文章编号: 1000-5862(2012)02-0171-06

求矩阵方程 *AXB+CYD=E* 自反最佳逼近解的迭代算法

孙合明[1], 祁正萍[1], 杨家稳[2]

(1. 河海大学理学院, 江苏 南京 211100; 2. 滁州职业技术学院基础部, 安徽 滁州 239000)

摘要: 利用复合最速下降法的迭代算法对基于自反矩阵(或反自反矩阵)下广义 Sylvester 矩阵方程 **AXB+CYD=E** 最佳逼近解进行了研究, 证明了无论矩阵方程 **AXB+CYD=E** 是否相容, 该算法都可以用于计算其最佳逼近解. 最后, 通过 2 个数值实验证明了该算法的可行性.

关键词: Sylvester 矩阵方程; Kronecker 积; 复合最速下降法; 最佳逼近; 自反矩阵

中图分类号: O 24 文献标志码: A

0 引言

首先介绍本文中的符号. $\mathbf{R}^{m\times n}$ 表示 $m\times n$ 阶实矩阵的集合, $\boldsymbol{I}$ 表示单位矩阵, $\boldsymbol{A}^{\mathrm{T}}$ 表示矩阵 $\boldsymbol{A}$ 的转置. 对于矩阵 $\boldsymbol{A}$, $\boldsymbol{B}\in\mathbf{R}^{m\times n}$, $\boldsymbol{A}\otimes\boldsymbol{B}$ 表示 $\boldsymbol{A}$ 与 $\boldsymbol{B}$ 的 Kronecker 积, $\langle\boldsymbol{A},\boldsymbol{B}\rangle=\mathrm{trace}(\boldsymbol{B}^{\mathrm{T}}\boldsymbol{A})$ 表示 $\boldsymbol{A}$ 与 $\boldsymbol{B}$ 的内积. $\|\boldsymbol{A}\|$ 表示矩阵 $\boldsymbol{A}$ 的 Frobenius 范数, $\|\boldsymbol{A}\|^2=\langle\boldsymbol{A},\boldsymbol{A}\rangle$. 定义矩阵 $\boldsymbol{A}=(a_{ij})\in\mathbf{R}^{m\times n}$ 的拉伸算子 $vec(\cdot)$ 为 $vec(\boldsymbol{A})=[a_{11},a_{21},\cdots,a_{m1},a_{12},a_{22},\cdots,a_{m2},\cdots a_{1n},a_{2n},\cdots,a_{mn}]^{\mathrm{T}}$., H 表示实的 Hilbert 空间. 对于映射 $T:H\to H$, 所有关于 T 的不动点的集合表示为 $Fix(T):=\{x\in H\mid T(x)=x\}$. 若 $\boldsymbol{P}^2=\boldsymbol{I}$ 且 $\boldsymbol{P}^{\mathrm{T}}=\boldsymbol{P}$, 则称实矩阵 $\boldsymbol{P}$ 是反射矩阵. 若 $\boldsymbol{A}=\boldsymbol{PAP}$ (或 $\boldsymbol{A}=-\boldsymbol{PAP}$), 则称矩阵 $\boldsymbol{A}$ 为关于 $\boldsymbol{P}$ 的自反矩阵(或反自反矩阵). $R_r^{n\times n}(\boldsymbol{P})$ (或 $R_a^{n\times n}(\boldsymbol{P})$)为关于反射矩阵 $\boldsymbol{P}$ 的 n 阶自反矩阵空间(或反自反矩阵空间). 设 $K=\left\{\begin{pmatrix}vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\mid \boldsymbol{X}\in R_r^{p\times p}(\boldsymbol{P}),\ \boldsymbol{Y}\in R_r^{q\times q}(\boldsymbol{Q})\right\}$, 可证明 K 是凸集. P_K 表示到非空凸集 K 上的投影算子.

本文考虑如下问题:

问题Ⅰ 给定矩阵 $\boldsymbol{A}\in\mathbf{R}^{m\times p},\boldsymbol{B}\in\mathbf{R}^{p\times n},\boldsymbol{C}\in\mathbf{R}^{m\times q},\boldsymbol{D}\in\mathbf{R}^{q\times n},\boldsymbol{E}\in\mathbf{R}^{m\times n}$, 求 $\boldsymbol{X}\in R_r^{p\times p}(\boldsymbol{P})$ 和 $\boldsymbol{Y}\in R_r^{q\times q}(\boldsymbol{Q})$ 使得

$$\|\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}\|=\min.$$

问题Ⅱ 设问题Ⅰ的解集合为 S_E, 给定 $\boldsymbol{X}^*\in R_r^{p\times p}(\boldsymbol{P})$ 和 $\boldsymbol{Y}^*\in R_r^{q\times q}(\boldsymbol{Q})$, 求 $(\hat{\boldsymbol{X}},\hat{\boldsymbol{Y}})\in S_E$ 使得

$$\|\hat{\boldsymbol{X}}-\boldsymbol{X}^*\|+\|\hat{\boldsymbol{Y}}-\boldsymbol{Y}^*\|=\min_{(\boldsymbol{X},\boldsymbol{Y})\in S_E}\left(\|\boldsymbol{X}-\boldsymbol{X}^*\|+\|\boldsymbol{Y}-\boldsymbol{Y}^*\|\right).$$

对于矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$, 若它们是相容的, S_E 是解的集合; 若它们不相容, S_E 就是最小二乘解的集合. 所以无论矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 是否相容, 都可以认为 S_E 是问题Ⅰ的解集. 问题Ⅱ是在 S_E 上找和矩阵 $\boldsymbol{X}^*\in R_r^{p\times p}(\boldsymbol{P})$ 和 $\boldsymbol{Y}^*\in R_r^{q\times q}(\boldsymbol{Q})$ 最接近的矩阵. 由于数据不完整或者修改数据, 在线性系统的测试和复原过程中, 会提出问题Ⅱ, 未知矩阵的估计值 $\boldsymbol{X}^*$ 和 $\boldsymbol{Y}^*$ 可以通过实验观测和统计分布的信息获得.

文献[1-4]已经讨论了矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 的解或最小二乘解, 然而矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 的最佳逼近解的研究却不多. 若矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 是相容的, 下面的文献讨论了最佳逼近解. 文献[5]利用广义奇异值分解(generalized singular value decomposition, GSVD)给出了极小范数最小二乘解; 文献[6]为求极小范数最小二乘解提出一个算法并且推广了这个算法, 用于计算给定矩阵的最佳逼近解; 文献[7]利用共轭梯度的思想给出一个算法用于计算给定矩阵的中心对称最佳逼近解; 文献[8]给出了一个算法用于求解自反矩阵(或反自

收稿日期: 2011-12-20

基金项目: 安徽省高校省级自然科学基金(KJ2011B119)资助项目.

作者简介: 孙合明(1970-), 男, 山东诸城人, 副教授, 博士,主要从事智能计算和优化理论的研究.

反矩阵)最佳逼近解；文献[9]介绍了最佳逼近解的表示方法. 上述文献中，文献[5]仅求解了极小范数最小二乘解，而其它文献研究了任意给定矩阵的最佳逼近解.

关于不相容矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 最佳逼近解的研究成果较少见. 对于矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$，文献[10]阐述了极小范数对称最小二乘解；对于矩阵方程 $\boldsymbol{AXA}^{\mathrm{T}}+\boldsymbol{BYB}^{\mathrm{T}}=\boldsymbol{C}$，文献[11]给出了极小范数最小二乘解的表达式；对于矩阵方程 $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{XB}+\boldsymbol{B}^{\mathrm{T}}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{A}=\boldsymbol{D}$，文献[12]讨论了它的极小范数最小二乘解；对于任意给定矩阵，文献[13]提出一种算法求它的最佳逼近解. 在文献[5-6, 9, 12-13]中，未知矩阵都是一般矩阵，也就是说未知矩阵没有特殊的约束条件. 对于同一矩阵方程，无约束未知矩阵的最佳逼近问题要比有约束未知矩阵的最佳逼近问题容易些.

关于反射矩阵 $\boldsymbol{P}$ 的自反矩阵或反自反矩阵在系统与控制理论、工程、科学计算等领域中都有广泛应用[14-16]. 若矩阵方程 $\boldsymbol{A}^{H}\boldsymbol{XB}=\boldsymbol{C}$ 和 $\boldsymbol{AX}=\boldsymbol{B}$ 是相容的，文献[17-18]分别研究了自反(或反自反)最佳逼近解. 若矩阵方程 $\boldsymbol{A}_1\boldsymbol{XB}_1=\boldsymbol{C}_1$，$\boldsymbol{A}_2\boldsymbol{XB}_2=\boldsymbol{C}_2$ 是不相容的,文献[19]提出一种算法求自反矩阵(或反自反矩阵)的最佳逼近解.

I. Yamada 提出了复合最速下降法(HSDM)[20]. HSDM 已经应用于信号处理和正定 Toeplitz 矩阵集合上最佳逼近解的求解[21]. 本文借助于 HSDM 研究当矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 不相容时，求它的自反矩阵(或反自反矩阵)最佳逼近解.

1 迭代算法

首先给出解决问题 II 的迭代算法，然后证明该算法收敛，最后研究如何计算到非空凸集 K 上的投影.

算法 **1**:

步骤 **1** 输入矩阵 $\boldsymbol{A}\in\mathbf{R}^{m\times p},\boldsymbol{B}\in\mathbf{R}^{p\times n},\boldsymbol{C}\in\mathbf{R}^{m\times q},\boldsymbol{D}\in\mathbf{R}^{q\times n},\boldsymbol{E}\in\mathbf{R}^{m\times n},\boldsymbol{X}_0\in\mathbf{R}^{p\times p},\boldsymbol{Y}_0\in\mathbf{R}^{q\times q},\boldsymbol{X}^*\in R_r^{p\times p}(\boldsymbol{P})$ 和 $\boldsymbol{Y}^*\in R_r^{q\times q}(\boldsymbol{Q})$, k:=1;

步骤 **2** 计算 $\boldsymbol{M}_1=2(\boldsymbol{BB}^{\mathrm{T}})\otimes(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}),\boldsymbol{M}_2=2(\boldsymbol{BD}^{\mathrm{T}})\otimes(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{C}),\boldsymbol{N}_1=2(\boldsymbol{DB}^{\mathrm{T}})\otimes(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{A}),\boldsymbol{N}_2=2(\boldsymbol{DD}^{\mathrm{T}})\otimes(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{C}),\gamma=1\Big/\left\|\begin{pmatrix}\boldsymbol{M}_1 & \boldsymbol{M}_2\\ \boldsymbol{N}_1 & \boldsymbol{N}_2\end{pmatrix}\right\|$;

步骤 **3** 令

$$\boldsymbol{F}=\left(\boldsymbol{BB}^{\mathrm{T}}\right)\otimes\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}\right)vec\left(\boldsymbol{X}_{k-1}\right)+\left(\boldsymbol{BD}^{\mathrm{T}}\right)\otimes\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{C}\right)vec\left(\boldsymbol{Y}_{k-1}\right)-\left(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm{T}}\right)vec\left(\boldsymbol{E}\right),$$

$$\boldsymbol{G}=\left(\boldsymbol{DB}^{T}\right)\otimes\left(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{A}\right)vec\left(\boldsymbol{X}_{k-1}\right)+\left(\boldsymbol{DD}^{\mathrm{T}}\right)\otimes\left(\boldsymbol{C}^{\mathrm{T}}\boldsymbol{C}\right)vec\left(\boldsymbol{Y}_{k-1}\right)-\left(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm{T}}\right)vec\left(\boldsymbol{E}\right),$$

计算

$$\nabla\psi\left(\begin{pmatrix}vec\left(\boldsymbol{X}_{k-1}\right)\\ vec\left(\boldsymbol{Y}_{k-1}\right)\end{pmatrix}\right)=2\begin{pmatrix}\boldsymbol{F}\\ \boldsymbol{G}\end{pmatrix},$$

$$T\begin{pmatrix}vec(\boldsymbol{X}_{k-1})\\ vec(\boldsymbol{Y}_{k-1})\end{pmatrix}=P_K\left\{\begin{pmatrix}vec(\boldsymbol{X}_{k-1})\\ vec(\boldsymbol{Y}_{k-1})\end{pmatrix}-\frac{1}{\gamma}\nabla\psi(\boldsymbol{X}_{k-1},\boldsymbol{Y}_{k-1})\right\},$$

$$\begin{pmatrix}vec(\boldsymbol{X}_{k})\\ vec(\boldsymbol{Y}_{k})\end{pmatrix}=\left(1-\frac{2}{k}\right)T\begin{pmatrix}vec(\boldsymbol{X}_{k-1})\\ vec(\boldsymbol{Y}_{k-1})\end{pmatrix}+\frac{2}{k}\begin{pmatrix}vec(\boldsymbol{X}^*)\\ vec(\boldsymbol{Y}^*)\end{pmatrix};$$

步骤 **4** 若 $\|\boldsymbol{X}_k-\boldsymbol{X}_{k-1}\|+\|\boldsymbol{Y}_k-\boldsymbol{Y}_{k-1}\|<\varepsilon$，则停止；否则，置 $k:=k+1$，转向步骤 3.

为了证明该算法的收敛性，首先给出如下的定义和定理.

定义 **1** 若 $\forall x,y\in S\subset H$，$\exists\ \kappa>0$，使 $\|T(x)-T(y)\|\leqslant\kappa\|x-y\|$，则称映射 $T:H\to H$ 为 κ-Lipschitzian (或 κ-Lipschitzian 连续)；特别地，若对于一切 $x,y\in H$，有 $\|T(x)-T(y)\|\leqslant\|x-y\|$，则称映射 $T:H\to H$ 为非扩展映射.

定义 **2** 对于给定的集合 $S\subset H$，若 $\forall x,y\in S$，$\exists\ \eta>0$，使 $\langle F(x)-F(y),x-y\rangle\geqslant\eta\|x-y\|^2$，则称映射 $F:H\to H$ 在集合 S 上 η 强单调.

定理 **1**(hybrid steepest descent method) 设 $T:H\to H$ 是非扩展映射且 $Fix(T)\neq\varnothing$，函数 $\theta:H\to\mathbf{R}\cup\infty$ 存在 Gâteaux 导数 $\theta':H\to H$，且 θ' 在 $T(H)$ 上 κ-Lipschitzian 且 η 强单调，$\forall u_0\in H$ 和正实数序列 $(\lambda_n)_{n\geqslant1}$ 满足:

(L1) $\lim\limits_{n\to\infty}\lambda_n=0$;

(L2) $\sum\limits_{n\geqslant1}\lambda_n=\infty$;

(L3) $\lim\limits_{n\to\infty}(\lambda_n-\lambda_{n+1})\lambda_{n+1}^{-2}=0$,

或者

(W1) $\lim\limits_{n\to\infty}\lambda_n=0$;

(W2) $\sum\limits_{n\geqslant1}\lambda_n=\infty$;

(W3) $\sum\limits_{n\geqslant1}|\lambda_n-\lambda_{n+1}|<\infty$,

则由 $u_{n+1}:=T(u_n)-\lambda_{n+1}\theta'(T(u_n))(n\geqslant0)$ 生成的序列 $(u_n)_{n\geqslant0}$ 强收敛于唯一的 u^*，其中 u^* 满足 $\theta(u^*)=$

$\inf\{\theta(Fix(T))\}$.

注 1　$\lambda_n = 1/n$ 是满足(W_1)~(W_3)的序列 $(\lambda_n)_{n\geqslant 1}$.

定理 2[22]　设 $K\subset H$ 是闭凸子集，假设

(i) $\psi: H\to \mathbf{R}\cup\{\infty\}$ 是 Gâteaux 可微的凸函数且它的导数 ψ'：$H\to H$ 在 H 上满足 γ-Lipschitzian；

(ii) $\theta: H\to \mathbf{R}\cup\{\infty\}$ 是 Gâteaux 可微的凸函数且它的导数 θ'：$H\to H$ 在 $T(H)$ 上满足 κ-Lipschitzian 和 η 强单调，其中 $T:=P_k(I-v\psi')$，$v\in(0,2/\gamma]$，则 $\forall u_0\in H$，由 $u_n:=T(u_{n-1})-\lambda_n\theta'(T(u_{n-1}))$ 产生的序列 $(u_n)_{n\geqslant 0}$ 强收敛于唯一的 $x^*=\arg\inf\limits_{x\in K_\psi}\theta(x)$，其中 $K_\psi:=\arg\inf\limits_{x\in K}\psi(x)\neq\varnothing$.

现在对于问题 II，在集合 K 上定义函数 ψ 和 θ，并证明它们满足定理 2 的条件.

定理 3　设 $\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\|\boldsymbol{AXB}+\boldsymbol{CYD}-\boldsymbol{E}\|^2$,则

$$\nabla\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=2\begin{pmatrix}\boldsymbol{F}\\ \boldsymbol{G}\end{pmatrix},\tag{1}$$

其中 $\boldsymbol{F}=(\boldsymbol{BB}^{\mathrm T})\otimes(\boldsymbol{A}^{\mathrm T}\boldsymbol{A})vec(\boldsymbol{X})+(\boldsymbol{BD}^{\mathrm T})\otimes(\boldsymbol{A}^{\mathrm T}\boldsymbol{C})vec(\boldsymbol{Y})-(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm T})vec(\boldsymbol{E}),\boldsymbol{G}=(\boldsymbol{DB}^{T})\otimes(\boldsymbol{C}^{\mathrm T}\boldsymbol{A})vec(\boldsymbol{X})+(\boldsymbol{DD}^{\mathrm T})\otimes(\boldsymbol{C}^{\mathrm T}\boldsymbol{C})vec(\boldsymbol{Y})-(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm T})vec(\boldsymbol{E})$.

证　$\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\|\boldsymbol{AXB}+\boldsymbol{CYD}-\boldsymbol{E}\|^2=\langle\boldsymbol{AXB}+\boldsymbol{CYD}-\boldsymbol{E},\boldsymbol{AXB}+\boldsymbol{CYD}-\boldsymbol{E}\rangle=\langle\boldsymbol{AXB},\boldsymbol{AXB}\rangle+\langle\boldsymbol{CYD},\boldsymbol{CYD}\rangle+2\langle\boldsymbol{AXB},\boldsymbol{CYD}\rangle-2\langle\boldsymbol{AXB},\boldsymbol{E}\rangle-2\langle\boldsymbol{CYD},\boldsymbol{E}\rangle+\langle\boldsymbol{E},\boldsymbol{E}\rangle=\langle vec(\boldsymbol{AXB}),vec(\boldsymbol{AXB})\rangle+\langle vec(\boldsymbol{CYD}),vec(\boldsymbol{CYD})\rangle+2\langle vec(\boldsymbol{AXB}),vec(\boldsymbol{CYD})\rangle-2\langle vec(\boldsymbol{AXB}),vec(\boldsymbol{E})\rangle-2\langle vec(\boldsymbol{CYD}),vec(\boldsymbol{E})\rangle+\langle vec(\boldsymbol{E}),vec(\boldsymbol{E})\rangle=\langle\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}),\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X})\rangle+\langle\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y}),\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y})\rangle+2\langle\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}),\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y})\rangle-2\langle\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}),vec(\boldsymbol{E})\rangle-2\langle\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y}),vec(\boldsymbol{E})\rangle+\langle vec(\boldsymbol{E}),vec(\boldsymbol{E})\rangle=(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})vec(\boldsymbol{X})+(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})vec(\boldsymbol{Y})+2(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})vec(\boldsymbol{Y})-2(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A}vec(\boldsymbol{X}))^{\mathrm T}vec(\boldsymbol{E})-2(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C}vec(\boldsymbol{Y}))^{\mathrm T}vec(\boldsymbol{E})+(vec(\boldsymbol{E}))^{\mathrm T}vec(\boldsymbol{E})=(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})^{\mathrm T}(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})vec(\boldsymbol{X})+(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})(vec(\boldsymbol{Y})+2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})vec(\boldsymbol{Y})-2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})^{\mathrm T}vec(\boldsymbol{E})-\left[2(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})^{\mathrm T}-(vec(\boldsymbol{E}))^{\mathrm T}\right]vec(\boldsymbol{E})=(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm T})(\boldsymbol{B}^{\mathrm T}\otimes\boldsymbol{A})vec(\boldsymbol{X})+(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm T})(\boldsymbol{D}^{\mathrm T}\otimes\boldsymbol{C})vec(\boldsymbol{Y})+2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm T})(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm T})vec(\boldsymbol{Y})-2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm T})vec(\boldsymbol{E})-2(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm T})vec(\boldsymbol{E})+(vec(\boldsymbol{E}))^{\mathrm T}vec(\boldsymbol{E})=(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{BB}^{\mathrm T})\otimes(\boldsymbol{A}^{\mathrm T}\boldsymbol{A})vec(\boldsymbol{X})+(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{DD}^{\mathrm T})\otimes(\boldsymbol{C}^{\mathrm T}\boldsymbol{C})vec(\boldsymbol{Y})+(vec(\boldsymbol{E}))^{\mathrm T}vec(\boldsymbol{E})+2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{BD}^{\mathrm T})\otimes(\boldsymbol{A}^{\mathrm T}\boldsymbol{C})vec(\boldsymbol{Y})-2(vec(\boldsymbol{X}))^{\mathrm T}(\boldsymbol{B}\otimes\boldsymbol{A}^{\mathrm T})vec(\boldsymbol{E})-2(vec(\boldsymbol{Y}))^{\mathrm T}(\boldsymbol{D}\otimes\boldsymbol{C}^{\mathrm T})vec(\boldsymbol{E})$，则定理 3 得证.

定理 4　设 $\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\|\boldsymbol{X}-\boldsymbol{X}^*\|^2+\|\boldsymbol{Y}-\boldsymbol{Y}^*\|^2$，则

$$\nabla\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=2\begin{pmatrix} vec(\boldsymbol{X}-\boldsymbol{X}^*)\\ vec(\boldsymbol{Y}-\boldsymbol{Y}^*)\end{pmatrix}.\tag{2}$$

证　$\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\|\boldsymbol{X}-\boldsymbol{X}^*\|^2+\|\boldsymbol{Y}-\boldsymbol{Y}^*\|^2=\langle\boldsymbol{X}-\boldsymbol{X}^*,\boldsymbol{X}-\boldsymbol{X}^*\rangle+\langle\boldsymbol{Y}-\boldsymbol{Y}^*,\boldsymbol{Y}-\boldsymbol{Y}^*\rangle=\langle vec(\boldsymbol{X}-\boldsymbol{X}^*),vec(\boldsymbol{X}-\boldsymbol{X}^*)\rangle+\langle vec(\boldsymbol{Y}-\boldsymbol{Y}^*),vec(\boldsymbol{Y}-\boldsymbol{Y}^*)\rangle=(vec(\boldsymbol{X}-\boldsymbol{X}^*))^{\mathrm T}vec(\boldsymbol{X}-\boldsymbol{X}^*)+(vec(\boldsymbol{Y}-\boldsymbol{Y}^*))^{\mathrm T}vec(\boldsymbol{Y}-\boldsymbol{Y}^*)$，则定理 4 得证.

易证 $\psi'\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\nabla\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 和 $\theta'\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)=\nabla\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$，因此，分别用 $\nabla\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 和 $\nabla\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 来代替 $\psi'\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 和 $\theta'\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$.

定理 5　设 $\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 和 $\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 如定理 3 和定理 4 中所示，则

(i) $\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 是凸函数；

(ii) $\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 是凸函数；

(iii) $\nabla\psi\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 是 γ-Lipschitzian；

(iv) $\nabla\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 是 κ-Lipschitzian；

(v) $\nabla\theta\left(\begin{pmatrix} vec(\boldsymbol{X})\\ vec(\boldsymbol{Y})\end{pmatrix}\right)$ 是 η 强单调的.

证 由凸函数的定义可以证明(i), (ii)成立.

(iii)令

$M_1 = 2(BB^{\mathrm{T}})\otimes(A^{\mathrm{T}}A)$, $M_2 = 2(BD^{\mathrm{T}})\otimes(A^{\mathrm{T}}C)$,

$N_1 = 2(DB^{\mathrm{T}})\otimes(C^{\mathrm{T}}A)$, $N_2 = 2(DD^{\mathrm{T}})\otimes(C^{\mathrm{T}}C)$,

根据(1)式知,

$$\left\|\nabla\psi\left(\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}\right)-\nabla\psi\left(\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right)\right\|=$$

$$2\left\|\begin{pmatrix}F_1\\G_1\end{pmatrix}-\begin{pmatrix}F_2\\G_2\end{pmatrix}\right\|=\left\|\begin{pmatrix}M_1 & M_2\\N_1 & N_2\end{pmatrix}\begin{pmatrix}vec(X_1)-vec(X_2)\\vec(Y_1)-vec(Y_2)\end{pmatrix}\right\|\leqslant$$

$$\left\|\begin{pmatrix}M_1 & M_2\\N_1 & N_2\end{pmatrix}\right\|\left\|\begin{pmatrix}vec(X_1)-vec(X_2)\\vec(Y_1)-vec(Y_2)\end{pmatrix}\right\|=$$

$$\left\|\begin{pmatrix}M_1 & M_2\\N_1 & N_2\end{pmatrix}\right\|\left\|\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}-\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right\|,$$

其中 $F_1=(BB^{\mathrm{T}})\otimes(A^{\mathrm{T}}A)vec(X_1)+(BD^{\mathrm{T}})\otimes(A^{\mathrm{T}}C)vec(Y_1)-(B\otimes A^{\mathrm{T}})vec(E), G_1=(DB^{\mathrm{T}})\otimes(C^{\mathrm{T}}A)vec(X_1)+(DD^{\mathrm{T}})\otimes(C^{\mathrm{T}}C)vec(Y_1)-(D\otimes C^{\mathrm{T}})vec(E), F_2=(BB^{\mathrm{T}})\otimes(A^{\mathrm{T}}A)vec(X_2)+(BD^{\mathrm{T}})\otimes(A^{\mathrm{T}}C)vec(Y_2)-(B\otimes A^{\mathrm{T}})vec(E), G_2=(DB^{\mathrm{T}})\otimes(C^{\mathrm{T}}A)vec(X_2)+(DD^{\mathrm{T}})\otimes(C^{\mathrm{T}}C)vec(Y_2)-(D\otimes C^{\mathrm{T}})vec(E)$,

因此 $\nabla\psi\left(\begin{pmatrix}vec(X)\\vec(Y)\end{pmatrix}\right)$ 是 γ-Lipschitizian，其中

$$\gamma=\left\|\begin{pmatrix}M_1 & M_2\\N_1 & N_2\end{pmatrix}\right\|.$$

(iv)由(2)式有

$$\left\|\nabla\theta\left(\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}\right)-\nabla\theta\left(\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right)\right\|=$$

$$\left\|2\begin{pmatrix}vec(X_1-X^*)\\vec(Y_1-Y^*)\end{pmatrix}-2\begin{pmatrix}vec(X_2-X^*)\\vec(Y_2-Y^*)\end{pmatrix}\right\|=$$

$$2\left\|\begin{pmatrix}vec(X_1-X_2)\\vec(Y_1-Y_2)\end{pmatrix}\right\|=2\left\|\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}-\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right\|,$$

因此 $\nabla\theta\left(\begin{pmatrix}vec(X)\\vec(Y)\end{pmatrix}\right)$ 是 κ-Lipschitizian, 其中 $\kappa=2$.

(v)再由(2)式可得

$$\left\langle\nabla\theta\left(\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}\right)-\nabla\theta\left(\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right),\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}-\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right\rangle=\left\langle 2vec\begin{pmatrix}X_1-X_2\\Y_1-Y_2\end{pmatrix},vec\begin{pmatrix}X_1-X_2\\Y_1-Y_2\end{pmatrix}\right\rangle=$$

$$2\left\|vec\begin{pmatrix}X_1-X_2\\Y_1-Y_2\end{pmatrix}\right\|^2=2\left\|\begin{pmatrix}vec(X_1)\\vec(Y_1)\end{pmatrix}-\begin{pmatrix}vec(X_2)\\vec(Y_2)\end{pmatrix}\right\|^2,$$

由定义 2 知, $\nabla\theta\left(\begin{pmatrix}vec(X)\\vec(Y)\end{pmatrix}\right)$ 是 η 强单调的, 其中 $\eta=2$.

从上述证明中可以发现 $\psi(X,Y)$ 和 $\theta(X,Y)$ 满足定理 2 的条件, 由定理 2 可以推出算法 1 是收敛的.

在算法 1 中需要计算到凸集 K 上的投影, 有如下定理.

定理 6 $\forall X_0\in\mathbf{R}^{p\times p},\forall Y_0\in\mathbf{R}^{q\times q}$，则 $\begin{pmatrix}vec(X_0)\\vec(Y_0)\end{pmatrix}$ 到 K 上的投影计算公式为

$$P_K\begin{pmatrix}vec(X_0)\\vec(Y_0)\end{pmatrix}=\left\langle\begin{pmatrix}vec(X_0)\\vec(Y_0)\end{pmatrix},\alpha_1\right\rangle\alpha_1+\left\langle\begin{pmatrix}vec(X_0)\\vec(Y_0)\end{pmatrix},\alpha_2\right\rangle\alpha_2+\cdots+\left\langle\begin{pmatrix}vec(X_0)\\vec(Y_0)\end{pmatrix},\alpha_s\right\rangle\alpha_s,\quad(3)$$

其中 $\alpha_1,\alpha_2,\cdots,\alpha_s$ 是矩阵 $\begin{pmatrix}P^{\mathrm{T}}\otimes P-I & 0\\0 & Q^{\mathrm{T}}\otimes Q-I\end{pmatrix}$ 零空间的标准正交基.

证 由自反矩阵的定义知,

$$\begin{cases}X=PXP,\\Y=QYQ.\end{cases}\quad(4)$$

对(4)式两边运用拉伸算子 $vec(\cdot)$，有

$$\begin{cases}vec(X)=vec(PXP)=P^{\mathrm{T}}\otimes Pvec(X),\\vec(Y)=vec(QYQ)=Q^{\mathrm{T}}\otimes Qvec(Y).\end{cases}\quad(5)$$

(5)式也可以写成

$$\begin{cases}(P^{\mathrm{T}}\otimes P-I)vec(X)=\mathbf{0},\\(Q^{\mathrm{T}}\otimes Q-I)vec(Y)=\mathbf{0},\end{cases}$$

因此有

$$\begin{pmatrix}P^{\mathrm{T}}\otimes P-I & \mathbf{0}\\\mathbf{0} & Q^{\mathrm{T}}\otimes Q-I\end{pmatrix}\begin{pmatrix}vec(X)\\vec(Y)\end{pmatrix}=\mathbf{0}.\quad(6)$$

K 中元素就是线性系统方程(6)的解. 因此只要给定(6)式解集合的标准正交基, 就可以利用(3)式计算 K 上的投影.

对于问题 I 和问题 II, 若规定矩阵 X, Y 是反自反矩阵集合, 同样可以得到反自反最佳逼近解.

2 数值试验

接下来给出 2 个数值例子来说明上述结论, 所有的测试都是在 MATLAB2007R 中进行的.

例 1 考虑矩阵方程 $AXB+CYD=E$，其中

$$A=\begin{pmatrix}1&1&1\\1&1&2\\-1&-1&3\end{pmatrix},\quad B=\begin{pmatrix}1&1&3\\2&-4&-3\\1&2&1\end{pmatrix},$$

$$C=\begin{pmatrix}2&2&-1\\2&-1&-2\\1&-2&1\end{pmatrix},\quad D=\begin{pmatrix}-2&-1&-1\\2&1&1\\3&1&4\end{pmatrix},$$

$$\boldsymbol{E}=\begin{pmatrix}-4&5&7\\2&13&9\\10&16&2\end{pmatrix},\quad \boldsymbol{P}=\boldsymbol{Q}=\begin{pmatrix}1&0&0\\0&1&0\\0&0&-1\end{pmatrix},$$

$$\boldsymbol{X}^*=\begin{pmatrix}2&0&0\\2&0&0\\0&0&2\end{pmatrix},\quad \boldsymbol{Y}^*=\begin{pmatrix}0.5&-0.5&0\\0.5&-0.5&0\\0&0&0.5\end{pmatrix},$$

$$\boldsymbol{Y}_0=\boldsymbol{X}_0=\begin{pmatrix}10&10&10\\10&10&10\\10&10&10\end{pmatrix}.$$

该方程有多个自反解. 令 $\varepsilon=1.0\times 10^{-9}$, 利用算法 1, 可得

$$\boldsymbol{X}_{211\,729}=\begin{pmatrix}1.999\,998\,052\,875\,936&-0.000\,000\,374\,425\,795&0\\1.999\,998\,052\,875\,936&-0.000\,000\,374\,425\,795&0\\0&0&2.999\,998\,671\,469\,364\end{pmatrix},$$

$$\boldsymbol{Y}_{211\,729}=\begin{pmatrix}0.499\,997\,153\,848\,615&-0.500\,002\,024\,930\,655&0\\0.499\,997\,638\,690\,795&-0.500\,002\,509\,772\,626&0\\0&0&0.999\,998\,567\,367\,707\end{pmatrix},$$

相应的余项为

$$R_{211\,729}=\left\|\boldsymbol{A}\boldsymbol{X}_{211\,729}\boldsymbol{B}+\boldsymbol{C}\boldsymbol{Y}_{211\,729}\boldsymbol{D}-\boldsymbol{E}\right\|=9.528\,48\times10^{-6},$$

其中下标是迭代步数. 在计算误差之内, 可以认为解满足矩阵方程, 另外可以证明解是自反矩阵.

例 2　仍然考虑矩阵方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$, $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$, $\boldsymbol{X}_0$, $\boldsymbol{Y}_0$, $\boldsymbol{X}^*$和 $\boldsymbol{Y}^*$同例 1 中的一样. 令

$$\boldsymbol{P}=\boldsymbol{Q}=\begin{pmatrix}1&0&0\\0&0&-1\\0&-1&0\end{pmatrix},\quad \boldsymbol{E}=\begin{pmatrix}-14&5&7\\2&13&9\\10&16&22\end{pmatrix},$$

可以证明上述矩阵方程是不相容的.

令 $\varepsilon=1.0\times10^{-9}$, 得

$$\boldsymbol{X}_{5\,676\,196}=\begin{pmatrix}-8.795\,112\,231\,442\,715&-4.456\,447\,081\,495\,659&4.456\,447\,081\,495\,659\,4\\-2.687\,357\,408\,923\,304&2.965\,530\,890\,840\,564&0.232\,157\,278\,421\,583\\2.687\,357\,761\,271\,955&0.232\,157\,278\,421\,583&2.965\,531\,243\,189\,215\end{pmatrix},$$

$$\boldsymbol{Y}_{5\,676\,196}=\begin{pmatrix}4.409\,244\,781\,880\,800&-0.023\,118\,647\,531\,487&0.023\,118\,559\,444\,309\\-0.303\,163\,961\,628\,089&-4.428\,330\,726\,955\,296&2.898\,493\,297\,352\,236\\0.303\,164\,049\,715\,267&2.898\,493\,297\,352\,236&-4.428\,330\,550\,780\,939\end{pmatrix},$$

相应的余项为

$$R_{5\,676\,196}=\left\|\boldsymbol{A}\boldsymbol{X}_{5\,676\,196}\boldsymbol{B}+\boldsymbol{C}\boldsymbol{Y}_{5\,676\,196}\boldsymbol{D}-\boldsymbol{E}\right\|=4.433\,944\,972\,729\,042,$$

可以验证 $\boldsymbol{X}_{5\,676\,196}\in R_r^{3\times3}(\boldsymbol{P})$, $\boldsymbol{Y}_{5\,676\,196}\in R_r^{3\times3}(\boldsymbol{Q})$.

3　结论

本文根据HSDM, 提出求解方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 自反最佳逼近解的迭代算法. 无论 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 是否相容, 所给的算法都可以计算其最佳逼近解. 数值试验充分说明了算法的可行性, 但算法要求自反矩阵(反自反矩阵)的集合是凸集, 而且存在收敛速度慢的缺点. 若其它矩阵方程的未知约束矩阵是凸集, 则可以应用 HSDM 解决其最佳逼近问题. 针对广义自反矩阵或 $(\boldsymbol{P},\boldsymbol{Q})$ 广义自反矩阵的方程最佳逼近解已有一些研究成果, 如何应用 HSDM 求解广义自反矩阵或 $(\boldsymbol{P},\boldsymbol{Q})$ 广义自反矩阵下方程 $\boldsymbol{AXB}+\boldsymbol{CYD}=\boldsymbol{E}$ 最佳逼近问题是今后的研究工作.

4　参考文献

[1] Yasuda K, Skelton R E. Assigning controllability and observability gramians in feedback control [J]. Journal Guidance Control and Dynamics, 1991, 14(5): 878-885.

[2] Fujioka H, Hara S. State covariance assignment problem with measurement noise: a unified approach based on a symmetric matrix equation [J]. Linear Algebra and its Applications, 1994, 203/204: 579-605.

[3] Xu Guiping, Wei Musheng, Zheng Daosheng. On solutions of matrix equation **AXB+CYD=F** [J]. Linear Algebra and its Applications, 1998, 279(1/2/3): 93-109.

[4] Baksalary J K, Kaka R. The matrix equation **AXB+CYD=E** [J]. Linear Algebra and its Applications, 1980, 30: 141-147.

[5] 袁永新. 矩阵方程的最小二乘解 [J]. 高等学校计算数学学报,

2001, 23(4): 324-329.

[6] Peng Zhenyun, Peng Yaxin. An efficient iterative method for solving the matrix equation ***AXB+CYD=E*** [J]. Numerical Linear Algebra with Applications, 2006, 13(6): 473-485.

[7] 刘大瑾, 周海林, 袁东锦. ***AXB+CYD=F*** 的中心对称解及其最佳逼近解的迭代算法 [J]. 扬州大学学报: 自然科学版, 2008, 11(3): 9-13.

[8] Dehghan Mehdi, Hajarian Masoud. Finite iterative algorithms for the reflexive and anti-reflexive solutions of the matrix equation $A_1X_1B_1 + A_2X_2B_2 = C$ [J]. Mathematical and Computer Modelling, 2009, 49(9/10): 1937-1959.

[9] 彭振赟. 矩阵方程 ***AXC+BYD=E*** 的解及其最佳逼近 [J]. 数学理论与应用, 2002, 22(2): 99-103.

[10] 袁仕芳, 廖安平, 雷渊. 矩阵方程 ***AXB+CYD=E*** 的对称极小范数最小二乘解 [J]. 计算数学, 2007, 29(2): 203-216.

[11] 廖安平, 白中治. 矩阵方程 $AXA^{\mathrm{T}} + BYB^{\mathrm{T}} = C$ 的对称与反对称最小范数最小二乘解 [J]. 计算数学, 2005, 27(1): 81-95.

[12] 袁永新, 戴华. 矩阵方程 $A^{\mathrm{T}}XB + B^{\mathrm{T}}X^{\mathrm{T}}A = D$ 的极小范数最小二乘解 [J]. 高等学校计算数学学报, 2005, 27(3): 232-238.

[13] 盛兴平, 苏友峰, 陈果良. 矩阵方程 $A^{\mathrm{T}}XB + B^{\mathrm{T}}X^{\mathrm{T}}A = D$ 的极小范数最小二乘解的迭代解法 [J]. 高等学校计算数学学报, 2008, 30(4): 352-362.

[14] Chen Hsin Chu. Generalized reflexive matrices: special properties and applications [J]. SIAM J Matrix Anal Appl, 1998, 19(1): 140-153.

[15] Chen Hsin Chu, Sameh A H. Numerical linear algebra algorithms on the cedar system [J]. Parallel Computations and Their Impact on Mechanics, 1987, 86(1): 101-125.

[16] Chen Hsin Chu. The SAS domain decomposition method for structural analysis: center for supercomputing research and development [R]. Urbana: University of Illinois, 1988.

[17] Peng Xiangyang, Hu Xiyan, Zhang Lei. The reflexive and anti-reflexive solutions of the matrix equation $A^{H}XB = C$ [J]. Journal of Computational and Applied Mathematics, 2007, 200(2): 749-760.

[18] Peng Zhenyun, Hu Xiyan. The reflexive and anti-reflexive solutions of the matrix equation $AX = B$ [J]. Linear Algebra and its Applications, 2003, 375: 147-155.

[19] Peng Zhuohua, Hu Xiyan, Zhang Lei. An efficient algorithm for the least-squares reflexive solution of the matrix equation $A_1XB_1=C_1$, $A_2XB_2=C_2$ [J]. Applied Mathematics and Computation, 2006, 181(2): 988-999.

[20] Yamada I. The hybrid steepest descent method for the variational inequality problem over the intersection of fixed point sets of nonexpansive mappings[J]. Stud Comput Math, 2001, 8: 473-504.

[21] Konstantinos Slavakis, Yamada Isao, Sakaniwa Kohichi. Computation of symmetric positive definite Toeplitz matrices by the hybrid steepest descent method [J]. Signal Processing, 2003, 83(5): 1135-1140.

[22] Yamada I, Ogura N, Shirakawa N.A numerically robust hybrid steepest descent method for the convexly constrained generalized inverse problems [J]. Contemporary Mathematics, 2002, 313: 269-305.

The Iterative Algorithm for the Reflexive Optimal Approximation Solutions of Matrix Equations *AXB+CYD=E*

SUN He-ming[1], QI Zheng-ping[1], YANG Jia-wen[2]

(1. College of Science, Hehai University, Nanjing Jiangsu 211100, China;
2. Department of Fundations, Chuzhou Vocational and Technical College, Chuzhou Anhui 239000, China)

Abstract: An iterative algorithm to calculate the optimal approximation solutions of the generalized Sylvester matrix equations ***AXB+CYD=E*** over reflexive (anti-reflexive) matrix is studied by making use of the hybrid steepest descent method (HSDM). The given algorithm can be used to compute the optimal approximation solutions whether matrix equations ***AXB+CYD=E*** are consistent or not. The effectiveness of the proposed algorithm is verified by two numerical examples.

Key words: Sylvester matrix equations; Kronecker product; hybrid steepest descent method; optimal approximation; reflexive matrix

(责任编辑: 曾剑锋)