第60卷第6期
2020年11月

大连理工大学学报
Journal of Dalian University of Technology

Vol. 60, No. 6
Nov. 2020

文章编号: 1000-8608(2020)06-0642-05

求解均衡问题的分离惯性算法

高　辉[1,2]， 张明堃[1]， 王晓亮[1]， 庞丽萍[*1]

(1. 大连理工大学 数学科学学院, 辽宁 大连　116024;
2. 大连海洋大学 信息工程学院, 辽宁 大连　116023)

摘要: 研究两个函数和的非光滑均衡问题,对这类问题提出了一个结合惯性方法的分离算法. 每次迭代,交替求解两个简单的强凸子问题. 在不要求函数是 Lipschitz 连续或 Hölder 连续的条件下,证明了算法的收敛性. 通过与已有的几个算法比较,验证了算法的有效性.

关键词: 两个函数和;均衡问题;分离算法;惯性

中图分类号: O221.1　　**文献标识码:** A　　**doi:** 10.7511/dllgxb202006012

0　引　言

假设 $\boldsymbol{C}$ 是实希尔伯特空间　　的一个非空闭凸集,对于每个 $\boldsymbol{x}\in\boldsymbol{C}$,函数 $f:\boldsymbol{C}\times\boldsymbol{C}\to\mathbf{R}$ 都有 $f(\boldsymbol{x},\boldsymbol{x})=0$. 考虑下面的均衡问题:

求 $\boldsymbol{x}^*\in\boldsymbol{C}$,使得 $f(\boldsymbol{x}^*,\boldsymbol{y})\geqslant 0,\ \forall\,\boldsymbol{y}\in\boldsymbol{C}$

在本文中,假设 $f(\boldsymbol{x},\boldsymbol{y})=f_1(\boldsymbol{x},\boldsymbol{y})+f_2(\boldsymbol{x},\boldsymbol{y})$,其中 $f_i(\boldsymbol{x},\boldsymbol{x})=0\,(i=1,2)$, $\forall\,\boldsymbol{x}\in\boldsymbol{C}$,于是均衡问题转化为:求 $\boldsymbol{x}^*\in\boldsymbol{C}$,使得

$$f_1(\boldsymbol{x}^*,\boldsymbol{y})+f_2(\boldsymbol{x}^*,\boldsymbol{y})\geqslant 0,\ \forall\,\boldsymbol{y}\in\boldsymbol{C}$$

用 $S(f,\boldsymbol{C})$ 表示均衡问题的解集,而且假设解集非空. 许多数学模型都能看成均衡问题的特殊情形,例如:变分不等式、不动点问题、优化问题、鞍点问题、互补问题等. 近年来,求解均衡问题的方法很多,其中外梯度法是一个比较受欢迎的方法. 该方法由 Korpelevich[1] 在解单调变分不等式问题时引入. 随后,为提高这个方法的有效性,它的一些改进方法被提出,如非精确梯度法[2]、投影梯度法[3]、内部外梯度法[4]、黄金比方法[5]、分离法[6-8]等.

本文采用分离算法来解强伪单调的均衡问题. 在文献[7]中,算法的收敛性需要假设每个分离出的函数满足 Hölder 连续. 为避免 Hölder 连续条件,在文献[8]中,Muu 等提出了一个结合梯度方法和 Mann 迭代的分离算法来解均衡问题和非扩张映射的不动点问题,其中函数不需要满足 Lipschitz 连续和 Hölder 连续的条件. 对比文献[8],本文引入惯性(inertial)技术. 惯性思想最早由 Alvarez 等[9]提出,它可以有效地加快邻近点算法的收敛速度. 近年来,惯性思想被广泛地用于各类算法中. 例如:Douglas-Rachford 算子分裂惯性算法[10]、惯性邻近法[11]、邻近梯度法[12]等. 基于分离方法,本文将惯性思想应用到其中,提出分离惯性算法. 同时,结合文献[2-3],其迭代的步长不依赖 Lipschitz 常数.

1　预备知识

定义 1[13]　函数 $f:\boldsymbol{C}\times\boldsymbol{C}\to\mathbf{R}\cup\{+\infty\}$ 被称为

(1)在 $\boldsymbol{C}$ 上强 γ-单调,如果存在常数 $\gamma>0$ 使得

$$f(\boldsymbol{x},\boldsymbol{y})+f(\boldsymbol{y},\boldsymbol{x})\leqslant-\gamma\|\boldsymbol{x}-\boldsymbol{y}\|^2;\ \forall\,\boldsymbol{x},\boldsymbol{y}\in\boldsymbol{C}$$

(2)在 $\boldsymbol{C}$ 上单调,如果

$$f(\boldsymbol{x},\boldsymbol{y})+f(\boldsymbol{y},\boldsymbol{x})\leqslant 0;\ \forall\,\boldsymbol{x},\boldsymbol{y}\in\boldsymbol{C}$$

(3)在 $\boldsymbol{C}$ 上伪单调,如果

$$f(\boldsymbol{x},\boldsymbol{y})\geqslant 0\Rightarrow f(\boldsymbol{y},\boldsymbol{x})\leqslant 0;\ \forall\,\boldsymbol{x},\boldsymbol{y}\in\boldsymbol{C}$$

(4)在 $\boldsymbol{C}$ 上强 γ-伪单调,如果存在常数 $\gamma>0$,

收稿日期: 2019-12-11; **修回日期:** 2020-05-27.
基金项目: 国家自然科学基金资助项目(11801054).
作者简介: 高　辉(1978-),女,博士生,E-mail:gaohui@dlou.edu.cn;庞丽萍*(1968-),女,教授,博士生导师,E-mail:lppang@dlut.edu.cn.

且 $f(\boldsymbol{x},\boldsymbol{y})\geqslant 0$，则

$$f(\boldsymbol{y},\boldsymbol{x})\leqslant -\gamma\|\boldsymbol{x}-\boldsymbol{y}\|^2;\ \forall\, \boldsymbol{x},\boldsymbol{y}\in \boldsymbol{C}$$

定义 2[13]　设 $g:\boldsymbol{R}^n\to\mathbf{R}\cup\{+\infty\}$ 是正常下半连续凸函数，$t>0$，函数 g 在 $\boldsymbol{x}$ 处的邻近映射定义为

$$\mathrm{prox}_{tg}(\boldsymbol{x}):=\arg\min\left\{tg(\boldsymbol{y})+\frac{1}{2}\|\boldsymbol{x}-\boldsymbol{y}\|^2\right\}$$

引理 1[14]　设 $g:\boldsymbol{R}^n\to\mathbf{R}\cup\{+\infty\}$ 是正常下半连续凸函数，$\boldsymbol{u}\in\boldsymbol{R}^n$，$t>0$，令 $\boldsymbol{v}=\mathrm{prox}_{tg}(\boldsymbol{u})$，则

$$tg(\boldsymbol{w})-tg(\boldsymbol{v})\geqslant\langle \boldsymbol{u}-\boldsymbol{v},\boldsymbol{w}-\boldsymbol{v}\rangle;\ \forall\, \boldsymbol{w}\in\boldsymbol{R}^n$$

而且，进一步有

$$g(\boldsymbol{w})-g(\boldsymbol{v})\geqslant(\|\boldsymbol{u}-\boldsymbol{v}\|^2+\|\boldsymbol{w}-\boldsymbol{v}\|^2-\|\boldsymbol{u}-\boldsymbol{w}\|^2)/2t;\ \forall\, \boldsymbol{w}\in\boldsymbol{R}^n \tag{1}$$

引理 2[15]　假设 $\{\alpha_k\}_{k=0}^{+\infty}$，$\{\gamma_k\}_{k=0}^{+\infty}$，$\{\delta_k\}_{k=0}^{+\infty}$ 都是非负实序列，且

$$\alpha_{k+1}\leqslant(1-\gamma_k)\alpha_k+\gamma_k\alpha_{k-1}+\delta_k$$

其中 $\{\gamma_k\}_{k=0}^{+\infty}\subset\left[0,\frac{1}{2}\right]$，$\sum_{k=0}^{+\infty}\delta_k<+\infty$，则序列 $\{\alpha_k\}_{k=0}^{+\infty}$ 收敛.

引理 3[15]　假设 $\{\alpha_k\}_{k=0}^{+\infty}$，$\{\gamma_k\}_{k=0}^{+\infty}$，$\{\delta_k\}_{k=0}^{+\infty}$ 和 $\{t_k\}_{k=0}^{+\infty}$ 都是非负实序列，且

$$\alpha_{k+1}\leqslant(1-t_k-\gamma_k)\alpha_k+\gamma_k\alpha_{k-1}+\delta_k$$

其中 $\sum_{k=0}^{+\infty}t_k=+\infty$，$\{\gamma_k\}_{k=0}^{+\infty}\subset\left[0,\frac{1}{2}\right]$ 是递减序列，$\sum_{k=0}^{+\infty}\delta_k<+\infty$，则序列 $\{\alpha_k\}_{k=0}^{+\infty}$ 收敛于 0.

2　算法和收敛性

为证明算法的收敛性，做如下假设：

(1) 每个 $\boldsymbol{x}\in\boldsymbol{C}$，函数 $f_i(\boldsymbol{x},\cdot)(i=1,2)$ 是下半连续凸函数；

(2) 函数 f 在 $\boldsymbol{C}$ 上强 γ-伪单调；

(3) 如果 $\{\boldsymbol{x}^k\}\subset\boldsymbol{C}$ 有界，则序列 $\{\boldsymbol{g}_i{}^k\in\partial(f_i(\boldsymbol{x}^k,\cdot))(\boldsymbol{x}^k)\}(i=1,2)$ 有界.

算法 1

初始化：选择两个非负递减序列 $\{\alpha_n\}_{n=0}^{+\infty}\subset\left[0,\frac{1}{2}\right]$ 和 $\{\beta_n\}_{n=0}^{+\infty}$，满足

$$\sum_{n=0}^{+\infty}\beta_n=+\infty,\ \sum_{n=0}^{+\infty}\beta_n^2<+\infty \tag{2}$$

迭代步：给定 $\boldsymbol{x}_{n-1},\boldsymbol{x}_n\in\boldsymbol{C}$，计算

$$\boldsymbol{w}_n=\boldsymbol{x}_n+\alpha_n(\boldsymbol{x}_{n-1}-\boldsymbol{x}_n) \tag{3}$$

选取 $\boldsymbol{g}_1^n\in\partial(f_1(\boldsymbol{w}_n,\cdot))(\boldsymbol{w}_n)$，$\boldsymbol{g}_2^n\in\partial(f_2(\boldsymbol{w}_n,\cdot))(\boldsymbol{w}_n)$.

计算：

$$\eta_n=\max\{1,\|\boldsymbol{g}_1^n\|,\|\boldsymbol{g}_2^n\|\},\lambda_n=\frac{\beta_n}{\eta_n} \tag{4}$$

和

$$\boldsymbol{y}_n=\arg\min\left\{\lambda_n f_1(\boldsymbol{w}_n,\boldsymbol{y})+\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{w}_n\|^2\mid \boldsymbol{y}\in\boldsymbol{C}\right\},$$
$$\boldsymbol{x}_{n+1}=\arg\min\left\{\lambda_n f_2(\boldsymbol{w}_n,\boldsymbol{y})+\frac{1}{2}\|\boldsymbol{y}-\boldsymbol{y}_n\|^2\mid \boldsymbol{y}\in\boldsymbol{C}\right\} \tag{5}$$

如果 $\boldsymbol{x}_{n+1}=\boldsymbol{y}_n=\boldsymbol{w}_n$，则算法停止.

关于算法 1 解释如下：

(1) 对于 $\boldsymbol{w}_n=\boldsymbol{x}_n+\alpha_n(\boldsymbol{x}_{n-1}-\boldsymbol{x}_n)$，$0\leqslant\alpha_n<1$，其中 $\boldsymbol{w}_n$ 是 $\boldsymbol{x}_n$ 和 $\boldsymbol{x}_{n-1}$ 的一个凸组合，本文 $\boldsymbol{w}_n$ 与文献[15]相同. 当然对于 $\boldsymbol{w}_n$ 还有其他选择，如在文献[3]中，$\boldsymbol{w}_n=\boldsymbol{x}_n+\alpha_n(\boldsymbol{x}_n-\boldsymbol{x}_{n-1})$，$0\leqslant\alpha_n<1$，其中 $\alpha_n(\boldsymbol{x}_n-\boldsymbol{x}_{n-1})$ 被称为惯性效果，可以加速算法的收敛性.

(2) 当 $\alpha_n=0$，本文算法是不带加速步的分离算法.

定理 1　假设 $\{\boldsymbol{x}_n\}$ 是由本文算法生成的序列，对于每个 $\boldsymbol{y}\in\boldsymbol{C}$，有

$$\|\boldsymbol{x}_{n+1}-\boldsymbol{y}\|^2\leqslant\|\boldsymbol{w}_n-\boldsymbol{y}\|^2+2\lambda_n f(\boldsymbol{w}_n,\boldsymbol{y})+5\beta_n^2$$

证明　根据式(1)，且 $t=\lambda_n$，$g(\cdot)=f_1(\boldsymbol{w}_n,\cdot)$，$\boldsymbol{w}=\boldsymbol{y}$，$\boldsymbol{v}=\boldsymbol{y}_n$，$\boldsymbol{u}=\boldsymbol{w}_n$，则

$$\lambda_n(f_1(\boldsymbol{w}_n,\boldsymbol{y})-f_1(\boldsymbol{w}_n,\boldsymbol{y}_n))\geqslant(\|\boldsymbol{w}_n-\boldsymbol{y}_n\|^2+\|\boldsymbol{y}-\boldsymbol{y}_n\|^2-\|\boldsymbol{w}_n-\boldsymbol{y}\|^2)/2;\ \forall\, \boldsymbol{y}\in\boldsymbol{C}$$

整理得

$$\|\boldsymbol{y}-\boldsymbol{y}_n\|^2\leqslant\|\boldsymbol{w}_n-\boldsymbol{y}\|^2+2\lambda_n f_1(\boldsymbol{w}_n,\boldsymbol{y})-2\lambda_n f_1(\boldsymbol{w}_n,\boldsymbol{y}_n)-\|\boldsymbol{y}_n-\boldsymbol{w}_n\|^2;\ \forall\, \boldsymbol{y}\in\boldsymbol{C} \tag{6}$$

对于式(5)中的 $\boldsymbol{x}_{n+1}$，按照类似的方法，有

$$\|\boldsymbol{x}_{n+1}-\boldsymbol{y}\|^2\leqslant\|\boldsymbol{y}-\boldsymbol{y}_n\|^2+2\lambda_n f_2(\boldsymbol{w}_n,\boldsymbol{y})-2\lambda_n f_2(\boldsymbol{w}_n,\boldsymbol{x}_{n+1})-\|\boldsymbol{x}_{n+1}-\boldsymbol{y}_n\|^2 \tag{7}$$

式(6)和(7)相加，则有

$$\|\boldsymbol{x}_{n+1}-\boldsymbol{y}\|^2\leqslant\|\boldsymbol{w}_n-\boldsymbol{y}\|^2-\|\boldsymbol{y}_n-\boldsymbol{w}_n\|^2-\|\boldsymbol{x}_{n+1}-\boldsymbol{y}_n\|^2+2\lambda_n f(\boldsymbol{w}_n,\boldsymbol{y})-2\lambda_n(f_1(\boldsymbol{w}_n,\boldsymbol{y}_n)+f_2(\boldsymbol{w}_n,\boldsymbol{x}_{n+1})) \tag{8}$$

又 $\boldsymbol{g}_1^n\in\partial(f_1(\boldsymbol{w}_n,\cdot))(\boldsymbol{w}_n)$，$f_1(\boldsymbol{w}_n,\boldsymbol{w}_n)=0$，所以

$$f_1(\boldsymbol{w}_n,\boldsymbol{y}_n)=f_1(\boldsymbol{w}_n,\boldsymbol{y}_n)-f_1(\boldsymbol{w}_n,\boldsymbol{w}_n)\geqslant$$

$$\langle \boldsymbol{g}_1^n, \boldsymbol{y}_n - \boldsymbol{w}_n \rangle$$

进一步可推出

$$-2\lambda_n f_1(\boldsymbol{w}_n, \boldsymbol{y}_n) \leqslant -2\lambda_n \langle \boldsymbol{g}_1^n, \boldsymbol{y}_n - \boldsymbol{w}_n \rangle \leqslant 2\beta_n \|\boldsymbol{y}_n - \boldsymbol{w}_n\| \tag{9}$$

其中式(9)的第2个不等式由柯西-施瓦茨不等式和式(4)得到.

类似地,估计式(8)的$-2\lambda_n f_2(\boldsymbol{w}_n, \boldsymbol{x}_{n+1})$为

$$-2\lambda_n f_2(\boldsymbol{w}_n, \boldsymbol{x}_{n+1}) \leqslant 2\beta_n \|\boldsymbol{x}_{n+1} - \boldsymbol{w}_n\| \leqslant 2\beta_n(\|\boldsymbol{x}_{n+1} - \boldsymbol{y}_n\| + \|\boldsymbol{y}_n - \boldsymbol{w}_n\|) \tag{10}$$

将式(9)、(10)代入式(8),得

$$\begin{aligned}\|\boldsymbol{x}_{n+1} - \boldsymbol{y}\|^2 \leqslant & \|\boldsymbol{w}_n - \boldsymbol{y}\|^2 + 2\lambda_n f(\boldsymbol{w}_n, \boldsymbol{y}) - \\ & \|\boldsymbol{y}_n - \boldsymbol{w}_n\|^2 - \|\boldsymbol{x}_{n+1} - \boldsymbol{y}_n\|^2 + \\ & 4\beta_n \|\boldsymbol{y}_n - \boldsymbol{w}_n\| + 2\beta_n \|\boldsymbol{x}_{n+1} - \boldsymbol{y}_n\| = \\ & \|\boldsymbol{w}_n - \boldsymbol{y}\|^2 + 2\lambda_n f(\boldsymbol{w}_n, \boldsymbol{y}) + 5\beta_n^2 - \\ & (\|\boldsymbol{y}_n - \boldsymbol{w}_n\| - 2\beta_n)^2 - \\ & (\|\boldsymbol{x}_{n+1} - \boldsymbol{y}_n\| - \beta_n)^2 \leqslant \\ & \|\boldsymbol{w}_n - \boldsymbol{y}\|^2 + 2\lambda_n f(\boldsymbol{w}_n, \boldsymbol{y}) + 5\beta_n^2\end{aligned}$$

□

定理2 假设条件(1)~(3)成立,由本文算法生成的序列$\{\boldsymbol{x}_n\}$强收敛于均衡问题的解.

证明 假设$\boldsymbol{p} \in S(f, \boldsymbol{C})$,由定理1知,

$$\|\boldsymbol{x}_{n+1} - \boldsymbol{p}\|^2 \leqslant \|\boldsymbol{w}_n - \boldsymbol{p}\|^2 + 2\lambda_n f(\boldsymbol{w}_n, \boldsymbol{p}) + 5\beta_n^2 \tag{11}$$

因为f强γ-伪单调,式(11)可转换成

$$\|\boldsymbol{x}_{n+1} - \boldsymbol{p}\|^2 \leqslant (1 - 2\gamma\lambda_n)\|\boldsymbol{w}_n - \boldsymbol{p}\|^2 + 5\beta_n^2 \tag{12}$$

另一方面,使用$\|\cdot\|^2$的凸性,则

$$\|\boldsymbol{w}_n - \boldsymbol{p}\|^2 = \|(1-\alpha_n)(\boldsymbol{x}_n - \boldsymbol{p}) + \alpha_n(\boldsymbol{x}_{n-1} - \boldsymbol{p})\|^2 \leqslant (1-\alpha_n)\|\boldsymbol{x}_n - \boldsymbol{p}\|^2 + \alpha_n\|\boldsymbol{x}_{n-1} - \boldsymbol{p}\|^2 \tag{13}$$

结合式(12)、(13),推出

$$\begin{aligned}\|\boldsymbol{x}_{n+1} - \boldsymbol{p}\|^2 \leqslant & [1 - 2\gamma\lambda_n - \alpha_n(1 - 2\gamma\lambda_n)]\|\boldsymbol{x}_n - \boldsymbol{p}\|^2 + \\ & \alpha_n(1 - 2\gamma\lambda_n)\|\boldsymbol{x}_{n-1} - \boldsymbol{p}\|^2 + 5\beta_n^2 \leqslant \\ & [1 - \alpha_n(1 - 2\gamma\lambda_n)]\|\boldsymbol{x}_n - \boldsymbol{p}\|^2 + \\ & \alpha_n(1 - 2\gamma\lambda_n)\|\boldsymbol{x}_{n-1} - \boldsymbol{p}\|^2 + 5\beta_n^2\end{aligned} \tag{14}$$

因此,$A_{n+1} \leqslant (1-\gamma_n)A_n + \gamma_n A_{n-1} + \delta_n$,其中$A_n = \|\boldsymbol{x}_n - \boldsymbol{p}\|^2$,$\gamma_n = \alpha_n(1 - 2\gamma\lambda_n) \in \left[0, \frac{1}{2}\right]$,$\delta_n = 5\beta_n^2$. 根据引理2,则$\lim\limits_{n \to +\infty}\|\boldsymbol{x}_n - \boldsymbol{p}\|^2$存在. 从而,$\{\boldsymbol{x}_n\}$是有界序列. 由式(3)知,$\{\boldsymbol{w}_n\}$也是一个有界序列. 根据假设条件(3),存在$M_1 > 0$,$M_2 > 0$,使得$\|\boldsymbol{g}_1^n\| \leqslant M_1$,$\|\boldsymbol{g}_2^n\| \leqslant M_2$.

令$M := \max\{1, M_1, M_2\}$,结合式(4),则$\beta_n \geqslant \lambda_n = \frac{\beta_n}{\eta_n} \geqslant \frac{1}{M}\beta_n$. 由式(2)知,

$$\sum_{n=0}^{+\infty} \lambda_n = +\infty \tag{15}$$

整理式(14)得

$$\|\boldsymbol{x}_{n+1} - \boldsymbol{p}\|^2 \leqslant [1 - 2\gamma\lambda_n(1-\alpha_n) - \alpha_n]\|\boldsymbol{x}_n - \boldsymbol{p}\|^2 + \alpha_n\|\boldsymbol{x}_{n-1} - \boldsymbol{p}\|^2 + 5\beta_n^2 \tag{16}$$

结合式(15)、(16)和引理3,且$t_n = 2\gamma\lambda_n - 2\gamma\lambda_n\alpha_n$,$\delta_n = 5\beta_n^2$,可推出

$$\lim_{n \to +\infty}\|\boldsymbol{x}_n - \boldsymbol{p}\|^2 = 0$$

□

3 数值实验

本文通过初步数值实验来说明算法的可行性和有效性. 在数值实验中,采用Matlab R2015b软件编写程序,软件的运行环境为PC Desktop Intel(R) Core(TM) i5-8250U CPU @ 1.60 GHz, RAM 8.00 GB.

考虑均衡问题满足

$$f: \boldsymbol{R}^5 \times \boldsymbol{R}^5 \to \mathbf{R},\ f = f_1 + f_2$$

$$f_1(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{P}\boldsymbol{x} + \boldsymbol{Q}\boldsymbol{y} + \boldsymbol{q}, \boldsymbol{y} - \boldsymbol{x} \rangle$$

$$f_2(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{y}\|^4 - \|\boldsymbol{x}\|^4$$

其中

$$\boldsymbol{P} = \begin{pmatrix} 3.1 & 2.0 & 0 & 0 & 0 \\ 2.0 & 3.6 & 0 & 0 & 0 \\ 0 & 0 & 3.5 & 2.0 & 0 \\ 0 & 0 & 2.0 & 3.3 & 0 \\ 0 & 0 & 0 & 0 & 3.0 \end{pmatrix}$$

$$\boldsymbol{Q} = \begin{pmatrix} 1.6 & 1.0 & 0 & 0 & 0 \\ 1.0 & 1.6 & 0 & 0 & 0 \\ 0 & 0 & 1.5 & 1.0 & 0 \\ 0 & 0 & 1.0 & 1.5 & 0 \\ 0 & 0 & 0 & 0 & 2.0 \end{pmatrix}$$

$$\boldsymbol{q} = (-1\quad -2\quad -1\quad 2\quad -1)^{\mathrm{T}}$$

可行集$\boldsymbol{C} = \{\boldsymbol{x} \in \boldsymbol{R}^5 : \|\boldsymbol{x} - (5\quad -3\quad -2\quad 4\quad 2)^{\mathrm{T}}\| \leqslant 1\}$.

例1 研究本文算法的数值效果. 从本文算法可知,若$\boldsymbol{x}_{n+1} = \boldsymbol{y}_n = \boldsymbol{w}_n$,则$\boldsymbol{x}_{n+1}$是均衡问题的

解. 因此，使用

$$D_n = \|x_{n+1} - y_n\|^2 + \|y_n - w_n\|^2; \ n = 0, 1, \cdots$$

作为本文算法终止的判别量. 选取 $\alpha_n = \frac{1}{n+1}$，$\beta_n = \frac{1}{(n+1)^p}$ ($p = 0.3, 0.5, 1.0$)，初始点 $x_0 = (1 \quad 1 \quad 1 \quad 1 \quad 1)^{\mathrm{T}}$，$\varepsilon = 10^{-3}$，停止准则 $\max\{\|x_{n+1} - y_n\|, \|y_n - w_n\|\} \leqslant \varepsilon$.

由表 1 可以看出，实验结果跟 p 的取值有关. 在给定停止准则，且 $p = 1.0$ 时，本文算法迭代次数最少.

表 1　本文算法数值实验结果

Tab. 1　Numerical experimental results of the algorithm in this paper

p	迭代次数	t_{CPU}/s
0.3	28	1.51
0.5	34	1.65
1.0	21	1.21

例 2　比较本文算法和另两个算法，分别为不带加速步的本文算法（记为 SA）和文献[4]的内部外梯度算法（记为 IEGA）. 在本文算法中，$\alpha_n = \beta_n = \frac{1}{n+1}$；在算法 SA 中，$\beta_n = \frac{1}{n+1}$ 和 $\alpha_n = 0$；算法 IEGA 的 $\mu = 1, \nu = 7, \varphi(t) = t - \ln t - 1$. 3 个算法中初始点 $x_0 = (1 \quad 1 \quad 1 \quad 1 \quad 1)^{\mathrm{T}}$，$\varepsilon = 10^{-3}$，停止准则

算法 SA：$\max\{\|x_{n+1} - y_n\|, \|y_n - x_n\|\} \leqslant \varepsilon$

算法 IEGA：$\|x_n - y_n\| \leqslant \varepsilon$

由表 2 可以看出，本文算法比 SA 在迭代次数和所用时间上都要少. 本文算法比 IEGA 在迭代过程中 CPU 运行的时间少. 通过比较，说明了本文算法的有效性.

表 2　本文算法、SA、IEGA 的比较

Tab. 2　Comparison on the algorithm in this paper, SA and IEGA

算法	迭代次数	t_{CPU}/s
本文算法	21	1.21
SA	116	3.48
IEGA	15	4.24

例 3　考虑均衡问题满足

$$f: \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}, \ f = f_1 + f_2$$

$$f_1(x, y) = \langle Ax, y - x \rangle$$

$$f_2(x, y) = \|y\|^4 - \|x\|^4$$

其中 $A = BB^{\mathrm{T}} + n^2 I_n$，$I_n$ 是单位矩阵，矩阵 B 是元素全为 1 的 n 阶方阵. 可行集 $C = \{x \in \mathbf{R}^n \mid \|x\| \leqslant 1\}$，比较本文算法和文献[15]的算法（记为 EGA）. 在两个算法中，$x_0 = (1 \quad 2 \quad \cdots \quad n)^{\mathrm{T}}$，停止准则 $\|x_k - x^*\| \leqslant 10^{-3}$，其中 $x^* = (0 \quad 0 \quad \cdots \quad 0)^{\mathrm{T}}$，是该问题的最优解.

由表 3 可以看出，本文算法尽管需要多的迭代次数，但在 CPU 运行时间上都比 EGA 少，说明本文算法对维数高的算例也是有效的.

表 3　本文算法和 EGA 的比较

Tab. 3　Comparison between the algorithm in this paper and EGA

n	算法	迭代次数	t_{CPU}/s
50	本文算法	45	1.92
	EGA	35	2.67
100	本文算法	47	3.75
	EGA	35	4.78
500	本文算法	101	41.25
	EGA	39	56.88

4　结　语

本文采用分离算法来解强伪单调的非光滑均衡问题. 本文算法结合了惯性，同时，其迭代步长不依赖 Lipschitz 常数，在满足一定条件的假设下，证明了本文算法的强收敛性. 与已有的几个算法进行比较，说明了本文算法的有效性.

参考文献：

[1]　KORPELEVICH G M. The extragradient method for finding saddle points and other problems [J]. **Ekonomikai Matematicheskie Metody**, 1976, **12**: 747-756.

[2]　SANTOS P, SCHEIMBERG S. An inexact subgradient algorithm for equilibrium problems [J]. **Computational and Applied Mathematics**, 2011, **30**: 91-107.

[3]　VINH N T, GIBALI A. Gradient projection-type algorithms for solving equilibrium problems and its

applications [J]. **Computational and Applied Mathematics**, 2019, **38**: 119-137.

[4] VAN NGUYEN T T, STRODIOT J J, NGUYEN V H. The interior proximal extragradient method for solving equilibrium problems [J]. **Journal of Global Optimization**, 2009, **44**(2): 175-192.

[5] VINH N T. Golden ratio algorithms for solving equilibrium problems in Hilbert spaces [EB/OL]. (2018-04-05). https://arxiv.org/abs/1804.01829.

[6] PHAM K A, HAI T N. Splitting extragradient-like algorithms for strongly pseudomonotone equilibrium problems [J]. **Numerical Algorithms**, 2017, **76**: 67-91.

[7] HAI T N, VINH N T. Two new splitting algorithms for equilibrium problems [J]. **Revista De La Real Academia De Ciencias Exactas, Fisicas Y Naturales. Serie A. Mathematicas**, 2017, **111**: 1051-1069.

[8] MUU L D, LE X T. A splitting algorithm for finding fixed points of nonexpansive mappings and solving equilibrium problems [J]. **Journal of Fixed Point Theory and Applications**, 2018, **20**: 130-146.

[9] ALVAREZ F, ATTOUCH H. An inertial proximal method for maximal monotone operators via discretization of a nonlinear oscillator with damping [J]. **Set-Valued Analysis**, 2001, **9**: 3-11.

[10] BOT R I, CSETNEK E R, HENDRICH C. Inertial Douglas-Rachford splitting for monotone inclusion problems [J]. **Applied Mathematics and Computation**, 2015, **256**: 472-487.

[11] VAN HIEU D. An inertial-like proximal algorithm for equilibrium problems [J]. **Mathematical Methods of Operations Research**, 2018, **88**(3): 399-415.

[12] WEN Bo, CHEN Xiaojun, PONG T K. Linear convergence of proximal gradient algorithm with extrapolation for a class of nonconvex nonsmooth minimization problems [J]. **SIAM Journal on Optimization**, 2017, **27**(1): 124-145.

[13] BLUM E, OETTLI W. From optimization and variational inequalities to equilibrium problems [J]. **Mathematical Programming**, 1994, **63**: 123-145.

[14] CHBANI Z, RIAHI H. Weak and strong convergence of an inertial proximal method for solving Ky Fan minimax inequalities [J]. **Optimization Letters**, 2013, **7**: 185-206.

[15] QUOC T D, MUU L D, NGUYEN V H. Extragradient algorithms extended to equilibrium problems [J]. **Optimization**, 2008, **57**(6): 749-776.

An inertial splitting algorithm for solving equilibrium problems

GAO Hui[1,2], ZHANG Mingkun[1], WANG Xiaoliang[1], PANG Liping[*1]

(1. School of Mathematical Sciences, Dalian University of Technology, Dalian 116024, China;
2. School of Information Engineering, Dalian Ocean University, Dalian 116023, China)

Abstract: Nonsmooth equilibrium problems given by a sum of two functions are studied and a splitting algorithm combining an inertial method is proposed for equilibrium problems. At each iteration, two simple strongly convex subproblems are solved alternatively. Without any Lipschitz continuous condition or Hölder continuity of the involved functions, the convergence of the algorithm is proved. Several experiments are performed to show the computational efficiency compared with related algorithms.

Key words: sum of two functions; equilibrium problem; splitting algorithm; inertial