Computer Science and Application 计算机科学与应用, 2025, 15(10), 232-239
Published Online October 2025 in Hans. https://www.hanspub.org/journal/csa
https://doi.org/10.12677/csa.2025.1510263

Hans汉斯

一种解决半定规划问题的投影神经网络方法

张柯冕，张　杰，杨嘉妮

辽宁师范大学数学学院，辽宁　大连

收稿日期：2025年9月15日；录用日期：2025年10月17日；发布日期：2025年10月28日

摘　要

本文研究了半定规划(SDP)问题的神经网络方法。首先提出了一种基于投影算子的神经网络方法，然后，建立了神经网络平衡点与半定规划问题最优解之间的等价性，并证明了平衡点具有Lyapunov稳定性。数值模拟进一步证明了该网络的有效性。通过利用投影映射和SDP问题的结构，该神经网络方法可以有效地解决优化任务，为解决各种半定规划问题提供了一个实用的计算框架。

关键词

半定规划，投影神经网络，平衡点

A Projection Neural Network Method for Solving Semidefinite Programming Problems

Kemian Zhang, Jie Zhang, Jiani Yang

School of Mathematics, Liaoning Normal University, Dalian Liaoning

Received: September 15, 2025; accepted: October 17, 2025; published: October 28, 2025

Abstract

This paper studies the neural network method for semidefinite programming (SDP) problems. Firstly, a neural network method based on projection operator is proposed. Then, the equivalence between the equilibrium point of the neural network and the optimal solution of the semidefinite programming problem is established, and it is proved that the equilibrium point has Lyapunov stability. Numerical simulation further proves the effectiveness of the network. By using the structure of projection mapping and SDP problem, the neural network method can effectively solve the optimization task, and

文章引用: 张柯冕, 张杰, 杨嘉妮. 一种解决半定规划问题的投影神经网络方法[J]. 计算机科学与应用, 2025, 15(10): 232-239. DOI: 10.12677/csa.2025.1510263

provides a practical computational framework for solving various semidefinite programming problems.

Keywords

Semidefinite Programming, Projection Neural Network, Equilibrium Point

1. 引言

近年来，由于半定规划(SDP)问题在工程和组合优化中的广泛应用，已引起了广泛关注。半定规划问题包含线性规划、最小－最大特征值问题、最大行列式问题以及对数切比雪夫逼近问题等多个重要优化问题作为特例[1]，这类问题在控制理论、图论、统计学、结构优化、组合优化以及其他学科领域频繁出现[2] [3]。

神经网络方法特别适用于实时优化问题，具备多项显著优势。其类似电路的结构易于通过模拟电路或专用硬件实现，支持低功耗、高并行运算，满足高实时性需求。通过设计能量函数，该方法将约束优化转化为动态系统平衡点求解，简化了传统内点法中复杂的障碍函数设计。此外，连续时间神经网络可直接借助动态系统理论和常微分方程数值方法有效求解约束优化问题，并具有快速收敛能力，适用于实时计算场景。

本文旨在提出一种新的基于投影算子的神经网络来解决半定规划问题。首先，我们回顾了半定规划问题和神经网络的定义。然后，构建了投影神经网络，分析了所提神经网络的平衡点和半定规划问题的解是等价的，并证明了平衡点是 Lyapunov 稳定的，最后通过数值实验展示了所提方法的有效性。

通过本文的研究，我们希望能够为解决半定规划问题的方法研究提供新的思路。

2. 预备知识

2.1. 半定规划问题

我们考虑半定规划问题[4]：

$$\begin{aligned}(P)\ &\min\ \langle C,X\rangle\\ &\text{s.t. } \langle A_i,X\rangle=b_i,\quad i=1,2,\cdots,m,\quad X\geq 0,\end{aligned}\tag{1}$$

和其对偶问题：

$$\begin{aligned}(D)\ &\max\ b^{\mathrm{T}}y\\ &\text{s.t.}\quad \sum_{i=1}^{m}y_iA_i+S=C,\quad S\geq 0,\end{aligned}\tag{2}$$

其中，C,A_i,X,S 是 n 阶矩阵，并且 X,S 是对称矩阵，$b,y\in R^m$，$\langle C,X\rangle,\langle A_i,X\rangle$ 是矩阵的迹积，$\geq$ 表示半正定，也就是对于实对称矩阵 A 和 B，$A\geq B$ 表示 $A-B$ 是半正定的。

半定规划问题和对偶半定规划问题的解满足最优性条件[5]：

$$\begin{cases} \langle A_i, X\rangle = b_i, \quad i=1,2,\cdots,m, \quad X\geq 0, \\ \sum_{i=1}^{m} y_i A_i + S = C, \quad S\geq 0, \\ \langle X, S\rangle = 0. \end{cases} \tag{3}$$

令 $x=vec(X)$ 并且定义 $u=\left(x^{\mathrm{T}}, y^{\mathrm{T}}\right)^{\mathrm{T}}$。如果 X^* 和 y^* 满足(3)中的最优性条件，那么称 $u^*=\left(x^{*\mathrm{T}}, y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 是半定规划问题的一组解。

2.2. 一阶微分方程

一阶微分方程基本形式为

$$\dot{y}(t)=\mathcal{F}\left(y(t)\right), \quad y(t_0)=y_0\in\mathbb{R}^n, \tag{4}$$

这里 $\mathcal{F}:\mathbb{R}^n\to\mathbb{R}^n$ 是一个映射。

定义 1. 若点 $y^*=y(t^*)$ 满足 $\mathcal{F}(y^*)=0$，则称 y^* 为微分方程(4)的平衡点。若存在 y^* 的一个领域 $\Omega^*\subset\mathbb{R}^n$，使得 $\mathcal{F}(y^*)=0$，且对所有 $y\in\Omega^*\setminus\{y^*\}$，都有 $\mathcal{F}(y)\neq 0$，则称 y^* 为孤立平衡点。

2.3. 投影算子

设 $P_\Omega:\mathbb{R}^n\to\Omega$ 是一个投影算子，定义为 $P_\Omega(u)=\arg\min_{v\in\Omega}\|u-v\|$。若 $\Omega=\Omega_1\times\mathbb{R}^m$，其中 $\times$ 表示笛卡尔积，$\Omega_1\in\mathbb{R}^{n^2}$，则对任意 $u=\left(x^{\mathrm{T}}, y^{\mathrm{T}}\right)^{\mathrm{T}}\in\mathbb{R}^{n^2+m}$，有 $P_\Omega=\begin{pmatrix} P_{\Omega_1}(x) \\ y \end{pmatrix}$。

引理 1. [6]对于任意的 $(A,B)\in S_+^m\times S_+^m$，其中 S_+^m 为所有 $m\times m$ 实对称半正定矩阵的集合，我们有 $\langle A,B\rangle=0$ 当且仅当 $A=P_{S_+^m}(A-B)$。

命题 1. 设 S_+^n 为所有 $n\times n$ 实对称半正定矩阵的集合，有以下等式成立：

$$vec\left(P_{S_+^n}(X)\right)=P_{vec\left(S_+^n\right)}\left(vec(X)\right)$$

命题 2. 如果 Ω 是一个 $\mathbb{R}^n$ 上的闭凸子集，那么在闭凸集上的投影的一个基本性质是

$$\left[u-P_\Omega(u)\right]^{\mathrm{T}}\left[P_\Omega(u)-v\right]\geq 0, \quad \forall u\in\mathbb{R}^n, \ v\in\Omega \tag{5}$$

3. 投影神经网络

接下来，我们介绍所提出的投影神经网络模型

$$\begin{aligned} &\frac{\mathrm{d}u}{\mathrm{d}t}=P_\Omega\left[u-\beta(Mu+q)\right]-u, \\ &u(t_0)=u_0, \quad \beta>0, \end{aligned} \tag{6}$$

其中 $\Omega=vec\left(S_+^n\right)\times\mathbb{R}^m$，

$$M=\begin{pmatrix} 0 & -\overline{A}^{\mathrm{T}} \\ \overline{A} & 0 \end{pmatrix}, \overline{A}=\left(a^1,a^2,\cdots,a^m\right)^{\mathrm{T}}, a^i=vec(A_i), (i=1,2,\cdots,m),$$

$$q=\begin{pmatrix} c \\ -b \end{pmatrix}, c=vec(C), u_0\in\left\{u=\left(x^{\mathrm{T}}, y^{\mathrm{T}}\right)^{\mathrm{T}}\in\mathbb{R}^{n^2+m} \mid x\in vec\left(S_+^n\right), y\in\mathbb{R}^m\right\},$$

β 是尺度参数，用以表征神经网络的收敛速率。为了便于我们的分析：

定理 1. 设 $u^*=\left(x^{*\mathrm{T}},y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 是神经网络(6)的一个平衡点，则 u^* 满足半定规划问题的最优性条件(3)。反之，若 $X^*\in S_+^n$ 是半定规划问题(1)的一个最优解，则存在 $y^*\in\mathbb{R}^m$，使得 $u^*=\left(vec\left(X^*\right)^{\mathrm{T}},y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 是神经网络(6)的一个平衡点。

证明：假设 $u^*=\left(x^{*\mathrm{T}},y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 是神经网络(6)的一个平衡点，那么 $P_\Omega\left[u^*-\beta\left(Mu^*+q\right)\right]-u^*=0$，因此

$$
\begin{aligned}
&u^*-P_\Omega\left[u^*-\beta\left(Mu^*+q\right)\right]\\
&=\begin{pmatrix}x^*\\y^*\end{pmatrix}-P_\Omega\left[\begin{pmatrix}x^*\\y^*\end{pmatrix}-\beta\begin{pmatrix}-vec\left(\sum_{i=1}^m y_i^*A_i\right)+c\\\left\langle A_1,X^*\right\rangle-b_1\\\vdots\\\left\langle A_m,X^*\right\rangle-b_m\end{pmatrix}\right]\\
&=\begin{pmatrix}x^*\\y^*\end{pmatrix}-P_\Omega\begin{bmatrix}x^*-\beta\left(c-vec\left(\sum_{i=1}^m y_i^*A_i\right)\right)\\y_1^*-\beta\left(\left\langle A_1,X^*\right\rangle-b_1\right)\\\vdots\\y_m^*-\beta\left(\left\langle A_m,X^*\right\rangle-b_m\right)\end{bmatrix}\\
&=\begin{pmatrix}x^*\\y^*\end{pmatrix}-\begin{pmatrix}P_{vec\left(S_+^n\right)}\left[x^*-\beta\left(c-vec\left(\sum_{i=1}^m y_i^*A_i\right)\right)\right]\\y_1^*-\beta\left(\left\langle A_i,X^*\right\rangle-b_1\right)\\\vdots\\y_m^*-\beta\left(\left\langle A_m,X^*\right\rangle-b_m\right)\end{pmatrix}
\end{aligned}
$$

根据命题 1，得出

$$
P_{vec\left(S_+^n\right)}\left[x^*-\beta\left(c-vec\left(\sum_{i=1}^m y_i^*A_i\right)\right)\right]=vec\left[P_{S_+^n}\left[X^*-\beta\left(C-\sum_{i=1}^m y_i^*A_i\right)\right]\right]
$$

代入原式得

$$
\begin{aligned}
&u^*-P_\Omega\left[u^*-\beta\left(Mu^*+q\right)\right]\\
&=\begin{pmatrix}x^*-vec\left[P_{S_+^n}\left[X^*-\beta\left(C-\sum_{i=1}^m y_i^*A_i\right)\right]\right]\\\beta\left(\left\langle A_i,X^*\right\rangle-b_1\right)\\\vdots\\\beta\left(\left\langle A_m,X^*\right\rangle-b_m\right)\end{pmatrix}\\
&=0
\end{aligned}
$$

我们有

$$
\begin{cases}X^*=P_{S_+^n}\left[X^*-\beta\left(C-\sum_{i=1}^m y_i^*A_i\right)\right]\\\left\langle A_i,X^*\right\rangle=b_i,\quad i=1,2,3,\cdots,m.\end{cases}
$$

将 $A=X^*, B=\beta\left(C-\sum_{i=1}^{m} y_i^* A_i\right)$ 代入到引理 1 中，我们得到

$$\begin{cases}\left\langle A_i, X^*\right\rangle = b_i, \quad i=1,2,3,\cdots,m,\\ C-\sum_{i=1}^{m} y_i^* A_i \succeq 0, \quad X \succeq 0,\\ \left\langle X^*, C-\sum_{i=1}^{m} y_i^* A_i\right\rangle = 0.\end{cases}$$

这与(3)是等价的，即 $u^*=\left(x^{*\mathrm{T}}, y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 满足半定规划问题的最优性条件。反之亦然。

证毕。

定理 2. 神经网络(6)的平衡点 $u^*=\left(x^{*\mathrm{T}}, y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 在 Lyapunov 意义上是稳定的。

证明：令 $u^*=\left(x^{*\mathrm{T}}, y^{*\mathrm{T}}\right)^{\mathrm{T}}$ 是神经网络(6)的一个平衡点，我们有

$$u^*=P_{\Omega}\left[u^*-\beta\left(Mu^*+q\right)\right]-u^*$$

考虑以下函数：

$$V(u)=\beta\left[F(u)\right]^{\mathrm{T}}\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]-\frac{1}{2}\left\|u-P_{\Omega}\left[u-\beta F(u)\right]\right\|^2+\frac{1}{2}\left\|u-u^*\right\|^2, \quad u\in\Omega$$

这里 $F(u)=Mu+q$。显然，$F(u)$ 是连续可微的，并且 $V\left(u^*\right)=0$。

将 $\Omega=vec\left(S_+^n\right)\times\mathbb{R}^m$ (它是 $\mathbb{R}^{n^2+m}$ 的一个闭凸子集)、$u-\beta F(u)\in\mathbb{R}^{n^2+m}$、$u\in\Omega$ 代入(5)，我们有

$$\left[u-\beta F(u)-P_{\Omega}\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\left[P_{\Omega}\left[u-\beta F(u)\right]-u\right]\geq 0,$$

那么

$$\beta\left[F(u)\right]^{\mathrm{T}}\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]\geq\left\|u-P_{\Omega}\left[u-\beta F(u)\right]\right\|^2.$$

因此

$$V(u)\geq\frac{1}{2}\left\|u-P_{\Omega}\left[u-\beta F(u)\right]\right\|^2+\frac{1}{2}\left\|u-u^*\right\|^2\geq 0, \quad \forall u\in\Omega,$$

且显然有

$$V(u)>0, \quad \forall u\in\Omega, u\neq u^*.$$

与文献[7]中定理 3.2 的证明类似，我们可知：

$$\nabla V(u)=\beta F(u)+\left(\beta\nabla F(u)-I_n\right)\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]+u-u^*,$$

其中，I_n 是 n 阶单位矩阵。因此，沿着神经网络(6)的轨迹 $u=u\left(t,u_0\right)\left(t\geq 0\right)$，我们有

$$\begin{aligned}\frac{\mathrm{d}V\left[u(t)\right]}{\mathrm{d}t}&=\left[\nabla V(u)\right]^{\mathrm{T}}\frac{\mathrm{d}u}{\mathrm{d}t}\\ &=-\left[u-u^*+\beta F(u)\right]^{\mathrm{T}}\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]+\left\|u-P_{\Omega}\left[u-\beta F(u)\right]\right\|^2\\ &\quad-\beta\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\nabla F(u)\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right].\end{aligned}$$

因为 F 是可微的，并且

$$\nabla F(u)=M=\begin{pmatrix}0 & -\overline{A}^{\mathrm{T}}\\ \overline{A} & 0\end{pmatrix},$$

这是一个反对称矩阵，因此

$$\beta\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\nabla F(u)\left[u-P_{\Omega}\left[u-\beta F(u)\right]\right]=0.$$

将$u-\beta F(u)\in R^{n^2+m}, u^*\in\Omega$代入(3)中，我们得到

$$\begin{aligned}
&\left[u-\beta F(u)-P_\Omega\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\left[P_\Omega\left[u-\beta F(u)\right]-u^*\right]\\
&=\left[u-\beta F(u)-P_\Omega\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\left[P_\Omega\left[u-\beta F(u)\right]-u+u-u^*\right]\\
&=\left[u-P_\Omega\left[u-\beta F(u)\right]\right]^{\mathrm{T}}\left[\beta F(u)+u-u^*\right]-\left\|u-P_\Omega\left[u-\beta F(u)\right]\right\|^2-\beta F(u)^{\mathrm{T}}\left(u-u^*\right)\\
&=\left[\beta F(u)+u-u^*\right]^{\mathrm{T}}\left[u-P_\Omega\left[u-\beta F(u)\right]\right]-\left\|u-P_\Omega\left[u-\beta F(u)\right]\right\|^2-\beta\left(u-u^*\right)^{\mathrm{T}}F(u)\\
&\geq 0.
\end{aligned}$$

这意味着，

$$\left[\beta F(u)+u-u^*\right]^{\mathrm{T}}\left[u-P_\Omega\left[u-\beta F(u)\right]\right]\geq\left\|u-P_\Omega\left[u-\beta F(u)\right]\right\|^2+\beta\left(u-u^*\right)^{\mathrm{T}}F(u).$$

因此

$$\frac{\mathrm{d}V\left[u(t)\right]}{\mathrm{d}t}\leq-\beta\left(u-u^*\right)^{\mathrm{T}}F(u). \tag{7}$$

对于任意的$u\in\Omega$，我们有

$$\left(u-u^*\right)^{\mathrm{T}}F(u)=\left(u-u^*\right)^{\mathrm{T}}\left[F(u)-F\left(u^*\right)\right]+u^{\mathrm{T}}F\left(u^*\right)-u^{*\mathrm{T}}F\left(u^*\right).$$

由神经网络(6)和最优性条件(3)，我们得到

$$\begin{aligned}
u^{*\mathrm{T}}F\left(u^*\right)&=\left(x^{*\mathrm{T}},y^{*\mathrm{T}}\right)\begin{pmatrix}-vec\left(\sum_{i=1}^m y_i^*A_i\right)+c\\ \left\langle A_1,X^*\right\rangle-b_1\\ \vdots\\ \left\langle A_m,X^*\right\rangle-b_m\end{pmatrix}\\
&=\left(x^{*\mathrm{T}},y^{*\mathrm{T}}\right)\begin{pmatrix}-vec\left(\sum_{i=1}^m y_i^*A_i\right)+c\\ 0\\ \vdots\\ 0\end{pmatrix}\\
&=\left\langle X^*,C-\sum_{i=1}^m y_i^*A_i\right\rangle\\
&=0.
\end{aligned}$$

因此

$$\begin{aligned}
&\left(u-u^*\right)^{\mathrm{T}}F(u)\\
&=\left(u-u^*\right)^{\mathrm{T}}\left[F(u)-F\left(u^*\right)\right]+u^{\mathrm{T}}F\left(u^*\right)\\
&=\left(u-u^*\right)^{\mathrm{T}}M\left(u-u^*\right)+\left(x^{\mathrm{T}},y^{\mathrm{T}}\right)\begin{pmatrix}-vec\left(\sum_{i=1}^m y_i^*A_i\right)+c\\ 0\\ \vdots\\ 0\end{pmatrix}\\
&=\left(u-u^*\right)^{\mathrm{T}}M\left(u-u^*\right)+\left\langle X,C-\sum_{i=1}^m y_i^*A_i\right\rangle.
\end{aligned}$$

上式结合反对称矩阵 $M, X \in S_+^n, C-\sum_{i=1}^m y_i^* A_i \in S_+^n, \beta>0$ 以及(7)，得到

$$\frac{\mathrm{d}V\left[u(t)\right]}{\mathrm{d}t} \le -\beta\left(u-u^*\right)^{\mathrm{T}} F(u) \le -\beta\left\langle X, C-\sum_{i=1}^m y_i^* A_i \right\rangle \le 0$$

这意味着神经网络(6)的平衡点在 Lyapunov 意义上是稳定的。

证毕。

4. 数值实验

下面的例子是根据文献[8]，经过适当修改而改编的。

考虑半定规划问题(1)和其对偶问题(2)，令 $n=3, m=2$，

$$A_1=\begin{pmatrix} -1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 4 \end{pmatrix}, A_2=\begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix},$$

$$b=\begin{pmatrix} -3 \\ 1 \end{pmatrix}, C=\begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 8 & 0 \\ 0 & 0 & 0 & 16 \end{pmatrix}.$$

实验在 MATLAB R2022 上进行，所使用的常微分方程求解器为 ode45。图 1 展示了基于神经网络(6)的 $y(t)$ 的轨迹，参数取 $\beta=100$，初始点取 $X_0=I_4$ 和 $y_0=\begin{pmatrix} 1 \\ 1 \end{pmatrix}$。

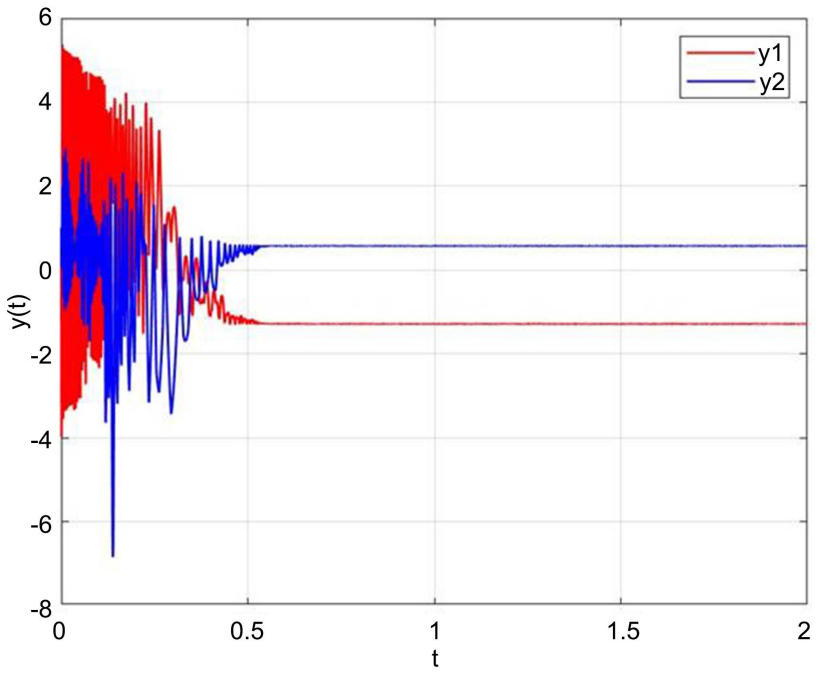

Figure 1. $y(t)$ trajectory based on neural network (6)

图 1. 基于神经网络(6)的 $y(t)$ 轨迹

5. 结论

本文提出了一种新的基于投影算子的神经网络来解决半定规划问题。

致　谢

本人衷心感谢张杰老师在学术道路上的悉心指导。

参考文献

[1] Vandenberghe, L. and Boyd, S. (1996) Semidefinite Programming. *SIAM Review*, **38**, 49-95. https://doi.org/10.1137/1038003

[2] Boyd, S., El Ghaoui, L., Feron, E. and Balakrishnan, V. (1994) Linear Matrix Inequalities in System and Control Theory. Society for Industrial and Applied Mathematics. https://doi.org/10.1137/1.9781611970777

[3] Mohar, B. and Poljak, S. (1993) Eigenvalues in Combinatorial Optimization. In: Brualdi, R.A., Friedland, S. and Klee, V., Eds., *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, Springer, 107-151. https://doi.org/10.1007/978-1-4613-8354-3_5

[4] Alizadeh, F. (1995) Interior Point Methods in Semidefinite Programming with Applications to Combinatorial Optimization. *SIAM Journal on Optimization*, **5**, 13-51. https://doi.org/10.1137/0805002

[5] Bi, H., Zhao, X. and Ren, J. (2022) A New Projection Contraction Algorithm for Semidefinite Programming. *Procedia Computer Science*, **208**, 627-634. https://doi.org/10.1016/j.procs.2022.10.086

[6] Tseng, P. (1998) Merit Functions for Semi-Definite Complemetarity Problems. *Mathematical Programming*, **83**, 159-185. https://doi.org/10.1007/bf02680556

[7] Fukushima, M. (1992) Equivalent Differentiable Optimization Problems and Descent Methods for Asymmetric Variational Inequality Problems. *Mathematical Programming*, **53**, 99-110. https://doi.org/10.1007/bf01585696

[8] Bazaraa, M.S., Jarvis, J.J. and Sherali, H.D. (2011) Linear Programming and Network Flows. John Wiley & Sons.