$

这个性质我们在证明点列意义下的收敛速率中需要用到.

2 统一框架算法框架下的一些算法

这本书后面篇章, 大部分是用统一框架指导改造或者设计算法. 这里, 我们先用一些简单的例子, 说明如何用统一框架来解释和设计算法.

2.1 平凡的松弛校正

使用平凡的松弛校正, 要求预测 (1.2) 中的矩阵 Q 本身是一个对称正定矩阵. 我们先以单块等式约束凸优化问题

$$\min\{\theta(x)\,|\,Ax=b,\,x\in\mathcal{X}\} \tag{2.1}$$

为例解释这样的预测. 问题 (2.1) 对应的拉格朗日函数是定义在 $\mathcal{X}\times\Re^m$ 上的

$$L(x,\lambda)=\theta(x)-\lambda^T(Ax-b).$$

相应的变分不等式是

$$w^*\in\Omega,\quad \theta(x)-\theta(x^*)+(w-w^*)^TF(w^*)\ge 0,\quad \forall\, w\in\Omega, \tag{2.2a}$$

其中

$$w=\begin{pmatrix} x\\ \lambda\end{pmatrix},\quad F(w)=\begin{pmatrix} -A^T\lambda\\ Ax-b\end{pmatrix},\quad \Omega=\mathcal{X}\times\Re^m. \tag{2.2b}$$

这相当于 (0.1) 中 $u=x$. 求解等式约束的问题一个被广泛接受的有效算法是增广拉格朗日乘子法. 对这个等式约束的问题, 增广拉格朗日函数是

$$\mathcal{L}_\beta(x,\lambda)=\theta(x)-\lambda^T(Ax-b)+\frac{\beta}{2}\|Ax-b\|^2.$$

增广拉格朗日乘子法的 k-步迭代从给定的 λ^k 开始, 通过

$$\text{(ALM)}\quad \left\{\begin{array}{ll} x^{k+1}=\arg\min\big\{\mathcal{L}_\beta(x,\lambda^k)\;\big|\;x\in\mathcal{X}\big\}, & (2.3a)\\ \lambda^{k+1}=\lambda^k-\beta(Ax^{k+1}-b), & (2.3b)\end{array}\right.$$

求得 $w^{k+1}=(x^{k+1},\lambda^{k+1})$. 这样的 $w^{k+1}\in\Omega$ 满足

$$\theta(x)-\theta(x^{k+1})+(w-w^{k+1})^TF(w^{k+1})\ge(\lambda-\lambda^{k+1})^T\frac{1}{\beta}(\lambda^k-\lambda^{k+1}),\ \ \forall\, w\in\Omega. \tag{2.4}$$

并能得到

$$\|\lambda^{k+1}-\lambda^*\|^2\le\|\lambda^k-\lambda^*\|^2-\|\lambda^k-\lambda^{k+1}\|^2.$$

这是增广拉格朗日乘子法收敛的关键不等式. 如果把 (2.3) 的输出作为预测点, 就是说

$$\left\{\begin{array}{rcl} \tilde{x}^k &=& \arg\min\big\{\mathcal{L}_\beta(x,\lambda^k)\;\big|\;x\in\mathcal{X}\big\},\\ \tilde{\lambda}^k &=& \lambda^k-\beta(A\tilde{x}^k-b),\end{array}\right. \tag{2.5}$$

相应的 (2.4) 就会变成 $\tilde{w}^k \in \Omega$ 并且

$$\theta(x) - \theta(\tilde{x}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k) \geq (\lambda - \tilde{\lambda}^k)^T \frac{1}{\beta}(\lambda^k - \tilde{\lambda}^k), \ \ \forall\, w \in \Omega. \tag{2.6}$$

上式相当于求解 (2.2) 的形如 (1.2) 的预测公式中取 $v = \lambda$ 和 $Q = \dfrac{1}{\beta} I_m$.

再看讨论的鞍点问题

$$\min_x \max_y \{\Phi(x, y) = \theta_1(x) - y^T A x - \theta_2(y) \,|\, x \in \mathcal{X}, y \in \mathcal{Y}\}.$$

这可以表述成紧凑的变分不等式形式:

$$u \in \Omega, \quad \theta(u) - \theta(u^*) + (u - u^*)^T F(u^*) \geq 0, \ \forall\, u \in \Omega, \tag{2.7a}$$

其中

$$u = \begin{pmatrix} x \\ y \end{pmatrix}, \quad \theta(u) = \theta_1(x) + \theta_2(y), \quad F(u) = \begin{pmatrix} -A^T y \\ Ax \end{pmatrix}, \quad \Omega = \mathcal{X} \times \mathcal{Y}. \tag{2.7b}$$

平凡松弛校正 如果把 02 讲中的 PPA 算法的输出当做预测点, 则有

$$\begin{cases} \tilde{x}^k = \text{argmin}\big\{\Phi(x, y^k) + \dfrac{r}{2}\|x - x^k\|^2 \,|\, x \in \mathcal{X}\big\}. & (2.8\text{a}) \\ \tilde{y}^k = \text{argmax}\big\{\Phi\big([2\tilde{x}^k - x^k], y\big) - \dfrac{s}{2}\|y - y^k\|^2\big\} & (2.8\text{b}) \end{cases}$$

预测点满足的变分不等式就是

$$\tilde{u}^k \in \Omega, \quad \theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T \{F(\tilde{u}^k) + Q(\tilde{u}^k - u^k)\} \geq 0, \ \forall u \in \Omega, \tag{2.9}$$

其中

$$Q = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix}. \tag{2.10}$$

这时 Q 是对称矩阵, 当 $rs > \|A^T A$ 时, 矩阵 Q 正定. 我们就用平凡的松弛校正

$$u^{k+1} = u^k - \alpha(u^k - \tilde{u}^k), \quad \alpha \in (0, 2)$$

产生新的迭代点. 对第三章提到的应用问题, 采用取 $\alpha \in [1.2, 1.8]$ 的松弛校正, 计算效率都有不同程度的提高.

2.2 必要的非平凡校正

当预测 (1.2) 中的矩阵 Q 不是对称矩阵的时候, 必须采用采用必要的非平凡校正. 这时, 无论在 (1.3) 和 (1.4) 中, 矩阵 M 都不是单位矩阵, 都需要一个正定矩阵 H (见 (1.5a), 使得 $HM = Q$. 在必要的非平凡校正中, 又有采用固定步长 和计算步长 的两种方法产生新的核心变量.

固定步长的非平凡校正

如果把 02 讲 PDHG 的输出当做预测点, 则有

$$\begin{cases} \tilde{x}^k = \mathrm{argmin}\{\theta_1(x) - x^T A^T y^k + \frac{r}{2}\|x - x^k\|^2 \mid x \in \mathcal{X}\}, & \text{(2.11a)} \\ \tilde{y}^k = \mathrm{argmin}\{\theta_2(y) + y^T A \tilde{x}^k + \frac{s}{2}\|y - y^k\|^2 \mid y \in \mathcal{Y}\}. & \text{(2.11b)} \end{cases}$$

根据 02 讲 的分析, 预测点满足的变分不等式就是

$$\tilde{u}^k \in \Omega, \quad \theta(u) - \theta(\tilde{u}^k) + (u - \tilde{u}^k)^T \{F(\tilde{u}^k) + Q(\tilde{u}^k - u^k)\} \geq 0, \ \forall u \in \Omega, \tag{2.12}$$

其中

$$Q = \begin{pmatrix} rI_n & A^T \\ 0 & sI_m \end{pmatrix}. \tag{2.13}$$

对由 (2.12) 得到的 $\tilde{u}^k$, 我们采用

$$u^{k+1} = u^k - M(u^k - \tilde{u}^k), \tag{2.14}$$

得到新的迭代点(也称校正点). 符合收敛条件 (1.5) 的矩阵 M 有多种选择, 下面我们举一些例子.

1. 第一种选择是取 M 为下三角块状矩阵

$$M = \begin{pmatrix} I_n & 0 \\ -\frac{1}{s}A & I_m \end{pmatrix}. \tag{2.15}$$

注意到这个矩阵的逆矩阵是

$$M = \begin{pmatrix} I_n & 0 \\ \frac{1}{s}A & I_m \end{pmatrix}.$$

对 (2.13) 中的矩阵 Q 和 (2.15) 中的矩阵 M, 通过简单计算就有

$$H = QM^{-1} = \begin{pmatrix} rI_n & A^T \\ 0 & sI_m \end{pmatrix} \begin{pmatrix} I_n & 0 \\ \frac{1}{s}A & I_m \end{pmatrix} = \begin{pmatrix} rI_n + \frac{1}{s}A^T A & A^T \\ A & sI_m \end{pmatrix}.$$

对任何 $r, s > 0$, H 是正定矩阵. 此外

$$\begin{aligned} G &= Q^T + Q - M^T H M = Q^T + Q - Q^T M \\ &= \begin{pmatrix} 2rI_n & A^T \\ A & 2sI_m \end{pmatrix} - \begin{pmatrix} rI_n & 0 \\ 0 & sI_m \end{pmatrix} = \begin{pmatrix} rI_n & A^T \\ A & sI_m \end{pmatrix}. \end{aligned}$$

当且仅当 $rs > \|A^T A\|$ 时矩阵 G 正定. 满足收敛性条件 (1.5).

2. 第二种选择是取 M 为上三角块状矩阵

$$M = \begin{pmatrix} I_n & \frac{1}{r}A^T \\ 0 & I_m \end{pmatrix}. \tag{2.16}$$

它的逆矩阵是

$$M = \begin{pmatrix} I_n & -\frac{1}{r}A^T \\ 0 & I_m \end{pmatrix}.$$

对 (2.13) 中的矩阵 Q 和 (2.16) 中的矩阵 M, 通过简单计算

$$H = QM^{-1} = \begin{pmatrix} rI_n & A^T \\ 0 & sI_m \end{pmatrix} \begin{pmatrix} I_n & -\frac{1}{r}A^T \\ 0 & I_m \end{pmatrix} = \begin{pmatrix} rI_n & 0 \\ 0 & sI_m \end{pmatrix}.$$

对任何 $r, s > 0$, H 是正定矩阵. 此外,

$$G = Q^T + Q - M^T HM = Q^T + Q - Q^T M = \begin{pmatrix} rI_n & 0 \\ 0 & sI_m - \frac{1}{r}AA^T \end{pmatrix}.$$

同样, 当且仅当 $rs > \|A^T A\|$ 时矩阵 G 正定. 满足收敛性条件 (1.5).

需要指出的是, 我们只需验证收敛性条件 (1.5) 是否满足, 并不要求算出矩阵 H 和 G.

当预测 (1.2) 中的矩阵 Q 并不对称正定的时候, 必须做*必要的非平凡校正*. 我们建议采用计算步长的方法的方式确定新的迭代点, 效率一般都有相当程度的提高. 从数值计算的花费看, 计算步长额外工作量是求

$$(v^k - \tilde{v}^k) \text{ 和 } Q(v^k - \tilde{v}^k) \qquad \text{以及} \qquad M(v^k - \tilde{v}^k) \text{ 和 } Q(v^k - \tilde{v}^k)$$

的内积. 这与求得预测点的工作量比起来, 往往是微不足道的.

3 统一框架中的算法收敛性质

统一框架采用相同的预测(1.2), 分别采用*固定步长*和*计算步长*的校正产生新的核心变量 v^{k+1}, 我们对采用不同校正的方法讨论其收敛性质. 证明的关键技术, 在[10, 12]中已经多次提到. 我们给出完整简要的证明.

3.1 采用固定步长校正的算法收敛性

首先对统一框架中采用 (1.2) 预测和 (1.3) 校正, 在条件 (1.5) 成立的情况下证明有关收敛性质.

定理 3.1 *用统一框架的算法求解变分不等式 (0.1), 分别以 (1.2) 预测和 (1.3) 校正生成迭代序列* $\{\tilde{w}^k\}$, $\{v^k\}$. *如果条件 (1.5) 成立, 那么有*

$$\begin{aligned} \tilde{w}^k \in \Omega, \quad & \alpha\big\{\theta(u) - \theta(\tilde{u}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k)\big\} \\ & \geq \frac{1}{2}\big(\|v - v^{k+1}\|_H^2 - \|v - v^k\|_H^2\big) + \frac{\alpha}{2}\|v^k - \tilde{v}^k\|_G^2, \quad \forall w \in \Omega. \end{aligned} \tag{3.1}$$

证明. 首先, 对预测点满足的变分不等式 (1.2) 两边乘上大于零的 α, 并利用 $Q = HM$, 有

$$\alpha\{\theta(u) - \theta(\tilde{u}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k)\} \geq (v - \tilde{v}^k)^T \alpha HM(v^k - \tilde{v}^k), \quad \forall w \in \Omega. \tag{3.2}$$

利用校正公式 (3.2), 我们有

$$\alpha M(v^k - \tilde{v}^k) = v^k - v^{k+1}.$$

代入预测公式(1.2)的右端就得到

$$\alpha\{\theta(u) - \theta(\tilde{u}^k) + (w - \tilde{w}^k)^T F(\tilde{w}^k)\} \geq (v - \tilde{v}^k)^T H(v^k - v^{k+1}), \quad \forall w \in \Omega. \tag{3.3}$$

对上式的右端利用恒等式

$$(a-b)^T H(c-d) = \frac{1}{2}\{\|a-d\|_H^2 - \|a-c\|_H^2\} + \frac{1}{2}\{\|b-c\|_H^2 - \|b-d\|_H^2\}, \tag{3.4}$$

并令其中的 $a = v$, $b = \tilde{v}^k$, $c = v^k$ 和 $d = v^{k+1}$, 就有

$$\begin{aligned} &(v - \tilde{v}^k)^T H(v^k - v^{k+1}) \\ =\ & \frac{1}{2}\big(\|v - v^{k+1}\|_H^2 - \|v - v^k\|_H^2\big) + \frac{1}{2}\big(\|\tilde{v}^k - v^k\|_H^2 - \|\tilde{v}^k - v^{k+1}\|_H^2\big). \end{aligned} \tag{3.5}$$

对 (3.5) 式右端后一个圆括号中的部分利用校正公式(1.3), 就得到

$$\begin{aligned} &\|v^k - \tilde{v}^k\|_H^2 - \|v^{k+1} - \tilde{v}^k\|_H^2 \\ \stackrel{(1.3)}{=}\ & \|v^k - \tilde{v}^k\|_H^2 - \|(v^k - \tilde{v}^k) - \alpha M(v^k - \tilde{v}^k)\|_H^2 \\ =\ & 2\alpha(v^k - \tilde{v}^k)^T HM(v^k - \tilde{v}^k) - \alpha^2\|M(v^k - \tilde{v}^k)\|_H^2 \\ \stackrel{(1.5a)}{=}\ & \alpha(v^k - \tilde{v}^k)^T(Q^T + Q - \alpha M^T HM)(v^k - \tilde{v}^k) \\ \stackrel{(1.5b)}{=}\ & \alpha\|v^k - \tilde{v}^k\|_G^2. \end{aligned} \tag{3.6}$$

将 (3.5)和(3.6) 代入 (3.3), 就得到定理的结论. □

定理 3.1 的证明中用到的恒等式 (3.4) 虽然很初等, 却非常关键. 这个恒等式也被A. Beck 在他的专著[1]中借用(见pp.428-429)并予以特别标注. 由于 v^* 表示 w^* 的核心变量部分, 我们记 $\mathcal{V}^* = \{v^* | w^* \in \Omega^*\}$. 利用定理 3.1, 可以马上得到序列 $\{\|v^k - v^*\|_H^2\}$ 的收缩性质.

定理 3.2 用统一框架的算法求解变分不等式 (0.1), 分别以 (1.2) 预测和 (1.3) 校正生成迭代序列 $\{\tilde{w}^k\}$, $\{v^k\}$. 如果条件 (1.5) 成立, 那么有

$$\|v^{k+1} - v^*\|_H^2 \leq \|v^k - v^*\|_H^2 - \alpha\|v^k - \tilde{v}^k\|_G^2, \quad \forall v^* \in \mathcal{V}^*. \tag{3.7}$$

证明. 将 (3.1) 中任意的 $w \in \Omega$ 用任意固定的解点 w^* 代入, 我们得到

$$\begin{aligned} &\|v^k - v^*\|_H^2 - \|v^{k+1} - v^*\|_H^2 \\ \geq\ & \alpha\|v^k - \tilde{v}^k\|_G^2 + 2\alpha\{\theta(\tilde{u}^k) - \theta(u^*) + (\tilde{w}^k - w^*)^T F(\tilde{w}^k)\}. \end{aligned} \tag{3.8}$$

利用算子 F 的 $(\tilde{w}^k - w^*)^T F(\tilde{w}^k) = (\tilde{w}^k - w^*)^T F(w^*)$ 性质和 w^* 是最优点, 就有

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

因此从(3.8) 得到

$$\|v^k - v^*\|_H^2 - \|v^{k+1} - v^*\|_H^2 \ge \alpha \|v^k - \tilde{v}^k\|_G^2.$$

定理得证. □

定理3.2 的结论是统一框架算法收敛的关键. 收缩不等式 (3.7) 和 PPA 算法的收缩不等式有类似的形式, 因此这个统一框架提供的算法也可以看作PPA 算法的延伸与发展. 利用定理 3.1 还可以得到算法收敛速率方面的性质, 有兴趣的读者可以在作者主页的[12] 中查到相关的论述.

由于 $\tilde{w}^k$ 是变分不等式(0.1) 解的充分必要条件是$v^k = \tilde{v}^k$, 定理 3.2 的结论就保证了统一框架的算法收敛. 注意到两种方法采用同样的预测(有同样的Q), 校正中也用同样的M(因而有同样的H), 不同的只是取固定步长和计算步长. 后面, 我们每提出和介绍一个方法, 只要验证固定步长算法的收敛条件 (1.5).

在定理 3.1 中, 我们证明了关系式 (3.1). 保留式中任意的 $w \in \Omega$, 主要是为了收敛速率证明的需要. 仅仅为了证明定理 3.2 中的 (3.7), 我们可以直接从 (3.3) 开始, 以 $w = w^*$ 代入, 就得到

$$(v^k - v^{k+1})^T H(\tilde{v}^k - v^*) \ge \alpha\{\theta(\tilde{u}^k)\theta(u^*) + (\tilde{w}^k - w^*)^T F(\tilde{w}^k)\}.$$

由上式右端非负, 我们就有

$$(v^k - v^{k+1})^T H(\tilde{v}^k - v^*) \ge 0 \quad \forall v^* \in \mathcal{V}^*. \tag{3.9}$$

在恒等式

$$(a-b)^T H(c-d) = \frac{1}{2}\{\|a-d\|_H^2 - \|b-d\|_H^2\} - \frac{1}{2}\{\|a-c\|_H^2 - \|b-c\|_H^2\},$$

中令 $a = v^k$, $b = v^{k+1}$, $c = \tilde{v}^k$ 和 $d = v^*$, 就有

$$\begin{aligned}&(v^k - v^{k+1})^T H(\tilde{v}^k - v^*)\\ &\quad= \frac{1}{2}\big(\|v^k - v^*\|_H^2 - \|v^{k+1} - v^*\|_H^2\big) - \frac{1}{2}\big(\|v^k - \tilde{v}^k\|_H^2 - \|v^{k+1} - \tilde{v}^k\|_H^2\big).\end{aligned}$$

再根据 (3.9), 有

$$\|v^k - v^*\|_H^2 - \|v^{k+1} - v^*\|_H^2 \ge \|v^k - \tilde{v}^k\|_H^2 - \|v^{k+1} - \tilde{v}^k\|_H^2. \tag{3.10}$$

这里的 v^{k+1} 由 (1.3) 或者 (1.4) 给出.

当 v^{k+1} 由固定步长的 (1.3) 给出的时候, 根据 (3.6),

$$\|v^k - \tilde{v}^k\|_H^2 - \|v^{k+1} - \tilde{v}^k\|_H^2 = \alpha \|v^k - \tilde{v}^k\|_G^2.$$

将其代入 (3.10), 就得到定理 3.2 的结论.

参考文献

[1] A. Beck, First-Order Methods in convex optimization, MOS-SIAM Series on Optimization.

[2] 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.

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

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

[5] B. S. He, PPA-like contraction methods for convex optimization: a framework using variational inequality approach. J. Oper. Res. Soc. China **3** (2015) 391 – 420.

[6] B.S. He, L.Z. Liao, and X. Wang, Proximal-like contraction methods for monotone variational inequalities in a unified framework I: Effective quadruplet and primary methods, Comput. Optim. Appl., 51, 649-679, 2012

[7] B.S. He, L.Z. Liao, and X. Wang, Proximal-like contraction methods for monotone variational inequalities in a unified framework II: General methods and numerical experiments, Comput. Optim. Appl. 51, 681-708, 2012

[8] 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.

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

[10] B.S. He, My 20 years research on alternating directions method of multipliers (in Chinese). Oper. Res. Trans., 2018, 22: 1-31.

[11] 何炳生. 从变分不等式的统一收缩算法到凸优化的分裂收缩算法, 高等学校计算数学学报, 2016, 38: 74–96.

[12] 何炳生. 凸优化的一阶分裂算法–变分不等式为工具的统一框架, 见 http://maths.nju.edu. cn/~hebma 中的《My Talk》.

[13] 何炳生. 凸优化和单调变分不等式的收缩算法, 见 http://maths.nju.edu.cn/~hebma 中的系列讲义.