

广义鞍点问题的改进的类 SOR 算法^{*1)}

张 纯

(南京师范大学数学科学学院, 南京 210023;
中国人民解放军陆军工程大学基础部, 南京 211101)

贾泽慧

(南京信息工程大学数学与统计学院, 南京 210044)

蔡邢菊

(南京师范大学数学科学学院, 南京 210023)

韩德仁

(北京航空航天大学数学科学学院, 北京 100191)

摘 要

针对广义鞍点问题, 本文提出了一个改进的类逐次超松弛迭代算法, 在较弱的条件下, 分析了算法的收敛性及线性收敛率. 新算法的每步计算量与已有的算法类似, 都是需要 (近似) 求解线性方程组, 但新算法有更好的灵活度通过合适地选取参数矩阵, 每一步子问题可以容易地求解, 甚至可以有闭式解 (closed-form solution). 数值实验结果显示了新算法的有效性.

关键词: 鞍点问题; 类 SOR 算法; 全局收敛性; 收敛率

MR (2000) 主题分类: 65F10, 65H10

1. 引 言

我们考虑如下的广义鞍点问题:

$$\mathcal{A}u := \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} f \\ -g \end{pmatrix}, \quad (1.1)$$

其中 $A \in \mathbb{R}^{m \times m}$ 是对称正定矩阵, $C \in \mathbb{R}^{n \times n}$ 是非零对称半正定矩阵, $B \in \mathbb{R}^{m \times n} (m > n)$ 是列满秩矩阵, $x, f \in \mathbb{R}^m, y, g \in \mathbb{R}^n$. 为讨论方便, 我们记

$$u := \begin{pmatrix} x \\ y \end{pmatrix}, \quad F(u) := \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u - \begin{pmatrix} f \\ -g \end{pmatrix}. \quad (1.2)$$

假设 (1.1) 的解集 $\Omega^* = \{u \mid F(u) = 0\}$ 非空. 我们用 $\mathbb{R}, \mathbb{R}_+, \mathbb{R}_{++}$ 分别表示实数, 非负实数, 正实数, 用 $\mathbb{R}^n$ 和 $\mathbb{R}^{m \times n}$ 分别表示 n 维实向量空间和 $m \times n$ 维实矩阵空间. 对矩阵 $M \in \mathbb{R}^{m \times n}$, 向量 $v \in \mathbb{R}^n$, 我们用 M^T, v^T 分别表示 M, v 的转置, $\|M\| = \sqrt{\lambda_{\max}(M^T M)}$ 表示相应的

^{*} 2018 年 3 月 7 日收到.

¹⁾ 基金项目: 国家自然科学基金 (11625105, 11926358, 11871279, 11571178, 11801279), 江苏省自然科学基金 (BK2018078), 南京信息工程大学科研启动基金 (2017r059).

范数. 对向量 $u, v \in \mathbb{R}^n$, 用 $u^T v$ 表示 u, v 的内积, $\|u\| = \sqrt{u^T u}$ 表示相应的范数. 对矩阵 $X, Y \in \mathbb{R}^{n \times n}$, 用 $X \succ Y$ 表示矩阵 $X - Y$ 正定. 大规模广义鞍点问题应用于计算科学与工程领域的许多领域. 如参数识别问题^[11, 15], 约束加权最小二乘问题^[7, 10], 内点法的鞍点系统^[4], 最优控制^[6, 12], 电路与网络^[5, 16] 等等.

对于广义鞍点问题, 当 $A \in \mathbb{R}^{m \times m}$, $C \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{m \times n}$ ($m > n$) 是大型稀疏矩阵时, 迭代解法将比直接求解更为有效. 当 $C = 0$ 时, 广义鞍点问题 (1.1) 退化为鞍点问题, 对这类问题的迭代解法已经有相当多的研究, 比如参数化非精确 Uzawa 方法 (简称 PIU 算法)^[8], 广义逐次超松弛迭代算法 (简称 GSOR 算法)^[2], 局部对称斜对称分裂的迭代算法 (简称 HLSS 迭代算法)^[13] 等. 当 $C \neq 0$ 时, 也有相当多的研究^[3, 14] 等.

白中治等人^[3] 将求解鞍点问题的参数化非精确 Uzawa 方法^[2] 推广到求解广义鞍点问题中去, 提出了广义的 PIU 算法 (简称 GPIU 算法). 并在矩阵 A 正定, B 列满秩, $C = O$ 或 $C = \kappa Q$ (其中 Q 是奇异值分解矩阵 $B^T P^{-1} B + C$ 的近似矩阵), κ 是非零实常数, P 正定及相关参数假设条件下, 给出了 GPIU 算法的收敛性证明, 其算法和迭代的矩阵形式如下:

Algorithm 1: 求广义鞍点问题 (1.1) 的 GPIU 算法

Input: 设 $P \in \mathbb{R}^{m \times m}$, $Q \in \mathbb{R}^{n \times n}$ 是对称正定矩阵, 非零常数 $\eta, \theta \in \mathbb{R}$, 给定初始向量 $x_0 \in \mathbb{R}^m$, $y_0 \in \mathbb{R}^n$, 迭代误差 ERR, 误差界 ε , 最大迭代次数 $K_{\max}$.

```

1 while ERR >  $\varepsilon$  且  $k < K_{\max}$  do
2    $x_{k+1} = x_k + \eta P^{-1} (f - Ax_k - By_k)$ 
3    $y_{k+1} = y_k + \theta Q^{-1} (B^T x_{k+1} - Cy_k - g)$ 
4 end
```

将 GPIU 算法写成矩阵的形式, 即

$$\begin{pmatrix} \frac{1}{\eta}P & O \\ -B^T & \frac{1}{\theta}Q \end{pmatrix} \begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} \frac{1}{\eta}P - A & -B \\ O & \frac{1}{\theta}Q - C \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} f \\ -g \end{pmatrix}.$$

蒋美群等人^[13] 通过采用分裂预处理的方法提出了广义的修正局部对称斜对称分裂的迭代算法 (简称 GMLHSS 迭代算法), 其中 A 可以为非对称矩阵, 但 $H := \frac{1}{2}(A + A^T)$ 正定, B 为列满秩, C 为对称半正定矩阵. 对矩阵 C 进行分解

$$C = \begin{pmatrix} E & F \end{pmatrix} \begin{pmatrix} D & O \\ O & O \end{pmatrix} \begin{pmatrix} E^T \\ F^T \end{pmatrix}, \quad (1.3)$$

则相应地将矩阵 B 转化成 (B_1, B_2) , 其算法和矩阵迭代格式如下:

将 GMLHSS 算法写成矩阵的形式, 即

$$\begin{pmatrix} Q_1 + H & O & O \\ O & Q_2 + D & O \\ -B_2^T & O & Q_3 \end{pmatrix} \begin{pmatrix} x_{k+1} \\ z_{k+1} \\ p_{k+1} \end{pmatrix} = \begin{pmatrix} Q_1 - S & -B_1 & -B_2 \\ B_1^T & Q_2 & O \\ O & O & Q_3 \end{pmatrix} \begin{pmatrix} x_k \\ z_k \\ p_k \end{pmatrix} + \begin{pmatrix} f \\ g_1 \\ g_2 \end{pmatrix}.$$

Algorithm 2: 求广义鞍点问题 (1.1) 的 GMLHSS 算法

Input: 设 $Q_1 \in \mathbb{R}^{m \times m}$, $Q_2 \in \mathbb{R}^{n \times n}$ 是对称半正定矩阵, $D \in \mathbb{R}^{r \times r}$ 对角正定矩阵, $Q_3 \in \mathbb{R}^{(n-r) \times (n-r)}$ 是对称正定矩阵, $H = \frac{1}{2}(A + A^T) \in \mathbb{R}^{m \times m}$, $S = \frac{1}{2}(A - A^T) \in \mathbb{R}^{m \times m}$, $B_1 \in \mathbb{R}^{r \times m}$, $B_2 \in \mathbb{R}^{(n-r) \times m}$, $g_1 \in \mathbb{R}^r$, $g_2 \in \mathbb{R}^{n-r}$. 给定初始向量 $x_0 \in \mathbb{R}^m$, $z_0 \in \mathbb{R}^r$ 和 $p_0 \in \mathbb{R}^{n-r}$, 迭代误差 ERR, 误差界 ε , 最大迭代次数 $K_{\max}$.

```

1 while ERR >  $\varepsilon$  且  $k < K_{\max}$  do
2    $x_{k+1} = x_k - (Q_1 + H)^{-1}(f - Ax_k - B_1 z_k - B_2 p_k)$ 
3    $z_{k+1} = z_k + (Q_2 + D)^{-1}(g_1 + B_1^T x_k - Dz_k)$ 
4    $p_{k+1} = p_k + Q_3^{-1}(B_2^T x_{k+1} + g_2)$ 
5    $y_{k+1} = (z_{k+1}^T, p_{k+1}^T)^T$ 
6 end

```

对于广义鞍点问题 (1.1), 刘丽华等人^[14] 在矩阵 A 正定, C 半正定的条件下提出了新的逐次超松弛迭代算法 (简称 NSOR 算法), 但是仅在 A, C 正定的情况下给出了此算法的收敛性证明及数值实验结果, 其算法和矩阵迭代格式如下:

Algorithm 3: 求广义鞍点问题 (1.1) 的 NSOR 算法

Input: 设 $Q_1 \in \mathbb{R}^{m \times m}$, $Q_2 \in \mathbb{R}^{n \times n}$ 是正定矩阵, 给定初始向量 $x_0 \in \mathbb{R}^m$, $y_0 \in \mathbb{R}^n$ 和松弛参数 $\omega \neq 0$, $\sigma \neq 1$, $\alpha \neq 1$, $\alpha\sigma \neq 1$, 迭代误差 ERR, 误差界 ε , 最大迭代次数 $K_{\max}$.

```

1 while ERR >  $\varepsilon$  且  $k < K_{\max}$  do
2    $x_{k+1} = x_k + \omega Q_1^{-1}(f - Ax_k - By_k)$ 
3    $y_{k+1} = \left(I - \frac{\sigma}{1-\alpha\sigma} Q_2^{-1} C\right) y_k + \frac{\sigma}{1-\alpha\sigma} Q_2^{-1} (B^T x_{k+1} - g)$ 
4 end

```

将 NSOR 算法写成矩阵的形式, 即

$$\begin{pmatrix} Q_1 & O \\ -\sigma B^T & (1-\alpha\sigma)Q_2 \end{pmatrix} \begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} Q_1 - \omega A & -\omega B \\ O & (1-\alpha\sigma)Q_2 - \sigma C \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} \omega f \\ -\sigma g \end{pmatrix}.$$

本文中, 我们提出求解广义鞍点问题 (1.1) 的改进的类逐次超松弛迭代算法 (简称 NCSOR 算法), 在矩阵 A 正定, C 半正定的条件下给出 NCSOR 算法的收敛性证明, 并进一步地证明其线性收敛率. 我们将 NCSOR 算法与 GPIU 算法和 NSOR 算法进行比较, 在数值上有一定的优越性.

本文的结构如下, 第一节引言, 第二节提出求解广义鞍点问题 (1.1) 的改进的 NCSOR 算法, 第三节分析 NCSOR 算法的收敛性及线性收敛率, 第四节给出算法的数值实验结果, 第五节做了总结.

2. NCSOR 算法

本文考虑一个改进的迭代格式, 即 NCSOR 算法:

Algorithm 4: 求广义鞍点问题 (1.1) 的 NCSOR 算法

Input: 设 $R \in \mathbb{R}^{m \times m}$, $S \in \mathbb{R}^{n \times n}$ 是正定矩阵, 给定初始向量 $x^0 \in \mathbb{R}^m$, $y^0 \in \mathbb{R}^n$, 迭代误差 ERR, 误差界 ε , 最大迭代次数 $K_{\max}$.

```

1 while ERR >  $\varepsilon$  且  $k < K_{\max}$  do
2    $x_{k+1} = (A + R)^{-1} (Rx_k - By_k + f)$ 
3    $y_{k+1} = (C + S)^{-1} (B^T x_{k+1} + Sy_k - g)$ 
4 end

```

将 NCSOR 算法写成矩阵的形式, 即

$$\begin{pmatrix} A + R & O \\ -B^T & C + S \end{pmatrix} \begin{pmatrix} x_{k+1} \\ y_{k+1} \end{pmatrix} = \begin{pmatrix} R & -B \\ O & S \end{pmatrix} \begin{pmatrix} x_k \\ y_k \end{pmatrix} + \begin{pmatrix} f \\ -g \end{pmatrix}. \quad (2.1)$$

注. 相较于第一节提到的三种算法, NCSOR 算法有以下特点:

1. NCSOR 算法中, 取 $R = \frac{1}{\eta}P - A$, $S = \frac{1}{\theta}Q - C$, 则 NCSOR 算法从形式上退化到 GPIU 算法, 但是 NCSOR 算法只要求矩阵 R, S 是任意给定的正定矩阵, 且 $S \succ \frac{B^T B}{2\tau}$, 其中 τ 是矩阵 A 的最小特征值, 从而解决 C 为半正定矩阵的广义鞍点问题的收敛性及线性收敛率, 而 GPIU 算法要求矩阵 P, Q 对称正定, Q 是奇异值分解矩阵 $B^T P^{-1} B + C$ 的近似矩阵, η, θ 是非零实数, $C = O$ 或 $C = \kappa Q$, 其中 κ 是个非零实常数的情况下给予收敛性证明. 不管从需解决的广义鞍点问题本身还是从收敛的条件来看这都是两类不同的算法.
2. GMHLSS 算法, 采用分裂预处理方法 (1.3) 将矩阵 C 转化成 D , 相应的将矩阵 B 转化成 (B_1, B_2) , 其迭代矩阵也与 B_1, B_2 有关, 异于我们提出的 NCSOR 算法, 且我们进一步对 NCSOR 算法进行收敛率分析, 其可以达到线性收敛率.
3. NCSOR 算法中, 矩阵 R, S 可以是任意给定的正定矩阵, 只要 $S \succ \frac{B^T B}{2\tau}$, 其中 τ 是矩阵 A 的最小特征值, 因此在算法实现过程中有很大的自由度. 在此较弱的条件下, 我们给出算法的收敛性及线性收敛率. 其中, 当 R, S 分别取 $\frac{1}{\omega}Q_1 - A$, $\frac{1-\sigma\alpha}{\sigma}Q_2 - C$ 时, NCSOR 算法退化到 NSOR 算法.
4. 从收敛性证明上看, NCSOR 算法只需矩阵 A 正定, C 半正定就得到收敛性, 并达到线性收敛率, 而 NSOR 算法只在矩阵 A, C 正定的条件下给出了收敛性证明. 从每步迭代计算量上看, NCSOR 算法与 NSOR 算法比较, 计算量并未增加. 在 NCSOR 算法中选取合适的 R, S , 使得 (2.1) 的迭代矩阵的条件数更好, 从而数值稳定性好.

3. 收敛性及收敛率分析

本节在假设矩阵 A 正定, C 半正定的条件下得到 NCSOR 算法的收敛性及线性收敛率. 为此, 令 $\{u_k\} = \{(x_k^T, y_k^T)^T\}$ 为由 NCSOR 算法产生的迭代序列.

引理 1. 迭代序列 $\{u_k\}$ 有如下性质:

$$\begin{pmatrix} R & -B \\ O & S \end{pmatrix} (u_k - u_{k+1}) = \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u_{k+1} + \begin{pmatrix} -f \\ g \end{pmatrix}. \quad (3.1)$$

证明. 将 (2.1) 经过整理可得 (3.1). □

引理 2. 迭代序列 $\{u_k\}$ 满足如下不等式:

$$(u_{k+1} - u)^T \begin{pmatrix} R & O \\ O & S \end{pmatrix} (u_k - u_{k+1}) \geq -\frac{1}{4\tau} \|B(y_{k+1} - y_k)\|^2 + (u_{k+1} - u)^T F(u), \quad \forall u \in \mathbb{R}^{m+n}, \quad (3.2)$$

其中 $F(u)$ 由 (1.2) 给出, τ 是 A 的最小特征值.

证明. 由 (3.1), 我们有

$$\begin{aligned} & (u_{k+1} - u)^T \begin{pmatrix} R & -B \\ O & S \end{pmatrix} (u_k - u_{k+1}) \\ &= (u_{k+1} - u)^T \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u_{k+1} + (u_{k+1} - u)^T \begin{pmatrix} -f \\ g \end{pmatrix} \\ &= (u_{k+1} - u)^T \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} (u_{k+1} - u) + (u_{k+1} - u)^T \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u + (u_{k+1} - u)^T \begin{pmatrix} -f \\ g \end{pmatrix} \\ &= (u_{k+1} - u)^T \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} (u_{k+1} - u) + (u_{k+1} - u)^T \left[\begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u + \begin{pmatrix} -f \\ g \end{pmatrix} \right] \\ &= (u_{k+1} - u)^T \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} (u_{k+1} - u) + (u_{k+1} - u)^T F(u) \\ &= (x_{k+1} - x)^T A (x_{k+1} - x) + (y_{k+1} - y)^T C (y_{k+1} - y) + (u_{k+1} - u)^T F(u). \end{aligned}$$

由于矩阵 A 正定, 则其最小特征值 $\tau > 0$, 且对于任意的 $x_k - x$, 有 $(x_k - x)^T A (x_k - x) \geq \tau \|x_k - x\|^2$. 由于矩阵 C 半正定, 则对任意的 $y_k - y$, 有 $(y_k - y)^T C (y_k - y) \geq 0$, 则上式可化简为

$$(u_{k+1} - u)^T \begin{pmatrix} R & -B \\ O & S \end{pmatrix} (u_k - u_{k+1}) \geq \tau \|x_{k+1} - x\|^2 + (u_{k+1} - u)^T F(u),$$

整理上式, 可得

$$\begin{aligned}
 & (u_{k+1} - u)^T \begin{pmatrix} R & O \\ O & S \end{pmatrix} (u_k - u_{k+1}) \\
 & \geq -(u_{k+1} - u)^T \begin{pmatrix} O & -B \\ O & O \end{pmatrix} (u_k - u_{k+1}) + \tau \|x_{k+1} - x\|^2 + (u_{k+1} - u)^T F(u) \\
 & = -(x_{k+1} - x)^T (-B) (y_k - y_{k+1}) + \tau \|x_{k+1} - x\|^2 + (u_{k+1} - u)^T F(u) \\
 & \geq -\frac{1}{2} \left(2\tau \|x_{k+1} - x\|^2 + \frac{1}{2\tau} \|B(y_k - y_{k+1})\|^2 \right) + \tau \|x_{k+1} - x\|^2 + (u_{k+1} - u)^T F(u) \\
 & = -\frac{1}{4\tau} \|B(y_{k+1} - y_k)\|^2 + (u_{k+1} - u)^T F(u),
 \end{aligned} \tag{3.3}$$

其中 (3.3) 中的第二个不等式是运用公式

$$a^T b \leq \frac{1}{2} (r \|a\|^2 + \frac{1}{r} \|b\|^2), \quad \forall a, b \in \mathbb{R}^n, \quad \forall r \in \mathbb{R}_{++}$$

得到. $\square$

定理 1. 由 NCSOR 算法产生的迭代序列 $\{u_k\}$ 具有 G 度量下的 Fejer 单调性, 即

$$\|u_{k+1} - u^*\|_D^2 \leq \|u_k - u^*\|_D^2 - \|u_k - u_{k+1}\|_G^2, \quad \forall u^* \in \Omega^*, \tag{3.4}$$

其中 $D = \begin{pmatrix} R & O \\ O & S \end{pmatrix}$, $G = \begin{pmatrix} R & O \\ O & S - \frac{B^T B}{2\tau} \end{pmatrix}$.

证明. 由 (3.2) 有

$$(u_{k+1} - u)^T D (u_k - u_{k+1}) \geq -\frac{1}{4\tau} \|B(y_{k+1} - y_k)\|^2 + (u_{k+1} - u)^T F(u), \quad \forall u \in \mathbb{R}^{m+n},$$

运用等式 $b^T D (b - a) = \frac{1}{2} (\|b\|_D^2 - \|a\|_D^2 + \|a - b\|_D^2)$, 令 $b = u_{k+1} - u$, $a = u_k - u$ 代入上式有

$$-\frac{1}{2} (\|u_{k+1} - u\|_D^2 - \|u_k - u\|_D^2 + \|u_{k+1} - u_k\|_D^2) \geq -\frac{1}{4\tau} \|B(y_{k+1} - y_k)\|^2 + (u_{k+1} - u)^T F(u),$$

整理上式, 可得

$$\begin{aligned}
 & \|u_{k+1} - u\|_D^2 \\
 & \leq \|u_k - u\|_D^2 - \|u_k - u_{k+1}\|_D^2 + \frac{1}{2\tau} \|B(y_{k+1} - y_k)\|^2 - 2(u_{k+1} - u)^T F(u) \\
 & = \|u_k - u\|_D^2 - \|x_k - x_{k+1}\|_R^2 - \|y_k - y_{k+1}\|_S^2 + \frac{1}{2\tau} \|B(y_{k+1} - y_k)\|^2 - 2(u_{k+1} - u)^T F(u) \tag{3.5} \\
 & = \|u_k - u\|_D^2 - \|x_k - x_{k+1}\|_R^2 - \|y_k - y_{k+1}\|_{S - \frac{B^T B}{2\tau}}^2 - 2(u_{k+1} - u)^T F(u) \\
 & = \|u_k - u\|_D^2 - \|u_k - u_{k+1}\|_G^2 - 2(u_{k+1} - u)^T F(u).
 \end{aligned}$$

在 (3.5) 中令 $u = u^* \in \Omega^*$, 利用 $(u_{k+1} - u^*)^T F(u^*) = 0$, 我们有 (3.4) 成立. $\square$

定理 2. 假设矩阵 R, S 正定, 使得 $S \succ \frac{B^T B}{2\tau}$, 序列 $\{u_k\}$ 由 NCSOR 算法产生, 则序列 $\{u_k\}$ 收敛到 $u^* \in \Omega^*$.

证明. 由 (3.4) 和矩阵 R, S 正定, 且满足 $S \succ \frac{B^T B}{2\tau}$ 知序列 $\{u_k\}$ 是有界数列, 且

$$\lim_{k \rightarrow \infty} \|u_k - u_{k+1}\| = 0.$$

令 u^∞ 为数列 $\{u_k\}$ 的一个聚点, 则存在一个收敛到 u^∞ 的子列 $\{u_{k_j}\}$. 由 (3.1) 可知 $u^\infty \in \Omega^*$, 即 u^∞ 是广义鞍点问题 (1.1) 的解. 在 (3.4) 中取 u^* 为 u^∞ , 则我们有

$$\|u_{k+1} - u^\infty\|_D^2 \leq \|u_k - u^\infty\|_D^2 - \|u_k - u_{k+1}\|_G^2,$$

这时 $\|u_{k+1} - u^\infty\|$ 单调下降有极限, 且 $\{u_k\}$ 存在子列收敛到 u^∞ , 因此序列 $\{u_k\}$ 收敛到 $u^\infty \in \Omega^*$. $\square$

为分析算法的线性收敛性, 我们先建立迭代点列如下的性质, 它将迭代点列函数值的残差与前后两点的距离联系起来.

引理 3. 假设序列 $\{u_k\}$ 由 NCSOR 算法产生, 矩阵 $G = \begin{pmatrix} R & O \\ O & S - \frac{B^T B}{2\tau} \end{pmatrix} \succ O$, 矩阵 B 列满秩, 则有

$$\|u_{k+1} - u_k\|_G^2 \geq C_0 \|F(u_{k+1})\|^2, \quad (3.6)$$

其中 $C_0 = \lambda_{\min}(G) / \max\{2\|R\|^2, 2\|B^T B\| + \|S\|^2\}$.

证明. 由引理 1 知:

$$F(u_{k+1}) = \begin{pmatrix} A & B \\ -B^T & C \end{pmatrix} u_{k+1} - \begin{pmatrix} f \\ -g \end{pmatrix} = \begin{pmatrix} R & -B \\ O & S \end{pmatrix} (u_k - u_{k+1}),$$

故有

$$\begin{aligned} \|F(u_{k+1})\|^2 &= \|R(x_k - x_{k+1}) - B(y_k - y_{k+1})\|^2 + \|S(y_k - y_{k+1})\|^2 \\ &\leq 2\|R(x_k - x_{k+1})\|^2 + 2\|B(y_k - y_{k+1})\|^2 + \|S(y_k - y_{k+1})\|^2 \\ &\leq 2\|R\|^2 \|x_k - x_{k+1}\|^2 + 2\|B^T B\| \|y_k - y_{k+1}\|^2 + \|S\|^2 \|y_k - y_{k+1}\|^2 \\ &\leq C \|u_k - u_{k+1}\|^2, \end{aligned} \quad (3.7)$$

其中 $C = \max\{2\|R\|^2, 2\|B^T B\| + \|S\|^2\} > 0$, 第一个不等式是利用公式 $(a + b)^2 \leq 2a^2 + 2b^2$ 得到. 由于矩阵 G 正定, 利用 $\lambda_{\min}(G)\|u\|^2 \leq \|u\|_G^2$ 与 (3.7) 可知

$$\|F(u_{k+1})\|^2 \leq \frac{C}{\lambda_{\min}(G)} \|u_k - u_{k+1}\|_G^2,$$

从而 (3.6) 成立. $\square$

由于 F 是线性函数, 我们有如下结论:

引理 4. ^[18] 对任意的 $M > 0$, 总存在 $\varsigma > 0$, 使得

$$\text{dist}(x, F^{-1}(0)) \leq \varsigma \|F(x)\|, \quad \forall \|x\| < M.$$

由引理 3 和引理 4, 我们得出算法的线性收敛性结果:

定理 3. 设序列 $\{u_k\}$ 由 NCSOR 算法产生, 在引理 3 的假设条件下, 有

$$\text{dist}_D^2(u_{k+1}, \Omega^*) \leq \frac{1}{1+C_1} \text{dist}_D^2(u_k, \Omega^*),$$

其中 $C_1 = C_0/(\varsigma^2\|D\|)$.

证明. 由定理 1 知

$$\text{dist}_D^2(u_{k+1}, \Omega^*) \leq \text{dist}_D^2(u_k, \Omega^*) - \|u_k - u_{k+1}\|_G^2.$$

因为 $\{u_k\}$ 有界, 利用引理 4 的结论与 (3.6), 有

$$\begin{aligned} & \text{dist}_D^2(u_{k+1}, \Omega^*) \\ & \leq \text{dist}_D^2(u_k, \Omega^*) - C_0 \|F(u_{k+1})\|^2 \\ & \leq \text{dist}_D^2(u_k, \Omega^*) - \frac{C_0}{\varsigma^2} \text{dist}_D^2(u_{k+1}, \Omega^*) \\ & \leq \text{dist}_D^2(u_k, \Omega^*) - \frac{C_0}{\varsigma^2\|D\|} \text{dist}_D^2(u_{k+1}, \Omega^*). \end{aligned}$$

从而有 $\text{dist}_D^2(u_{k+1}, \Omega^*) \leq \frac{1}{1+C_0/(\varsigma^2\|D\|)} \text{dist}_D^2(u_k, \Omega^*)$, 故而得证. $\square$

4. 数值例子

本章节中, 我们通过一个简单的例子说明我们的算法的有效性. 为了方便比较数值效果, 我们选用文献^[1, 2, 9]中的 Stokes 模型. 针对矩阵 C 正定和半正定的情况, 将 NCSOR 算法与 NSOR 算法, GPIU 算法进行比较.

Stokes 模型问题是求 $\mathbf{u}, z$ 满足

$$\begin{cases} -\nu\Delta\mathbf{u} + \nabla z = \tilde{f}, & x \in \Omega, \\ \nabla \cdot \mathbf{u} = \tilde{g}, & x \in \Omega, \\ \mathbf{u} = 0, & x \in \partial\Omega, \\ \int_{\Omega} z(x)dx = 0, \end{cases} \quad (4.1)$$

其中 $\Omega = (0, 1) \times (0, 1) \subset \mathbb{R}^n$, $\partial\Omega$ 是 Ω 的边界, ν 是粘度等级, Δ 是分支的 Laplace 算子, $\mathbf{u} = (u^T, v^T)^T$ 是表示速度的向量值函数, z 是表示压力的标量值函数. 采用迎风格式^[1, 2]离散 (4.1), 并令 $T = \frac{1}{h^2} \cdot \text{tridiag}(-1, 2, -1) \in \mathbb{R}^{p \times p}$, $F = \frac{1}{h} \cdot \text{tridiag}(-1, 1, 0) \in \mathbb{R}^{p \times p}$, 则得到如 (1.1) 式的问题, 其系数矩阵的子矩阵有如下形式

$$A = \begin{pmatrix} I \otimes T + T \otimes I & O \\ O & I \otimes T + T \otimes I \end{pmatrix} \in \mathbb{R}^{2p^2 \times 2p^2}, \quad B = \begin{pmatrix} I \otimes F \\ F \otimes I \end{pmatrix} \in \mathbb{R}^{2p^2 \times p^2},$$

其中 $\otimes$ 表示 Kronecker 积, $h = \frac{1}{p+1}$ 表示离散化的网格大小.

具体地, 我们令初始值 $(x_0^T, y_0^T)^T = 0$, 误差 $\varepsilon = 10^{-6}$, 算法最大迭代次数 $K_{\max} = 1000$, 并令

$$\text{ERR} = \frac{\|b - Au_k\|}{\|b - Au_0\|}.$$

我们用 Iter., CPU, ERR 分别来表示算法的迭代步数, 计算时间 (秒), 第 k 步迭代误差. 所有算例均在 Windows 10 系统下用 Matlab 8.3.0.532 (R2014a) 计算.

表 1, 表 2 与表 3 分别给出了 NCSOR 算法, NSOR 算法与 GPIU 算法中矩阵及其参数的选取, 其中表 2 中的参数取值来自 [14], 其中 $q = \frac{\sigma}{1-\alpha\sigma}$, 表 3 中的参数取值来自 [17], 我们选取 [17] 中数值结果最好的 η, θ 值来进行数值实验比较.

表 1 NCSOR 算法中的矩阵及其参数的选取

δ	S	R	ε
2	I	I	10^{-6}

表 2 NSOR 算法中的矩阵及其参数的选取

Q_1	Q_2	ρ	ω	q	ε
$\frac{A}{\rho}$	$B^T B$	2	0.3	0.9	10^{-6}

表 3 GPIU 算法中的矩阵及其参数的选取

P	Q	η	θ	ε
A	C	0.6	0.8	10^{-6}

下面, 我们分矩阵 C 正定与半正定两种情况来进行数值实验.

a. 矩阵 A, C 正定的情形

我们采用 [14] 中的参数取法, 令 $C = \delta B^T B$, $\delta = 2$, 由于矩阵 B 列满秩, 故矩阵 C 正定. 表 4 给出了在矩阵 A, C 正定情况下, NCSOR 算法, NSOR 算法与 GPIU 算法在 p 选取不同值时的数值结果比较. 其数据显示, 不管是从迭代步数 (Iter.) 还是计算时间 (CPU) 来看, 我们的 NCSOR 算法都优于 NSOR 算法. 当 $p \geq 10$ 时, NCSOR 算法的迭代误差 (ERR) 也明显优于 NSOR 算法与 GPIU 算法. 图 1 分别给出了 p 取不同值时, 算法的迭代误差 (ERR) 随迭代步数 (Iter.) 的变化曲线.

表 4 NCSOR 算法, NSOR 算法及 GPIU 算法的数值结果比较 (矩阵 C 正定)

	NCSOR			NSOR			GPIU		
	Iter.	CPU	ERR	Iter.	CPU	ERR	Iter.	CPU	ERR
p=5	5	0.000	8.7781e-007	62	0.045	8.2336e-007	15	0.003	5.6713e-007
p=10	5	0.001	5.6440e-007	61	0.334	8.8294e-007	15	0.074	6.7102e-007
p=20	5	0.027	2.7467e-007	61	9.827	8.6137e-007	15	0.308	7.2170e-007
p=30	5	0.131	1.6689e-007	61	92.536	8.6055e-007	15	2.042	7.3657e-007

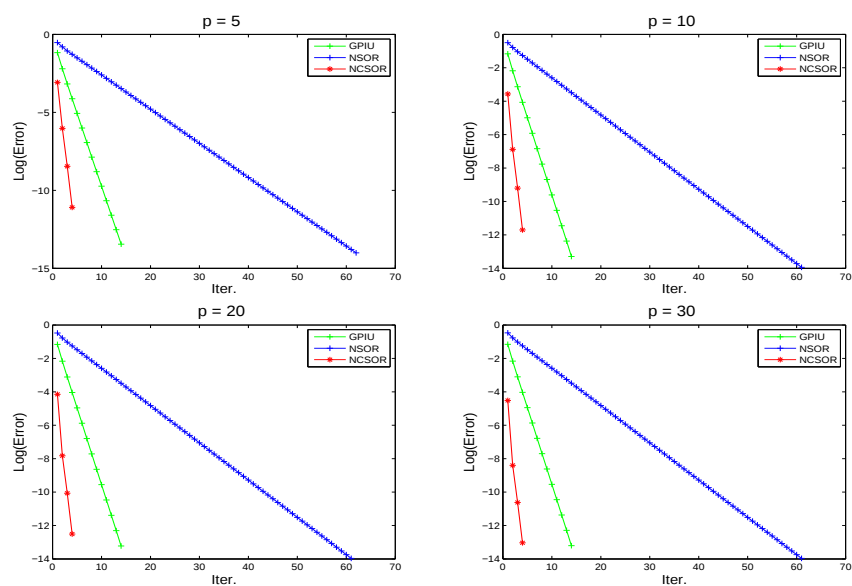


图 1 函数的迭代误差 (ERR) 随迭代步数 (Iter.) 的变化曲线 (矩阵 C 正定)

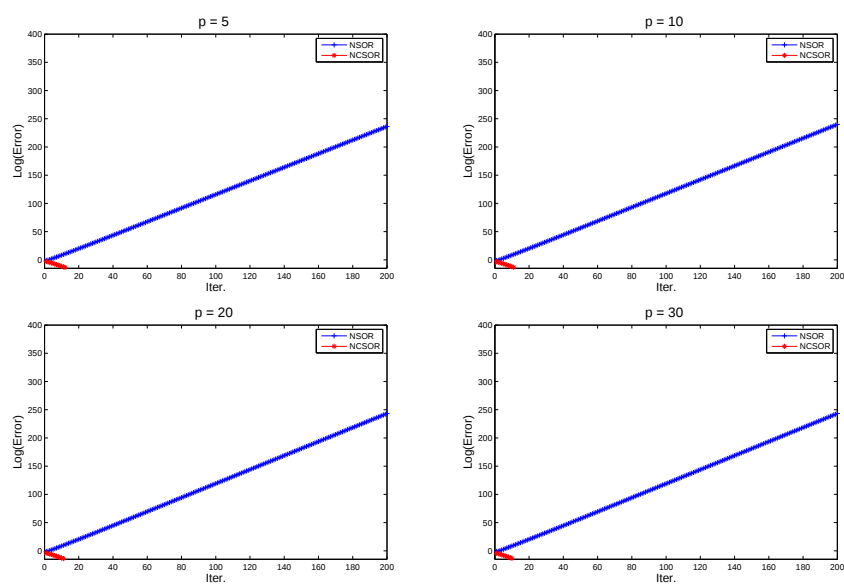


图 2 函数的迭代误差 (ERR) 随迭代步数 (Iter.) 的变化曲线 (矩阵 C 半正定)

b. 矩阵 A 正定, 矩阵 C 半正定的情形

首先, 我们构造半正定矩阵 C , 针对 a. 中的正定矩阵 C 的 p^2 个正的特征值, 随机选取其中的 $2 \times p$ 个, 将其设置为 0, 其余特征值不变, 故可构造出半正定矩阵 C . 表 5 给出了在矩阵 A 正定, C 半正定的情况下, NCSOR 算法与 NSOR 算法在问题的 p 选取不同值时的数值结果, 其数据显示, 我们的 NCSOR 算法可以计算当矩阵 C 半正定时的算例, 而 NSOR 算法却不能, 明显地看出我们的算法优于 NSOR 算法. 图 2 分别给出了 p

取不同值时, 算法的迭代误差 (ERR) 随迭代步数 (Iter.) 的变化曲线.

表 5 NCSOR 算法与 NSOR 算法的数值结果比较 (矩阵 C 半正定)

	NCSOR			NSOR		
	Iter.	CPU	ERR	Iter.	CPU	ERR
p=5	12	0.001	5.9185e-007	1000	0.110	NaN
p=10	12	0.003	4.7360e-007	1000	0.808	NaN
p=20	12	0.056	3.7468e-007	1000	55.884	NaN
p=30	11	0.274	5.8800e-007	1000	303.292	NaN

5. 结 论

本文给出了求解广义鞍点问题的改进的类 SOR 算法. 在主对角块 A 正定, C 半正定的条件下, 证明了算法的全局收敛性并分析了其线性收敛速率. 初步的数值结果表明, 新算法是有效的.

致谢. 作者感谢审稿人对本文初稿的评论和建议, 这些评论和建议对提升本文质量有极大的帮助.

参 考 文 献

- [1] Bai Z Z, Golub G H, Pan J Y. Preconditioned Hermitian and skew-Hermitian splitting methods for non-Hermitian positive semidefinite linear systems[J]. Numerische Mathematik, 2004, 98(1): 1–32.
- [2] Bai Z Z, Parlett B N, Wang Z Q. On generalized successive overrelaxation methods for augmented linear systems[J]. Numerische Mathematik, 2005, 102(1): 1–38.
- [3] Bai Z Z, Wang Z Q. On parameterized inexact Uzawa methods for generalized saddle point problems[J]. Linear Algebra and its Applications, 2008, 428(11–12): 2900–2932.
- [4] Bergamaschi L, Gondzio J, Zilli G. Preconditioning indefinite systems in interior point methods for optimization[J]. Computational Optimization and Applications, 2004, 28(2): 149–171.
- [5] Bergen A R. Power systems analysis[M]. Englewood Cliffs, NJ: Prentice-Hall, 1986.
- [6] Betts J T. Practical methods for optimal control using nonlinear programming[M]. Society for Industrial and Applied Mathematics, Philadelphia, 2001.
- [7] Björck Å. Numerical methods for least squares problems [M]. Society for Industrial and Applied Mathematics, Philadelphia, 1996.
- [8] Cao Y, Jiang M Q, Yao L Q. New choices of preconditioning matrices for generalized inexact parameterized iterative methods[J]. Journal of Computational and Applied Mathematics, 2010, 235(1): 263–269.
- [9] Elman H C. Preconditioning for the steady-state Navier-Stokes equations with low viscosity[J]. Siam Journal on Scientific Computing, 1999, 20: 1299–1316.
- [10] Golub G H, Van Loan C F. Matrix computations [M]. Johns Hopkins University Press, Baltimore, 1996.
- [11] Haber E, Ascher U M. Preconditioned all-at-once methods for large, sparse parameter estimation problems[J]. Inverse Problems, 2001, 17(6): 1847–1864.

- [12] Heinkenschloss M, Nguyen H. Neumann-neumann domain decomposition preconditioners for linear-quadratic elliptic optimal control problems[J]. Siam Journal on Scientific Computing, 2006, 28(3): 1001–1028.
- [13] Jiang M Q, Cao Y. On local Hermitian and skew-Hermitian splitting iteration methods for generalized saddle point problems[J]. Journal of Computational and Applied Mathematics, 2009, 231(2): 973–982.
- [14] 刘丽华, 马昌凤, 唐嘉. 求解广义鞍点问题的一个新的类 SOR 算法 [J]. 计算数学, 2016, 38(1): 83–95.
- [15] Martin B, Wolfram M. Numerical approximation of an SQP-type method for parameter identification[J]. Siam Journal on Numerical Analysis, 2003, 40(5): 1775–1797.
- [16] Strang G. Introduction to applied mathematics[M]. Wellesley-Cambridge Press, Wellesley, MA, 1986.
- [17] Zhang G F, Yang J L, Wang S S. On generalized parameterized inexact Uzawa method for a block two-by-two linear system[J]. Journal of Computational and Applied Mathematics, 2014, 255(285): 193–207.
- [18] Zheng X Y, Ng K F. Metric subregularity of piecewise linear multifunctions and applications to piecewise linear multiobjective optimization[J]. Siam Journal on Optimization, 2014, 24(1): 154–174.

AN IMPROVED SOR-TYPE ALGORITHM FOR SLOVING GENERALIZED SADDLE-POINT PROBLEMS

Zhang Chun

(School of Mathematical Sciences, Nanjing Normal University, Nanjing 210023, China;
Department of Basic Courses, The PLA Army Engineering University, Nanjing 211101, China)

Jia Zehui

(School of Mathematics and Statistics, Nanjing University of Information Science & Technology,
Nanjing 210044, China)

Cai Xingju

(School of Mathematical Sciences, Nanjing Normal University, Nanjing 210023, China)

Han Deren

(School of Mathematical Sciences, Beihang University, Beijing 100191, China)

Abstract

For the generalized saddle point problem, we develop an improved class of successive over relaxation algorithms. Under mild conditions, we prove its convergence and establish its linear rate of convergence. While, as the classical methods, it needs to solve some linear system of equations approximately to get the next iterate, the flexibility in choosing the involved matrices makes the subproblems easy or even to have closed form solutions, which leads the algorithm to be an efficient one. Preliminary numerical results show the effectiveness of the new method.

Keywords: saddle point problem; SOR algorithm; global convergence; convergence rate

2000 Mathematics Subject Classification: 65F10, 65H10