Math. Program., Ser. A (2016) 155:435–470
DOI 10.1007/s10107-014-0853-2

FULL LENGTH PAPER

A semismooth Newton-CG based dual PPA for matrix spectral norm approximation problems

Caihua Chen · Yong-Jin Liu · Defeng Sun · Kim-Chuan Toh

Received: 24 June 2013 / Accepted: 10 December 2014 / Published online: 21 December 2014

Abstract We consider a class of matrix spectral norm approximation problems for finding an affine combination of given matrices having the minimal spectral norm subject to some prescribed linear equality and inequality constraints. These problems arise often in numerical algebra, engineering and other areas, such as finding Chebyshev polynomials of matrices and fastest mixing Markov chain models. Based on classical analysis of proximal point algorithms (PPAs) and recent developments on

Caihua Chen is supported in part by the National Science Foundation of Jiangsu Province under Project Grant No. BK20130550, and the National Natural Science Foundation of China under Project Grant No. 71271112. Yong-Jin Liu is supported in part by the National Natural Science Foundation of China under Project Grant No. 11001180 and 11371255, the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry, and the Program for Liaoning Excellent Talents in Universities. Defeng Sun is supported in part by Academic Research Fund under Grant R-146-000-149-112. Kim-Chuan Toh is supported in part by Academic Research Fund under Grant R-146-000-168-112.

C. Chen (✉)
International Center of Management Science and Engineering, School of Management and Engineering, Nanjing University, Nanjing, Jiangsu, China
e-mail: cchenhuayx@gmail.com

Y.-J. Liu
School of Science, Shenyang Aerospace University, Shenyang, China
e-mail: yongjin.liu.77@gmail.com

D. Sun
Department of Mathematics and Risk Management Institute, National University of Singapore, 10 Lower Kent Ridge Road, Singapore 119076, Singapore
e-mail: matsundf@nus.edu.sg

K.-C. Toh
Department of Mathematics, National University of Singapore, 10 Lower Kent Ridge Road, Singapore 119076, Singapore
e-mail: mattohkc@nus.edu.sg

semismooth analysis of nonseparable spectral operators, we propose a semismooth Newton-CG based dual PPA for solving the matrix norm approximation problems. Furthermore, when the primal constraint nondegeneracy condition holds for the subproblems, our semismooth Newton-CG method is proven to have at least a superlinear convergence rate. We also design efficient implementations for our proposed algorithm to solve a variety of instances and compare its performance with the nowadays popular first order alternating direction method of multipliers (ADMM). The results show that our algorithm, which is warm-started with an initial point obtained from the ADMM, substantially outperforms the pure ADMM, especially for the constrained cases and it is able to solve the problems robustly and efficiently to a relatively high accuracy.

Keywords Spectral norm approximation · Spectral operator · PPA · Semismooth Newton-CG method

Mathematics Subject Classification 90C06 · 90C25 · 65F10

1 Introduction

Let $\Re^{m\times n}$ be the space of $m \times n$ real matrices equipped with the standard inner product $\langle X,\ Y\rangle = \mathrm{Tr}(X^T Y)$ for $X, Y \in \Re^{m\times n}$. Given a family of matrices $\{A_1, A_2, \ldots, A_p\}$ in $\Re^{m\times n}$, define the linear operator $\mathcal{A}$ and its adjoint $\mathcal{A}^*$, respectively, by

$$\mathcal{A}(X) := [\langle A_1, X\rangle, \ldots, \langle A_p, X\rangle]^T, \quad \mathcal{A}^*(y) := \sum_{k=1}^{p} y_k A_k, \quad \forall\, X \in \Re^{m\times n},\ y \in \Re^p.$$

In this paper, we consider the following matrix norm approximation (abbreviated as MNA) problem:

$$\min_{y\in\Re^p} \Big\{ \|A_0 - \mathcal{A}^* y\|_2 \mid By - b \in \mathcal{Q} \Big\}, \tag{1}$$

where $A_0 \in \Re^{m\times n}$, $B \in \Re^{(n_1+n_2)\times p}$ are given matrices, $b \in \Re^{n_1+n_2}$, $\mathcal{Q} = \{0\}^{n_1} \times \Re^{n_2}_+$ is a polyhedral cone, and $\|\cdot\|_2$ denotes the matrix spectral norm which is defined as the largest singular value of a matrix. Without loss of generality, we assume that $m \le n$.

For the unconstrained version of (1), some theoretical questions such as uniqueness and characterizations of the best approximation have been analyzed in [26,48]. In the more general setting of a normed linear space, a general characterization theorem for the best approximation of an element in a normed linear space by elements of a finite dimensional subspace was established in [36, p. 170]. Here, in our setting, the feasible set is no longer a linear subspace.

The MNA problems arise from a variety of fields, such as numerical algebra and engineering. An illustrative example is the problem of finding the degree-t Chebyshev polynomial of a given matrix $A \in \Re^{n\times n}$, as studied in [15,27,40]. In this problem, one is interested in finding the degree-t monic polynomial $\bar{q}_t$ which minimizes the spectral norm of $q_t(A)$, i.e.,

$$\min \{\|q_t(A)\|_2 \mid \ q_t \text{ is a monic polynomial of degree } t\}. \tag{2}$$

Since $q_t(A)$ is an affine combination of the matrices $I, A, \ldots, A^t$, the matrix Chebyshev polynomial approximation problem is actually a special case of the MNA problem (1).

In contrast to the unconstrained example (2), some other problems may have prescribed linear constraints, for example, the fastest mixing Markov chain (FMMC) problem studied in [2,3]. Let $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ be an undirected connected graph with n nodes. The FMMC problem is to find a symmetric stochastic matrix P with $P_{ij} = 0$ for $(i, j) \notin \mathcal{E}$ that minimizes $\mu(P)$, where

$$\mu(P) = \max_{i=2,\ldots,n} |\lambda_i(P)|$$

and $\lambda_i(P)$ is the ith largest eigenvalue of P in magnitude. Let d be the vector of transition probabilities on the non-self-loop edges (labeled by $l = 1, 2, \ldots, p$). Define the matrix $B \in \Re^{n\times p}$ by

$$B_{il} := \begin{cases} 1, & \text{if edge } l \text{ is incident to vertex } i, \\ 0, & \text{otherwise} \end{cases}$$

and the matrix $E^{(l)}$ by

$$E^{(l)}_{ij} := \begin{cases} 1, & \text{if edge } l \text{ is incident to vertices } i \text{ and } j, \ i \neq j, \\ -1, & \text{if edge } l \text{ is incident to vertex } i, \ i = j, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

Then by the analysis in [2,3], the FMMC problem can be rewritten as an MNA problem in terms of d as follows:

$$\min \left\{ \|I - (1/n)\mathbf{1}\mathbf{1}^T + \sum_{l=1}^{p} d_l E^{(l)}\|_2 \mid d \geq 0, \ \ Bd \leq 1 \right\}. \tag{4}$$

Note that if the constraints in (4) are dropped, the resulting unconstrained problem

$$\min_d \ \|I - (1/n)\mathbf{1}\mathbf{1}^T + \textstyle\sum_{l=1}^{p} d_l E^{(l)}\|_2 \tag{5}$$

is a mathematical model for the fastest distributed linear averaging (FDLA) problem with symmetric weights [44].

The above examples serve to motivate the study of numerical algorithms for solving the MNA problems. It is easy to show that the MNA problem (1) can be converted to the following semidefinite programming (SDP):

$$\min \left\{ t \mid \begin{bmatrix} tI_m & A_0 - \mathcal{A}^* y \\ (A_0 - \mathcal{A}^* y)^T & tI_n \end{bmatrix} \succeq 0, \quad By \in b + \mathcal{Q} \right\}. \tag{6}$$

Thus it may be solved by standard SDP packages such as SDPT3 [39], SeDuMi [37], or SDPNAL [47]. However, this SDP reformulation is potentially very computationally expensive since one has to deal with a linear matrix inequality involving matrices of dimensions $(m+n)\times(m+n)$ instead of matrices of dimensions $m\times n$ as in the original problem. The computational cost and memory requirement are especially unnecessarily high when we have large $m+n$, but $m \ll n$. In the extreme case when $m=1$, it is certainly not wise to solve the MNA problem (1) via (6). Instead one should deal with (1) directly since it is just a second-order cone problem, which requires far lower computational cost to solve compared to the SDP (6). For large scale problems, one may attempt to apply the projected subgradient method to the MNA problem (1) directly instead of via the SDP reformulation (6). However, as pointed by the authors in [2], the algorithm is relatively slow in terms of number of iterations and has no simple stopping criterion guaranteeing a certain level of suboptimality while compared to a primal-dual interior point method for FMMC problems. In addition, the projection step may not be easy to compute when the polyhedral constraints are not simple constraints such as $y\in\mathcal{Q}$.

In this paper, we propose a semismooth Newton-CG based dual PPA (SNDPPA) which is able to solve large scale MNA problems (1) with p or n large and m moderate. We also propose to use a first order alternating direction method of multipliers (ADMM) to obtain an initial point to warm-start the SNDPPA. As we shall see later, in each iteration of the ADMM, the subproblem involved can either be solved by a fast algorithm or has a closed form solution, due to recent advances in [9]. Hence, the ADMM can easily be applied to solve the MNA problem. However, the performance of the ADMM is well known to behave erratically in that for some problems, it can efficiently deliver a good accuracy solution, while frequently for some others, it stagnates with a low accuracy solution. The slow convergence/stagnation difficulty of the ADMM can be clearly seen in the numerical results presented later in the paper. The SNDPPA we propose here is able to overcome the weakness of the ADMM while also able to capitalize on the strength of the ADMM in using it for warm-start. For the dual PPA, we note that the subproblem in each iteration is an unconstrained minimization problem whose objective function is convex and continuously differentiable but not twice continuously differentiable. However, since the corresponding gradient is strongly semismooth, we are able to apply the inexact semismooth Newton method to solve the unconstrained minimization subproblem with at least a superlinear convergence rate. At each iteration of the semismooth Newton method, the Newton direction is computed by a preconditioned conjugate gradient (CG) method. In each CG iteration, one would calculate each matrix–vector multiplication in at most $O(m^2n+pmn)+O(p(n_1+n_2))$ flops. This is in contrast to the inexact interior point proposed in [38] for the SDP reformulation where the matrix–vector multiplication would require more than $O((m+n)^3+p(m+n)^2)+O(p(n_1+n_2))$ flops. We also design efficient implementations for our proposed algorithm to solve a variety of large scale matrix norm approximation problems and compare its performance with the popular first order ADMM. The numerical results show that our semismooth Newton-CG dual PPA substantially outperforms the pure ADMM, especially on constrained MNA and FMMC problems and it is able to solve a variety of large scale instances with p up to 19,176 and m up to 2,000 robustly and efficiently to a relatively high accuracy.

A problem related to (1) is to minimize the largest eigenvalue ($\lambda_1(A_0 - \mathcal{A}^* y)$) of an affine combination of given symmetric matrices. In [5], the authors studied the convex smooth approximations of the nonsmooth function $\lambda_1(A_0 - \mathcal{A}^* y)$. We should mention that it is possible to modify our proposed algorithm to solve the problem of minimizing $\lambda_1(A_0 - \mathcal{A}^* y)$ subject to polyhedral constraints on y. But this is not the focus of this paper.

The contributions of our paper are as follows. Firstly, we design a semismooth Newton-CG based PPA to solve the MNA problems involving the matrix spectral norm. The idea of using a semismooth Newton-CG based PPA to solve convex optimization problems involving matrix norms is not new. For example, Jiang et al. [22,23] considered similar ideas of using semismooth/smoothing Newton-CG method to solve nuclear norm regularized matrix least squares problems. However, there exists a fundamental difference on matrix spectral operators employed. The concept of spectral operators was defined by Ding [8] in his Ph.D. thesis for the study of matrix programming. See also Definition 2.2. While the spectral operator used in [22,23] is separable, the one used in our study is inherently non-separable due to the use of the spectral norm instead of the nuclear norm. If one is only interested in low accuracy solutions by employing the first order methods such as the ADMM, then there is virtually no conceptual difference in solving the nuclear norm or the spectral norm regularized convex matrix optimization problems. The big difference arises when the derivative of the spectral operator is involved as it is the case in this work. To the best of our knowledge, this is the first time that large scale MNA problems with matrix dimension larger than 500 can be solved with high accuracy, efficiently and reliably, with the help of the theory on non-separable spectral operators. Secondly, we address various numerical issues pertaining to the efficient implementation of the semismooth Newton-CG method for solving the subproblems in the dual PPA. In particular, we analyze the structure of matrix–vector multiplication in Newton-CG carefully in Sect. 4.4 and show that it can be computed efficiently only with the reduced SVD of an $m \times n$ matrix. Hence, with modest effort, the Newton-CG is easy to implement to solve the subproblems of the dual PPA. Thirdly, we establish the global and local convergence of the dual PPA under certain standard conditions. We also establish the suplinear convergence of the semismooth Newton-CG method under the primal constraint nondegeneracy conditions of the subproblems. Finally, we demonstrate numerically that although the proposed SNDPPA is more complex to implement compared to the ADMM, its superior efficiency and reliability in solving constrained MNA and FMMC problems in comparison to the ADMM fully reveals the necessity of having a second-order information employed method such as the SNDPPA for solving constrained MNA problems.

The remaining parts of this paper are organized as follows. In Sect. 2, we list some preliminaries on the semismooth mappings, the Moreau–Yosida regularization, and the spectral operator associated with the spectral norm. In Sect. 3, we introduce the framework of the inexact dual PPA for solving the MNA problem and establish its global and local convergence under certain conditions. In Sect. 4, we present a semismooth Newton-CG method for solving the subproblems in the dual PPA, and we establish its suplinear convergence under the primal constraint nondegeneracy conditions of the subproblems. In addition, some numerical issues pertaining to the

efficient implementation of the semismooth Newton-CG method are also addressed. In Sect. 5, we first give the details on the implementation of the classical ADMM for the MNA problem and then report numerical experiments for our proposed algorithm and the comparison of its performance against the ADMM. We conclude our paper in Sect. 6.

Notation For any given positive integer m and n, we denote by I_n, $\mathbf{1}_{m\times n}$ and $\mathbf{0}_{m\times n}$ the $n\times n$ identity matrix, the $m\times n$ matrices of ones and zeros, respectively. We also use $\mathbf{1}_n$ and $\mathbf{0}_n$ to denote the vectors of ones and zeros, respectively. We frequently drop m, n from the above notations when their dimensions are clear from the context. For any $x\in\Re^n$, Diag(x) denotes the diagonal matrix with diagonal entries $x_i, i=1,\ldots,n$, while for any $X\in\Re^{m\times n}$, diag(X) denotes the main diagonal of X. For any $\eta>0$, we write $\mathbb{B}_\eta := \{x\in\Re^m \mid \|x\|_1\le\eta\}$ and $\mathcal{B}_\eta := \{X\in\Re^{m\times n} \mid \|X\|_*\le\eta\}$, where $\|\cdot\|_*$ denotes the nuclear norm which is defined as the sum of the singular values of a matrix. If $\eta=1$, we just use $\mathbb{B}$ and $\mathcal{B}$ to denote the unit l_1 norm ball and nuclear norm ball, respectively. Let $\alpha\subseteq\{1,\ldots,n\}$ be an index set, we use $|\alpha|$ to represent the cardinality of α and X_α to denote the sub-matrix of X obtained by removing all the columns of X not in α. Let $\beta\subseteq\{1,\ldots,n\}$ be another index set, we use $X_{\alpha\beta}$ to denote the $|\alpha|\times|\beta|$ sub-matrix of X obtained by removing all the rows of $X\in\Re^{m\times n}$ not in α and all the columns of X not in β. The Hadamard product between two matrices is denoted by "$\circ$", i.e., for any two matrices X and Y in $\Re^{m\times n}$, the (i,j)-th entry of $Z=X\circ Y$ is $Z_{ij}=X_{ij}Y_{ij}$. We use $\|\cdot\|$ to denote the matrix Frobenius norm throughout this paper. For any given linear space H, $H^\perp$ denotes the orthogonal complement of H.

2 Preliminaries

In this section, we review and develop some results on the semismooth mappings, the Moreau–Yosida regularization and the spectral operator associated with the matrix spectral norm, which are useful for our subsequent discussion.

Let $\mathcal{X}$ and $\mathcal{Y}$ be two finite dimensional real Euclidean spaces and $\mathcal{O}$ be an open set in $\mathcal{X}$. Suppose that $\Phi:\mathcal{O}\to\mathcal{Y}$ is a locally Lipschitz continuous function on the open set $\mathcal{O}$. By Rademacher's theorem, Φ is almost everywhere F (Fréchet)-differentiable in $\mathcal{O}$. Let $\mathcal{D}_\Phi$ be the set of points where Φ is differentiable. Let $\Phi'(x)$ be the derivative of Φ at $x\in\mathcal{D}_\Phi$. Then the B-subdifferential of Φ at $x\in\mathcal{O}$ is defined by $\partial_B\Phi(x) := \left\{\lim_{\mathcal{D}_\Phi\ni x^k\to x}\Phi'(x^k)\right\}$ and Clarke's generalized Jacobian [6] of Φ at $x\in\mathcal{O}$, denoted by $\partial\Phi(x)$, is the convex hull of $\partial_B\Phi(x)$.

Definition 2.1 (c.f. [11,31]) Let $\Phi:\mathcal{O}\subset\mathcal{X}\to\mathcal{Y}$ be a locally Lipschitz continuous function on the open set $\mathcal{O}$ and x be a point in $\mathcal{O}$. The function Φ is said to be G-semismooth at $x\in\mathcal{O}$ if for any $y\to x$ and $V\in\partial\Phi(y)$, $\|\Phi(y)-\Phi(x)-V(y-x)\| = o(\|y-x\|)$. The function Φ is said to be strongly G-semismooth at x if for any $y\to x$ and $V\in\partial\Phi(y)$, $\|\Phi(y)-\Phi(x)-V(y-x)\| = O(\|y-x\|^2)$. Furthermore, if the (strongly) G-semismooth function Φ is also directionally differentiable at x, then Φ is said to be (strongly) semismooth at x.

Let $f : \mathcal{X} \to (-\infty, +\infty]$ be a closed proper convex function [34] and $\eta > 0$ be a positive constant. Then the Moreau–Yosida regularization [30,46] of f at $x \in \mathcal{X}$ associated with η is defined by

$$\psi_f^\eta(x) := \min_{y \in \mathcal{X}} \Big\{ f(y) + \frac{1}{2\eta}\|y - x\|^2 \Big\}. \tag{7}$$

The unique optimal solution to (7), denoted by $P_f^\eta(x)$, is called the proximal point at x associated with f and η. Let $\mathcal{C}$ be a closed convex set in $\mathcal{X}$ and $\chi_{\mathcal{C}}(\cdot)$ be the indicator function of $\mathcal{C}$. Then, for any $\eta > 0$, the proximal point at x associated with $\chi_{\mathcal{C}}(\cdot)$ is the metric projection of x onto $\mathcal{C}$, which is denoted by $\Pi_{\mathcal{C}}(x)$.

Proposition 2.1 (c.f. [19,24]) *Let $f : \mathcal{X} \to (-\infty, +\infty]$ be a closed proper convex function and $\eta > 0$ be a positive constant. Let ψ_f^η be the Moreau–Yosida regularization associated with f and η and P_f^η be the corresponding proximal point mapping. Then, the following properties hold.*

(i) *ψ_f^η is a continuously differentiable convex function, and*

$$\nabla \psi_f^\eta(x) = \frac{1}{\eta}\big(x - P_f^\eta(x)\big), \quad \forall\, x \in \mathcal{X}.$$

(ii) *Let $g : \mathcal{X} \to (-\infty, +\infty]$ be defined by*

$$g(x) = f^*(x/\eta), \quad \forall\, x \in \mathcal{X},$$

where f^* is the conjugate of f. Then any $x \in \mathcal{X}$ has the following unique Moreau decomposition

$$x = P_f^\eta(x) + P_g^\eta(x).$$

In what follows, we shall calculate the proximal point mapping associated with the spectral norm, which plays a crucial role in our numerical implementation. Let $X \in \Re^{m\times n}$ be given with the following singular value decomposition (SVD):

$$X = U[\mathrm{Diag}(\sigma(X))\ 0]V^T, \tag{8}$$

where $U \in \Re^{m\times m}$, $V \in \Re^{n\times n}$ are orthogonal matrices, $\sigma_1(X) \geq \cdots \geq \sigma_m(X) \geq 0$ are the singular values of X, and $\sigma(X) := (\sigma_1(X), \sigma_2(X), \ldots, \sigma_m(X))^T$. In later discussions, when the dependence of $\sigma_i(X)$ on X is clear from the context, we will drop X from these notations.

Proposition 2.2 *Let $f(\cdot) := \|\cdot\|_2$ be defined on $\Re^{m\times n}$ and $\eta > 0$. Let $X \in \Re^{m\times n}$ be given with the SVD as in (8). Then it holds that*

$$P_f^\eta(X) = X - \Pi_{\mathcal{B}_\eta}(X), \tag{9}$$

where $\Pi_{\mathcal{B}_\eta}(X) = U[\mathrm{Diag}(\Pi_{\mathbb{B}_\eta}(\sigma(X)))\ 0]V^T$.

Proof Recall that the conjugate of a norm is the indicator function of the dual norm unit ball. Since the dual norm of $\|\cdot\|_2$ is $\|\cdot\|_*$, we know that $f^* = \chi_{\mathcal{B}}$, the indicator function of the unit nuclear norm ball $\mathcal{B}$. Hence (9) follows from part (ii) of Proposition 2.1. Note that $\Pi_{\mathcal{B}_\eta}(X)$ is the unique solution of the following optimization problem: $\min\{\frac{1}{2}\|Y - X\|^2 \mid \|Y\|_* \le \eta\}$. Since both $\|\cdot\|_F$ and $\|\cdot\|_*$ are unitarily invariant, by von Neumann's trace inequality [41]:

$$\|\sigma(Y) - \sigma(X)\| \le \|Y - X\|, \quad \forall\, Y \in \Re^{m\times n},$$

we know that

$$\Pi_{\mathcal{B}_\eta}(X) = U[\mathrm{Diag}\big(\Pi_{\mathbb{B}_\eta}(\sigma(X))\big)\ 0]V^T,$$

where $\Pi_{\mathbb{B}_\eta}(\sigma(X))$ is the unique optimal solution to $\min\{\frac{1}{2}\|y - \sigma(X)\|^2 \mid \|y\|_1 \le \eta\}$. □

With the above preparation, we are ready to give the exact expression of the projection operator $\Pi_{\mathcal{B}_\eta}(\cdot)$. Let X admit the SVD as in (8). Define the vector $s(\sigma)$ by

$$s_i(\sigma) = \frac{1}{i}\Big(\sum\nolimits_{j=1}^{i}\sigma_j - \eta\Big), \quad i = 1, 2, \ldots, m.$$

Let $k_1(\sigma)$ and $k_2(\sigma)$ denote, respectively, the maximal indices of the following two sets:

$$\{i \mid \sigma_i > s_i(\sigma),\ 1 \le i \le m\}, \quad \{i \mid \sigma_i \ge s_i(\sigma),\ 1 \le i \le m\}. \tag{10}$$

From the breakpoint search algorithm in [18, Sect. 5], it follows that

$$\Pi_{\mathbb{B}_\eta}(\sigma) = \begin{cases} \sigma, & \text{if } \|\sigma\|_1 \le \eta, \\ \max\{\sigma - s_{k_1(\sigma)}(\sigma), 0\}, & \text{otherwise} \end{cases} \tag{11}$$

and hence $\Pi_{\mathcal{B}_\eta}(X)$ can be computed analytically.

Definition 2.2 (c.f. [8]) Let h be a symmetric mapping from $\Re^m$ to $\Re^m$, that is, for any signed permutation matrix Q,

$$h(x) = Q^T h(Qx), \quad \forall\, x \in \Re^m.$$

The spectral operator $H : \Re^{m\times n} \to \Re^{m\times n}$ with respect to the symmetric function h is defined by

$$H(X) := U[\mathrm{Diag}(h(\sigma))\ 0]V^T, \quad \forall\, X \in \Re^{m\times n}.$$

Since $\Pi_{\mathbb{B}_\eta}(\cdot)$ is a symmetric function, we know that $\Pi_{\mathcal{B}_\eta}(\cdot)$ is a matrix spectral operator with respect to the nonseparable function $\Pi_{\mathbb{B}_\eta}(\cdot)$. To employ the techniques developed in [8], we introduce some notations. Write

$$g(\sigma) := (g_1(\sigma), g_2(\sigma), \ldots, g_m(\sigma))^T = \Pi_{\mathbb{B}_\eta}(\sigma) \tag{12}$$

and use $\mu_1 > \mu_2 > \cdots > \mu_t$ to denote the nonzero singular values of X. Define the index sets α_k, $k = 1, 2, \ldots, t+1$ by

$$\alpha_k := \{i \,|\, \sigma_i = \mu_k,\ 1 \le i \le m\},\ k = 1, 2, \ldots, t;\quad \alpha_{t+1} := \{i \,|\, \sigma_i = 0,\ 1 \le i \le m\}.$$

If $\Pi_{\mathbb{B}_\eta}$ is F-differentiable at σ, we define three matrices $\Omega(\sigma), \Gamma(\sigma), \mathcal{F}(\sigma) \in \Re^{m\times m}$ by

$$[\Omega(\sigma)]_{ij} := \begin{cases} \frac{g_i(\sigma)-g_j(\sigma)}{\sigma_i-\sigma_j}, & \text{if } \sigma_i \ne \sigma_j, \\ (g'(\sigma))_{ii} - (g'(\sigma))_{i(i+1)}, & \text{if } \sigma_i = \sigma_j, i \in \alpha_k, |\alpha_k| \ne 1, \\ (g'(\sigma))_{ii}, & \text{otherwise}, \end{cases} \tag{13}$$

$$[\Gamma(\sigma)]_{ij} := \begin{cases} \frac{g_i(\sigma)+g_j(\sigma)}{\sigma_i+\sigma_j}, & \text{if } \sigma_i + \sigma_j \ne 0, \\ (g'(\sigma))_{ii}, & \text{otherwise}, \end{cases} \tag{14}$$

$$[\mathcal{F}(\sigma)]_{ij} := \begin{cases} (g'(\sigma))_{ij}, & \text{if } i \ne j, \\ (g'(\sigma))_{i(i+1)}, & \text{if } i = j \in \alpha_k, |\alpha_k| \ne 1, \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

and the vector $\Upsilon(\sigma) \in \Re^m$ by

$$[\Upsilon(\sigma)]_i = \begin{cases} \frac{g_i(\sigma)}{\sigma_i}, & \text{if } \sigma_i \ne 0, \\ (g'(\sigma))_{ii}, & \text{otherwise}. \end{cases}$$

Define the two linear operators S and T by

$$S(A) := \frac{1}{2}(A + A^T), \quad T(A) := \frac{1}{2}(A - A^T), \quad \forall\, A \in \Re^{m\times m}.$$

Proposition 2.3 *Suppose that $X \in \Re^{m\times n}$ has the SVD as in (8). Then*

(i) *The matrix spectral operator $\Pi_{\mathcal{B}_\eta}$ is F-differentiable at X if and only if σ satisfies $\|\sigma\|_1 < \eta$ or $\|\sigma\|_1 > \eta$ but $k_1(\sigma) = k_2(\sigma)$. In this case, the function g in (12) is F-differentiable at σ and the derivative of $\Pi_{\mathcal{B}_\eta}$ at X satisfies that for any $H \in \Re^{m\times n}$,*

$$\begin{aligned} \Pi'_{\mathcal{B}_\eta}(X)H &= U\Big[\Omega(\sigma)\circ S(\widetilde{H}_1) + \mathrm{Diag}\big(\mathcal{F}(\sigma)\mathrm{diag}(\widetilde{H}_1)\big) + \Gamma(\sigma)\circ T(\widetilde{H}_1),\quad \mathrm{Diag}(\Upsilon(\sigma))\widetilde{H}_2\Big]V^T \\ &= U\Big[\Omega(\sigma)\circ S(\widetilde{H}_1) + \mathrm{Diag}\big(\mathcal{F}(\sigma)\mathrm{diag}(\widetilde{H}_1)\big) + \Gamma(\sigma)\circ T(\widetilde{H}_1)\Big]V_1^T \\ &\quad + U\mathrm{Diag}(\Upsilon(\sigma))U^T H(V_2V_2^T), \end{aligned}$$

where $\widetilde{H}_1 \in \Re^{m\times m}$, $\widetilde{H}_2 \in \Re^{m\times(n-m)}$ *and* $[\widetilde{H}_1 \ \ \widetilde{H}_2] = U^T H V$; $V_1 \in \Re^{m\times m}$, $V_2 \in \Re^{m\times(n-m)}$ *and* $V = [V_1 \ V_2]$.

(ii) $\Pi_{\mathcal{B}_\eta}$ *is strongly G-semismooth everywhere on* $\Re^{m\times n}$.

Proof By Eq. (11), it can be easily verified that $\Pi_{\mathbb{B}_\eta}(\cdot)$ is F-differentiable at σ if and only if $\|\sigma\|_1 < \eta$ or $\|\sigma\|_1 > \eta$ but $k_1(\sigma) = k_2(\sigma)$. Then invoking [8, Theorem 3.6], one can establish part (i) of the assertion. Moreover, since $\Pi_{\mathbb{B}_\eta}(\cdot)$ is a piecewise linear function, we know $\Pi_{\mathbb{B}_\eta}(\cdot)$ is strongly G-semismooth. This, together with [8, Theorem 3.12], proves part (ii) of the assertion. □

Proposition 2.4 (c.f. [25]) *Suppose that* $X \in \Re^{m\times n}$ *has the SVD as in* (8). *Let* r *be the rank of* X. *Then the orthogonal matrices* P *and* W *satisfy*

$$P[\mathrm{Diag}(\sigma)\ 0] = [\mathrm{Diag}(\sigma)\ 0]W$$

if and only if there exist orthogonal matrices $Q \in \Re^{r\times r}$, $Q' \in \Re^{(m-r)\times(m-r)}$ *and* $Q'' \in \Re^{(n-r)\times(n-r)}$ *such that*

$$P = \begin{bmatrix} Q & 0 \\ 0 & Q' \end{bmatrix} \quad \text{and} \quad W = \begin{bmatrix} Q & 0 \\ 0 & Q'' \end{bmatrix}.$$

3 A dual proximal point algorithm framework

In this section, we shall introduce the framework of the inexact dual PPA for solving the MNA problem and establish its global and local convergence.

3.1 The proximal point algorithm

In various fields of applied mathematics, many problems can be equivalently formulated as a maximal monotone inclusion problem, that is, given a, possibly multi-valued, maximal monotone operator $\mathcal{T} : \mathcal{X} \to 2^{\mathcal{X}}$, where $\mathcal{X}$ is a real Hilbert space, it is to find an $x \in \mathcal{X}$ such that $0 \in \mathcal{T}(x)$. The PPA [35] applied to the maximal monotone inclusion problem takes the following scheme

$$x^{k+1} \approx p_{\lambda_k}(x^k) := (I + \lambda_k \mathcal{T})^{-1}(x^k),$$

where the parameter $\lambda_k > 0$ is bounded away from zero. Rockafellar [35] suggested computing x^{k+1} only approximately to satisfy the following accuracy criteria:

$$\left\| x^{k+1} - p_{\lambda_k}(x^k) \right\| \le \varepsilon_k, \quad \varepsilon_k > 0, \quad \sum_{k=1}^{\infty} \varepsilon_k < \infty, \tag{16}$$

$$\left\| x^{k+1} - p_{\lambda_k}(x^k) \right\| \le \delta_k \left\| x^{k+1} - x^k \right\|, \quad \delta_k > 0, \quad \sum_{k=1}^{\infty} \delta_k < \infty. \tag{17}$$

In [35], he showed that the sequence generated above converges (in the weak topology) to a zero point of $\mathcal{T}$, if it exists. Moreover, if $\lambda_k \uparrow \lambda_\infty \leq \infty$ and $\mathcal{T}^{-1}$ is Lipschitz continuous at 0, then condition (17) ensures that the local convergence is linear and the rate is approximately proportional to $1/\lambda_\infty$. If in addition $\lambda_\infty = \infty$, the convergence becomes superlinear.

Possibly due to its versatility and effectiveness, the PPA has received continuous attention from numerous researchers and is well accepted as a powerful tool for solving various classes of convex optimization problems, see, e.g., [16,28,33,42,47]. In this section, we consider the dual PPA, i.e., applying the idea to the maximal monotone operator associated with the dual problem. By rewriting (1) as

$$\min\Big\{\|X\|_2 \;\mid\; \mathcal{A}^*y + X = A_0,\;\; By - b \in \mathcal{Q}\Big\}, \tag{18}$$

we can derive the following explicit form of its dual

$$\begin{aligned}
&\max_{Z\in\Re^{m\times n},\, w\in\mathcal{Q}^*}\ \inf_{X\in\Re^{m\times n},\, y\in\Re^p} \Big\{\|X\|_2 - \langle Z,\ \mathcal{A}^*y + X - A_0\rangle - \langle w,\ By - b\rangle\Big\}\\
&= \max_{Z\in\Re^{m\times n},\, w\in\mathcal{Q}^*}\ \inf_{X\in\Re^{m\times n},\, y\in\Re^p} \Big\{\|X\|_2 - \langle X,\ Z\rangle - \langle \mathcal{A}Z + B^T w,\ y\rangle + \langle A_0,\ Z\rangle + \langle b,\ w\rangle\Big\}\\
&= \max_{Z\in\Re^{m\times n},\, w\in\mathcal{Q}^*} \Big\{\langle A_0,\ Z\rangle + \langle b,\ w\rangle - \chi_{\mathcal{B}}(Z) \mid \mathcal{A}Z + B^T w = 0\Big\}\\
&= \max_{Z\in\Re^{m\times n},\, w\in\mathcal{Q}^*} \Big\{\langle A_0,\ Z\rangle + \langle b,\ w\rangle \mid \mathcal{A}Z + B^T w = 0,\ Z\in\mathcal{B}\Big\},
\end{aligned}$$

where the second equality holds since the conjugate of $\|\cdot\|_2$ is the indicator function of the unit nuclear norm ball $\mathcal{B}$, and $\mathcal{Q}^*(:= \Re^{n_1}\times\Re^{n_2}_+)$ is the dual cone of $\mathcal{Q}$. Obviously, we can recast the dual maximization problem equivalently as the following minimization problem:

$$\min\Big\{-\langle A_0, Z\rangle - \langle b, w\rangle \mid \mathcal{A}Z + B^T w = 0,\ \|Z\|_* \leq 1,\ w\in\mathcal{Q}^*\Big\}. \tag{19}$$

For the convergence analysis later, we assume that the Slater condition for (19) holds, i.e., there exists $(Z, w) \in \Re^{m\times n}\times\Re^{n_1+n_2}$ such that

$$\mathcal{A}Z + B^T w = 0, \quad \|Z\|_* < 1, \quad w_i > 0,\ i = n_1+1,\ldots,n_1+n_2. \tag{20}$$

If in addition, the Slater condition also holds for (18), i.e., $\exists\ \hat{y}$ such that $B\hat{y} - b \in$ relint($\mathcal{Q}$), where "relint" denotes the relative interior, then the strong duality holds for (18) and (19), and (X, Z, w) is an optimal solution pair of the primal and dual problems if and only if it is the solution to the following KKT conditions:

$$\begin{aligned}
\mathcal{A}^*y + X = A_0, &\quad By - b\in\mathcal{Q},\\
\mathcal{A}Z + B^T w = 0, &\quad \|Z\|_*\leq 1,\ w\in\mathcal{Q}^*,\\
\langle X,\ Z\rangle - \|X\|_2 = 0, &\quad \langle By - b,\ w\rangle = 0.
\end{aligned}$$

Write

$$\mathcal{T}_l(X, y) = \begin{pmatrix} \partial \|X\|_2 \\ 0 \end{pmatrix} + \partial \chi_{\mathcal{F}_1}(X, y), \quad \forall\, X \in \Re^{m\times n},\ y \in \Re^p,$$

$$\mathcal{T}_g(Z, w) = -\begin{pmatrix} A_0 \\ b \end{pmatrix} + \partial \chi_{\mathcal{F}_2}(Z, w), \quad \forall\, Z \in \Re^{m\times n},\ w \in \Re^{n_1+n_2}$$

and

$$p_\lambda(Z, w) = (I + \lambda \mathcal{T}_g)^{-1}(Z, w), \quad \forall\, Z \in \Re^{m\times n},\ w \in \Re^{n_1+n_2},$$

where $\mathcal{F}_1$ and $\mathcal{F}_2$ are the feasible sets of (18) and (19) respectively.

For any given $Z^k \in \Re^{m\times n}$, $w^k \in \Re^{n_1+n_2}$ and $\lambda_k > 0$, it is easy to see that $p_{\lambda_k}(Z^k, w^k)$ is the unique optimal solution to the following minimization problem:

$$\begin{array}{l} \min\ -\langle A_0, Z\rangle - \langle b, w\rangle + \dfrac{1}{2\lambda_k}\|Z - Z^k\|^2 + \dfrac{1}{2\lambda_k}\|w - w^k\|^2 \\ \text{s.t. } \mathcal{A}Z + B^T w = 0,\ \|Z\|_* \le 1,\ w \in \mathcal{Q}^*. \end{array} \tag{21}$$

Let K be the epigraph of the nuclear norm, i.e.,

$$K := \{(t, X) \in \Re \times \Re^{m\times n} \mid t \ge \|X\|_*\}$$

and K^* be its dual which is the epigraph of the spectral norm. Note that the constraint that $\|Z\|_* \le 1$ is equivalent to $(1, -Z) \in K$. Direct computation shows that the dual of (21) is given by

$$\begin{aligned} &\max_{y\in\Re^p,\,(t,X)\in K^*} \inf_{Z\in\Re^{m\times n}, w\in\mathcal{Q}^*} \left\{ \begin{array}{l} -\langle A_0, Z\rangle - \langle b, w\rangle + \dfrac{1}{2\lambda_k}(\|Z - Z^k\|^2 + \|w - w^k\|^2) \\ +\langle y,\ \mathcal{A}Z + B^T w\rangle - t + \langle X,\ Z\rangle \end{array} \right\} \\ &= \max_{y\in\Re^p,\,X\in\Re^{m\times n}} \left\{ \begin{array}{l} -\|X\|_2 + \dfrac{1}{2\lambda_k}\left(\|Z^k\|^2 - \left\|Z^k - \lambda_k(\mathcal{A}^* y + X - A_0)\right\|^2\right) \\ +\dfrac{1}{2\lambda_k}\left(\|w^k\|^2 - \left\|\Pi_{\mathcal{Q}^*}\left[w^k - \lambda_k(By - b)\right]\right\|^2\right) \end{array} \right\} \\ &= \max_{y\in\Re^p} \left\{ \begin{array}{l} \dfrac{1}{2\lambda_k}\left\|\Pi_{\mathcal{B}}\left(Z^k - \lambda_k(\mathcal{A}^* y - A_0)\right) - \left(Z^k - \lambda_k\left(\mathcal{A}^* y - A_0\right)\right)\right\|^2 \\ +\dfrac{1}{2\lambda_k}\left(\|Z^k\|^2 - \left\|Z^k - \lambda_k(\mathcal{A}^* y - A_0)\right\|^2\right) \\ +\dfrac{1}{2\lambda_k}\left(\|w^k\|^2 - \left\|\Pi_{\mathcal{Q}^*}\left[w^k - \lambda_k(By - b)\right]\right\|^2\right) \end{array} \right\} \\ &:= \max_{y\in\Re^p} \theta_k(y), \end{aligned} \tag{22}$$

where the first equality holds when

$$t = \|X\|_2,\ w = \Pi_{\mathcal{Q}^*}[w^k - \lambda_k(By - b)] \tag{23}$$

and

$$Z = Z^k - \lambda_k(\mathcal{A}^* y + X - A_0), \tag{24}$$

and the second equality holds when

$$X = \frac{1}{\lambda_k}\Big[Z^k - \lambda_k(\mathcal{A}^* y - A_0) - \Pi_{\mathcal{B}}(Z^k - \lambda_k(\mathcal{A}^* y - A_0))\Big]. \tag{25}$$

Clearly, the Slater condition (20) asserts that the optimal solution set of (22) is non-empty. Let y^{k+1} be a maximizer of (22). Combining (23), (24) and (25), we have

$$p_{\lambda_k}(Z^k, w^k) = \begin{bmatrix} \Pi_{\mathcal{B}}(Z^k - \lambda_k(\mathcal{A}^* y^{k+1} - A_0)) \\ \Pi_{\mathcal{Q}^*}(w^k - \lambda_k(B y^{k+1} - b)) \end{bmatrix}. \tag{26}$$

Therefore, in order to implement the PPA, one needs to solve (22) and then update the variable (Z, w) by

$$(Z^{k+1}, w^{k+1}) \approx p_{\lambda_k}(Z^k, w^k).$$

and X by

$$X^{k+1} \approx (Z^k - \lambda_k(\mathcal{A}^* y^{k+1} - A_0) - Z^{k+1})/\lambda_k.$$

In view of (26), we are able to present an inexact dual PPA framework:

Algorithm 3.1 (A dual PPA framework)
Given (Z^0, w^0, y^0), $\lambda_0 > 0$ and $\varepsilon > 0$, at the k-th iteration, do the following steps:

Step 1. *For fixed Z^k, w^k and y^k, compute an approximate maximizer*

$$y^{k+1} \approx \arg\max\{\theta_k(y) \mid y \in \Re^p\},$$

where θ_k is defined in (22).

Step 2. *Update the variables Z^{k+1}, w^{k+1} and X^{k+1} via*

$$\begin{aligned} Z^{k+1} &= \Pi_{\mathcal{B}}\left(Z^k - \lambda_k(\mathcal{A}^* y^{k+1} - A_0)\right), \\ w^{k+1} &= \Pi_{\mathcal{Q}^*}\left(w^k - \lambda_k(B y^{k+1} - b)\right), \\ X^{k+1} &= \left(Z^k - \lambda_k\left(\mathcal{A}^* y^{k+1} - A_0\right) - Z^{k+1}\right)/\lambda_k. \end{aligned}$$

Step 3. *If $\max\{\|A_0 - \mathcal{A}^* y^{k+1} - X^{k+1}\|, \|\Pi_{\mathcal{Q}^*}(b - B y^{k+1})\|\} \le \varepsilon$, stop; else, update λ_k to λ_{k+1}, end.*

3.2 Convergence analysis

In Step 1 of the dual PPA, we use the following stopping criteria[1] suggested in [33]:

$$\max \theta_k(y) - \theta_k(y^{k+1}) \le \frac{\varepsilon_k^2}{2\lambda_k}, \quad \varepsilon_k > 0, \quad \sum_{k=1}^{\infty} \varepsilon_k < \infty, \tag{27}$$

$$\max \theta_k(y) - \theta_k(y^{k+1}) \le \frac{\delta_k^2}{2\lambda_k}(\|Z^{k+1} - Z^k\|^2 + \|w^{k+1} - w^k\|^2),$$

$$\delta_k > 0, \ \sum_{k=1}^{\infty} \delta_k < \infty, \tag{28}$$

$$\left\|\nabla_y \theta_k(y^{k+1})\right\| \le \frac{\delta_k'}{\lambda_k} \left\| \begin{pmatrix} Z^{k+1} - Z^k \\ w^{k+1} - w^k \end{pmatrix} \right\| \quad 0 \le \delta_k' \to 0. \tag{29}$$

Next we present two results on the global and local convergence of the dual PPA. The proofs are omitted since they follow directly from [33, Theorems 4–5].

Theorem 3.1 (Global Convergence) *Let the inexact PPA be executed with stopping criterion* (27). *Suppose that the primal problem* (18) *satisfies the Slater condition. Then the sequence* $\{(Z^{k+1}, w^{k+1})\} \subset \mathcal{B} \times \mathcal{Q}^*$ *generated by the inexact PPA converges to an optimal solution of* (19). *Moreover, the sequence* $\{y^k\}$ *is also bounded and any of its accumulation point is an optimal solution of* (1).

Theorem 3.2 (Local Convergence) *Let the dual PPA be executed with stopping criteria* (27) *and* (28). *Suppose that the Slater condition holds for* (18). *If* $\mathcal{T}_g^{-1}$ *is Lipschitz continuous at the origin with the modulus* a_g, *then* $\{(Z^{k+1}, w^{k+1})\}$ *converges to an optimal solution* $(\overline{Z}, \bar{w})$ *of* (19), *and*

$$\left\| \begin{pmatrix} Z^{k+1} - \overline{Z} \\ w^{k+1} - \bar{w} \end{pmatrix} \right\| \le \nu_k \left\| \begin{pmatrix} Z^k - \overline{Z} \\ w^k - \bar{w} \end{pmatrix} \right\|, \quad \text{for all } k \text{ sufficiently large}, \tag{30}$$

where $\nu_k = [a_g(a_g + \lambda_k^2)^{-1/2} + \delta_k](1 - \delta_k)^{-1} \to a_g(a_g^2 + \lambda_\infty^2)^{-1/2} < 1$. *Moreover, the conclusion about* $\{y^k\}$ *in Theorem* 3.1 *is valid.*

If in addition to (28) *and the condition on* $\mathcal{T}_g^{-1}$, *one also has* (29) *and that* $\mathcal{T}_l^{-1}$ *is Lipschitz continuous at the origin with modulus* a_l $(\ge a_g)$, *then* $\{y^{k+1}\}$ *converges to the unique optimal solution* $\bar{y}$ *of* (1), *and*

$$\left\| \begin{pmatrix} X^{k+1} - \overline{X} \\ y^{k+1} - \bar{y} \end{pmatrix} \right\| \le \nu_k' \left\| \begin{pmatrix} Z^{k+1} - Z^k \\ w^{k+1} - w^k \end{pmatrix} \right\|, \quad \text{for all } k \text{ sufficiently large},$$

where $\overline{X} = A_0 - \mathcal{A}^* \bar{y}$, *and* $\nu_k' = a_l(1 + \delta_k')/\lambda_k \to a_l/\lambda_\infty$.

[1] The stopping criteria introduced here are for the convergence analysis of the PPA. Since it is not easy to obtain the upper bound of $\max \theta_k(y)$, we use (29) to terminate the y-subproblem in the practical implementation of the dual PPA.

4 A semismooth Newton-CG method for inner problems

In this section, we will apply the well-known inexact semismooth Newton method to approximately solve the unconstrained subproblem (22). Using Proposition 2.1(i), we know that the first order optimality condition for (22) is given by

$$0 = \nabla\theta_k(y) := \mathcal{A}\Pi_{\mathcal{B}}\big[Z^k - \lambda_k(\mathcal{A}^* y - A_0)] + B^T \Pi_{\mathcal{Q}^*}[w^k - \lambda_k(By - b)\big].$$

Since $\Pi_{\mathcal{B}}(\cdot)$ and $\Pi_{\mathcal{Q}^*}(\cdot)$ are Lipschitz continuous, $\nabla\theta_k(\cdot)$ is also Lipschitz continuous. Hence Clarke's generalized Jacobian of $\nabla\theta_k$ (which is the generalized Hessian of θ_k and we denote it by $\partial^2\theta_k$) is well defined. Since it is difficult to derive an exact characterization of $\partial^2\theta_k$, we will slightly modify the classical semismooth Newton method by selecting elements in $\hat{\partial}^2\theta_k$ instead of $\partial^2\theta_k$, where $\hat{\partial}^2\theta_k$ is a set-valued mapping defined by

$$\begin{aligned}\hat{\partial}^2\theta_k(y) := -\lambda_k\big[&\mathcal{A}\partial\Pi_{\mathcal{B}}(Z^k - \lambda_k(\mathcal{A}^* y - A_0))\mathcal{A}^* \\ &+ B^T \partial\Pi_{\mathcal{Q}^*}(w^k - \lambda_k(By - b))B\big], \quad y \in \Re^p.\end{aligned}$$

Note that by [6, p. 75] and [20, Example 2.5],

$$\begin{aligned}\partial^2\theta_k(y)h &\subseteq \partial\Big(\mathcal{A}\Pi_{\mathcal{B}}\big[Z^k - \lambda_k(\mathcal{A}^* y - A_0)]\Big)h + \partial\Big(B^T \Pi_{\mathcal{Q}^*}[w^k - \lambda_k(By - b)]\Big)h \\ &= -\lambda_k\big[\mathcal{A}\partial\Pi_{\mathcal{B}}(Z^k - \lambda_k(\mathcal{A}^* y - A_0))\mathcal{A}^* h \\ &\quad + B^T \partial\Pi_{\mathcal{Q}^*}(w^k - \lambda_k(By - b))Bh\big] \\ &= \hat{\partial}^2\theta_k(y)h, \quad \forall\, h \in \Re^p.\end{aligned}$$

It is, however, not clear to us if $\partial^2\theta_k(y) = \hat{\partial}^2\theta_k(y)$ holds. Fortunately, $\hat{\partial}^2\theta_k(y)$ is sufficient for our use both theoretically and numerically.

4.1 Characterization of $\hat{\partial}^2\theta_k$

To obtain the explicit expression of $\hat{\partial}^2\theta_k$, it suffices to characterize $\partial\Pi_{\mathcal{B}}(\cdot)$ and $\partial\Pi_{\mathcal{Q}^*}(\cdot)$. For a given $Y \in \Re^{m\times n}$, suppose that it has the following SVD:

$$Y = U[\mathrm{Diag}(\sigma)\ 0]V^T, \tag{31}$$

where $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_m)^T$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_r > 0 = \sigma_{r+1} = \cdots = \sigma_m$. When $\|Y\|_* = 1$, we know from direct calculation that $k_1(\sigma) = r$ and $k_2(\sigma) = m$, where $k_1(\sigma)$ and $k_2(\sigma)$ are defined in (10) with $\eta = 1$ since $\mathcal{B}$ is the unit nuclear norm ball. For any positive integer $N \in [r,\ m]$, we define the following four index sets:

$$\begin{aligned}&\alpha_1 := \{1, 2, \ldots, r\}, && \alpha_2 := \{r+1, r+2, \ldots, N\}, \\ &\alpha_3 := \{N+1, N+2, \ldots, m\}, && \alpha_4 := \{m+1, m+2, \ldots, n\}.\end{aligned} \tag{32}$$

In another case that $\|Y\|_* > 1$, it is easily seen that $k_1(\sigma) \le k_2(\sigma) \le r$. Given an integer $N \in [k_1(\sigma),\ k_2(\sigma)]$, we partition the set $\{1, 2, \ldots, m\}$ into the following four subsets:

$$\begin{array}{ll} \beta_1 := \{1, 2, \ldots, k_1(\sigma)\}, & \beta_2 := \{k_1(\sigma)+1, k_1(\sigma)+2, \ldots, N\}, \\ \beta_3 := \{N+1, N+2, \ldots, k_2(\sigma)\}, & \beta_4 := \{k_2(\sigma)+1, k_2(\sigma)+2, \ldots, m\}. \end{array}$$

and write

$$\gamma_1 := \beta_1 \cup \beta_2, \quad \gamma_2 := \beta_3 \cup \beta_4.$$

Also, for the sake of simplicity, we will drop the parameter σ in $\Omega(\sigma)$, $\Gamma(\sigma)$, $F(\sigma)$ and $\Upsilon(\sigma)$, and represent the notations by Ω, Γ, $\mathcal{F}$ and Υ, respectively.

The next proposition characterizes exactly $\partial_B \Pi_{\mathcal{B}}(\cdot)$ at any $Y \in \Re^{m\times n}$ whose convex hull is Clarke's generalized Jacobian of $\Pi_{\mathcal{B}}(\cdot)$.

Proposition 4.1 *Let $Y \in \Re^{m\times n}$ admit the SVD as in (31).*

(i) *If $\|Y\|_* < 1$, then $\partial_B \Pi_{\mathcal{B}}(Y)$ is a singleton set consisting of the identity operator from $\Re^{m\times n}$ to itself.*

(ii) *If $\|Y\|_* = 1$, for $\mathcal{V} \in \partial_B \Pi_{\mathcal{B}}(Y)$, either $\mathcal{V}$ is the identity operator or there exist an integer $N \in [r,\ m]$, $(\Omega^{\infty}_{\alpha_2\alpha_3}, \Gamma^{\infty}_{\alpha_2\alpha_2}, \Gamma^{\infty}_{\alpha_2\alpha_3}, \Upsilon^{\infty}_{\alpha_2}) \in \mathcal{S}_N$ and singular vector matrices U^{∞}, V^{∞} of Y such that for any $H \in \Re^{m\times n}$,*

$$\mathcal{V}H = U^{\infty}\left[W^{\infty} - \frac{\mathrm{Tr}(\widetilde{H}_{11})}{N}\begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix},\ \mathrm{Diag}\begin{bmatrix} \boldsymbol{1}_r \\ \Upsilon^{\infty}_{\alpha_2} \\ 0 \end{bmatrix}\widetilde{H}_2 \right](V^{\infty})^T, \quad (33)$$

where the matrix $W^{\infty} \in \Re^{m\times m}$ is defined by

$$\begin{aligned} W^{\infty} := & \begin{bmatrix} & \boldsymbol{1}_{N\times N} & \boldsymbol{1}_{r\times(m-N)} \\ \boldsymbol{1}_{(m-N)\times r} & (\Omega^{\infty}_{\alpha_2\alpha_3})^T & \Omega^{\infty}_{\alpha_2\alpha_3} \\ & & 0 \end{bmatrix} \circ S(\widetilde{H}_1) \\ & + \begin{bmatrix} \boldsymbol{1}_{r\times r} & \boldsymbol{1}_{r\times(m-r)} & \\ \boldsymbol{1}_{(m-r)\times r} & \Gamma^{\infty}_{\alpha_2\alpha_2} & \Gamma^{\infty}_{\alpha_2\alpha_3} \\ & (\Gamma^{\infty}_{\alpha_2\alpha_3})^T & 0 \end{bmatrix} \circ T(\widetilde{H}_1) \end{aligned}$$

with $\widetilde{H}_1 \in \Re^{m\times m}$, $\widetilde{H}_2 \in \Re^{m\times(n-m)}$, $[\widetilde{H}_1\ \widetilde{H}_2] = (U^{\infty})^T H V^{\infty}$ and $\widetilde{H}_{11}$ is the matrix extracted from the first N columns and rows of $\widetilde{H}_1$, and $\mathcal{S}_N$ is the subset of $\Re^{(N-r)\times(m-N)} \times \Re^{(N-r)\times(N-r)} \times \Re^{(N-r)\times(m-N)} \times \Re^{N-r}$ defined by

$$\mathcal{S}_N := \left\{ \lim_{\mathcal{L}_N \ni Y^j \to Y} \left(\Omega(\sigma(Y^j))_{\alpha_2\alpha_3},\ \Gamma(\sigma(Y^j))_{\alpha_2\alpha_2},\ \Gamma(\sigma(Y^j))_{\alpha_2\alpha_3},\ \Upsilon(\sigma(Y^j))_{\alpha_2} \right) \right\}$$

with $\mathcal{L}_N = \{Z \mid Z \in \mathcal{D}_{\Pi_{\mathcal{B}}},\ \|Z\|_* > 1,\ k_1(\sigma(Z)) = N\}$.

(iii) *If* $\|Y\|_* > 1$, *for* $\mathcal{V} \in \partial_B \Pi_{\mathcal{B}}(Y)$, *there exist an integer* $N \in [k_1(\sigma),\, k_2(\sigma)]$, $\Omega^{\infty}_{\beta_2\beta_3} \in \mathcal{S}_N$ *and singular vector matrices* U^{∞}, V^{∞} *of* Y *such that for any* $H \in \Re^{m\times n}$,

$$\mathcal{V}H = U^{\infty}\left[W^{\infty} - \frac{\mathrm{Tr}(\widetilde{H}_{11})}{N}\begin{bmatrix} I_N & 0\\ 0 & 0\end{bmatrix},\ \ \mathrm{Diag}\begin{bmatrix}\Upsilon_{\gamma_1}\\ 0\end{bmatrix}\widetilde{H}_2\right](V^{\infty})^T, \tag{34}$$

where the matrix $W^{\infty} \in \Re^{m\times m}$ *is defined by*

$$W^{\infty} := \begin{bmatrix} \boldsymbol{1}_{N\times N} & \begin{matrix}\Omega_{\beta_1\beta_3}\\ \Omega^{\infty}_{\beta_2\beta_3}\end{matrix} & \Omega_{\gamma_1\beta_4}\\ (\Omega_{\beta_1\beta_3})^T\ (\Omega^{\infty}_{\beta_2\beta_3})^T & & 0\\ (\Omega_{\gamma_1\beta_4})^T & & \end{bmatrix} \circ S(\widetilde{H}_1) + \begin{bmatrix}\Gamma_{\gamma_1\gamma_1} & \Gamma_{\gamma_1\gamma_2}\\ (\Gamma_{\gamma_1\gamma_2})^T & 0\end{bmatrix} \circ T(\widetilde{H}_1),$$

here, $\widetilde{H}_1 \in \Re^{m\times m}$, $\widetilde{H}_2 \in \Re^{m\times(n-m)}$, $[\widetilde{H}_1\ \widetilde{H}_2] = (U^{\infty})^T H V^{\infty}$ *and* $\widetilde{H}_{11}$ *is the matrix extracted from the first* N *columns and rows of* $\widetilde{H}_1$, *and* $\mathcal{S}_N$ *is the subset of* $\Re^{(N-k_1(\sigma))\times(k_2(\sigma)-N)}$ *defined by*

$$\mathcal{S}_N := \left\{ \lim_{\mathcal{L}_N \ni Y^j \to Y} \Omega(\sigma(Y^j))_{\beta_2\beta_3} \right\}$$

with $\mathcal{L}_N = \{Z \mid Z \in \mathcal{D}_{\Pi_{\mathcal{B}}},\ k_1(\sigma(Z)) = N\}$.

Proof See the "Appendix". □

The characterization of $\partial\Pi_{\mathcal{Q}^*}(\cdot)$ at $z \in \Re^{n_1+n_2}$ is very simple given the special structure of the polyhedral cone $\mathcal{Q}^* = \Re^{n_1} \times \Re^{n_2}_+$. Define the following index sets:

$$\begin{aligned}
\mathcal{J}_1 &:= \{i : z_i > 0,\ n_1+1 \le i \le n_1+n_2\} \cup \{1, 2, \ldots, n_1\},\\
\mathcal{J}_2 &:= \{i : z_i = 0,\ n_1+1 \le i \le n_1+n_2\},\\
\mathcal{J}_3 &:= \{i : z_i < 0,\ n_1+1 \le i \le n_1+n_2\}.
\end{aligned}$$

For any given $z \in \Re^{n_1+n_2}$, direct computation shows that

$$\Pi_{\mathcal{Q}^*}(z) = \begin{cases} z_i, & i \in \{1, 2, \ldots, n_1\},\\ \max(z_i, 0), & i \in \{n_1+1, n_1+2, \ldots, n_1+n_2\}.\end{cases} \tag{35}$$

For any given $\xi \in \Re$, the subdifferential of $\max(\xi, 0)$ is given as follows:

$$\partial \max(\xi, 0) = \begin{cases} \{1\}, & \xi > 0,\\ \{0\}, & \xi < 0,\\ [0,\ 1], & \xi = 0.\end{cases} \tag{36}$$

By combining (35), (36) and the definitions of $\mathcal{J}_i$, $i = 1, 2, 3$, we have that $\mathcal{V}$ is an element of $\partial\Pi_{\mathcal{Q}^*}(z)$ if and only if there exists a vector $a \in [0, 1]^{|\mathcal{J}_2|}$ such that

$$\mathcal{V}h = \begin{bmatrix} h_{\mathcal{J}_1} \\ a \circ h_{\mathcal{J}_2} \\ 0 \end{bmatrix}, \ \forall\, h \in \Re^{n_1+n_2}. \tag{37}$$

Remark 4.1 In the implementation of our semismooth Newton-CG method, we need to select an element $\mathcal{V}_1^0 \in \partial \Pi_{\mathcal{B}}(Y)$ and an element $\mathcal{V}_2^0 \in \partial \Pi_{\mathcal{Q}^*}(z)$. If $\|Y\|_* \le 1$, $\mathcal{V}_1^0$ is chosen as the identity operator from $\Re^{m\times n}$ to $\Re^{m\times n}$. For the case that $\|Y\|_* > 1$, we take $U^\infty = U$, $V^\infty = V$ and $N = k_1(\sigma)$ in (34). Thus $\beta_2 = \emptyset$ and for any $H \in \Re^{m\times n}$,

$$\mathcal{V}_1^0 H = U^\infty \left[W^\infty - \frac{\mathrm{Tr}(\widetilde{H}_{11})}{N} \begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix}, \quad \mathrm{Diag}\left[\begin{matrix} \Upsilon_{\gamma_1} \\ 0 \end{matrix} \right] \widetilde{H}_2 \right] (V^\infty)^T, \tag{38}$$

where the matrix $W^\infty \in \Re^{m\times m}$ is defined by

$$W^\infty := \begin{bmatrix} \mathbf{1}_{N\times N} & \Omega_{\gamma_1\gamma_2} \\ (\Omega_{\gamma_1\gamma_2})^T & 0 \end{bmatrix} \circ S(\widetilde{H}_1) + \begin{bmatrix} \Gamma_{\gamma_1\gamma_1} & \Gamma_{\gamma_1\gamma_2} \\ (\Gamma_{\gamma_1\gamma_2})^T & 0 \end{bmatrix} \circ T(\widetilde{H}_1)$$

with $\widetilde{H}_1 \in \Re^{m\times m}$, $\widetilde{H}_2 \in \Re^{m\times(n-m)}$, $[\widetilde{H}_1 \ \widetilde{H}_2] = U^T H V$ and $\widetilde{H}_{11}$ being the matrix extracted from the first N columns and rows of $\widetilde{H}_1$. As to the selection of $\mathcal{V}_2^0$, we take $a = 0$ in (37) and then

$$\mathcal{V}_2^0 h = \begin{bmatrix} h_{\mathcal{J}_1} \\ 0 \end{bmatrix}, \ \forall\, h \in \Re^{n_1+n_2}. \tag{39}$$

4.2 Constraint nondegeneracy

For the convergence analysis of the semismooth Newton-CG method, we need the concept of constraint nondegeneracy which is originally introduced by Robinson [32] and extended by Bonnans and Shapiro [1]. Let $\mathcal{X}$ and $\mathcal{Y}$ be two finite dimensional spaces, $\Phi : \mathcal{X} \to \mathcal{Y}$ be a continuously differentiable function and $\mathcal{C}$ be a closed convex set. We use $T_{\mathcal{C}}(x)$ and $\mathrm{lin}(T_{\mathcal{C}}(x))$ to denote the tangent cone of $\mathcal{C}$ at x and its linearity space, respectively. A feasible point $\bar{x}$ to the feasibility problem $\{\Phi(x) \in \mathcal{C},\ x \in \mathcal{X}\}$ is said to be constraint nondegenerate if

$$\Phi'(\bar{x})\mathcal{X} + \mathrm{lin}(T_{\mathcal{C}}(\Phi(\bar{x}))) = \mathcal{Y}.$$

Thus the constraint nondegeneracy condition associated with the minimizer $(\widehat{Z}, \hat{w})$ of (21) has the form

$$\begin{bmatrix} \mathcal{A} & B^T \\ \mathcal{I} & 0 \\ 0 & \mathcal{I} \end{bmatrix} \begin{pmatrix} \Re^{m\times n} \\ \Re^{n_1+n_2} \end{pmatrix} + \begin{bmatrix} \{0\}^p \\ \mathrm{lin}(T_{\mathcal{B}}(\widehat{Z})) \\ \mathrm{lin}(T_{\mathcal{Q}^*}(\hat{w})) \end{bmatrix} = \begin{bmatrix} \Re^p \\ \Re^{m\times n} \\ \Re^{n_1+n_2} \end{bmatrix}, \tag{40}$$

or equivalently,

$$\mathcal{A}\mathrm{lin}(T_{\mathcal{B}}(\widehat{Z})) + B^T \mathrm{lin}(T_{\mathcal{Q}^*}(\hat{w})) = \Re^p. \tag{41}$$

Proposition 4.2 *Let $(\widehat{Z}, \hat{w})$ be the unique solution pair of (21). Let $\widehat{Z}$ have the following SVD:*

$$\widehat{Z} = U[\mathrm{Diag}(\sigma(\widehat{Z}))\ 0]V^T = [U_1\ U_2][\mathrm{Diag}(\sigma(\widehat{Z}))\ 0][V_1\ V_2]^T,$$

where $\sigma_1(\widehat{Z}) \geq \cdots \geq \sigma_r(\widehat{Z}) > 0 = \sigma_{r+1}(\widehat{Z}) = \cdots = \sigma_m(\widehat{Z})$, and $U_1 \in \Re^{m\times r}$, $U_2 \in \Re^{m\times(m-r)}$, $V_1 \in \Re^{n\times r}$, $V_2 \in \Re^{n\times(n-r)}$. Define the following two index sets κ_1 and κ_2 by

$$\kappa_1 := \{1, 2, \ldots, n_1\} \cup \{i \,|\, \hat{w}_i > 0,\ n_1 + 1 \leq i \leq n_1 + n_2\},$$
$$\kappa_2 := \{i \,|\, \hat{w}_i = 0,\ n_1 + 1 \leq i \leq n_1 + n_2\}.$$

Then it holds that:

(i) *if $\|\widehat{Z}\|_* < 1$, the constraint nondegeneracy holds at $(\widehat{Z}, \hat{w})$ if and only if*

$$B_{\kappa_1} y = 0,\ \mathcal{A}^* y = 0 \implies y = 0. \tag{42}$$

(ii) *if $\|\widehat{Z}\|_* = 1$, the constraint nondegeneracy holds at $(\widehat{Z}, \hat{w})$ if and only if, for any given $k \in \Re$,*

$$\begin{cases} B_{\kappa_1} y = 0,\ (U_1)^T(\mathcal{A}^* y)V_1 = kI_r, \\ (U_1)^T(\mathcal{A}^* y)V_2 = 0,\ (U_2)^T(\mathcal{A}^* y)V_1 = 0 \end{cases} \implies y = 0. \tag{43}$$

Proof (i) Under the condition that $\|\widehat{Z}\|_* < 1$, it is easy to see that

$$\mathrm{lin}(T_{\mathcal{B}}(\widehat{Z})) = \Re^{m\times n} \quad \text{and} \quad \mathrm{lin}(T_{\mathcal{Q}^*}(\hat{w})) = \begin{bmatrix} \Re^{|\kappa_1|} \\ \{0\}^{|\kappa_2|} \end{bmatrix}.$$

Thus the constraint nondegeneracy condition (41) is reduced to

$$\mathcal{A}\Re^{m\times n} + B_{\kappa_1}^T \Re^{|\kappa_1|} = \Re^p, \tag{44}$$

which is equivalent to (42).
(ii) Since

$$\mathrm{lin}(T_{\mathcal{Q}^*}(\hat{w})) = \begin{bmatrix} \Re^{|\kappa_1|} \\ \{0\}^{|\kappa_2|} \end{bmatrix},$$

the constraint nondegeneracy condition (41) is reduced to

$$\mathcal{A}\mathrm{lin}(T_{\mathcal{B}}(\widehat{Z})) + B_{\kappa_1}^T \Re^{|\kappa_1|} = \Re^p, \tag{45}$$

which is equivalent to

$$B_{\kappa_1} y = 0,\ \mathcal{A}^* y \in \mathrm{lin}(T_{\mathcal{B}}(\widehat{Z}))^{\perp} \implies y = 0. \tag{46}$$

From [6, Proposition 2.3.6 and Theorem 2.4.9], we have that

$$T_{\mathcal{B}}(\widehat{Z}) = \{H \in \Re^{m\times n} \mid \|\cdot\|'_*(\widehat{Z}; H) \le 0\},$$

where $\|\cdot\|'_*(\widehat{Z}; H)$ is the directional derivative of the nuclear norm function at $\widehat{Z}$ along the direction H. Moreover, by [21, p. 61], it holds that

$$\|\cdot\|'_*(\widehat{Z}; H) = \mathrm{Tr}((U_1)^T H V_1) + \|(U_2)^T H V_2\|_*$$

and then we have

$$T_{\mathcal{B}}(\widehat{Z}) = \{H \in \Re^{m\times n} \mid \mathrm{Tr}((U_1)^T H V_1) + \|(U_2)^T H V_2\|_* \le 0\}.$$

It therefore holds that

$$\begin{aligned}\mathrm{lin}(T_{\mathcal{B}}(\widehat{Z})) &= T_{\mathcal{B}}(\widehat{Z}) \cap \big(-T_{\mathcal{B}}(\widehat{Z})\big)\\ &= \{H \in \Re^{m\times n} \mid \mathrm{Tr}((U_1)^T H V_1) = 0,\ (U_2)^T H V_2 = 0\}\\ &= \{H \in \Re^{m\times n} \mid H V_1 \in (U_1)^{\perp},\ (U_2)^T H V_2 = 0\}.\end{aligned}$$

Note that for any $Y \in \mathrm{lin}(T_{\mathcal{B}}(\widehat{Z}))^{\perp}$ if and only if

$$\langle U^T Y V,\ U^T H V\rangle = \langle Y,\ H\rangle = 0,$$

for any H satisfies the conditions $H V_1 \in (U_1)^{\perp}$ and $(U_2)^T H V_2 = 0$. Then, we know

$$\begin{aligned}\mathrm{lin}(T_{\mathcal{B}}(\widehat{Z}))^{\perp} &= \{Y \in \Re^{m\times n} \mid \exists k \in \Re,\ (U_1)^T Y V_1\\ &\quad = k I_r, (U_1)^T Y V_2 = 0, (U_2)^T Y V_1 = 0\},\end{aligned}$$

which, together with (46), completes the proof. □

With the above proposition, we next establish a result which exploits the close relationship between the constraint nondegeneracy of the optimal solution of (21) and the negative definiteness of the elements of $\hat{\partial}^2\theta_k$.

Proposition 4.3 *Suppose that the problem* (21) *satisfies the Slater condition* (20). *Let* $(\widehat{Z}, \hat{w})$ *and* $\hat{y}$ *denote respectively the optimal solutions of* (21) *and* (22), *and* $\widehat{Y} := Z^k - \lambda_k(\mathcal{A}^*\hat{y} - A_0)$. *Then the following conditions are equivalent:*

(i) *The constraint nondegeneracy condition* (41) *holds at* $(\widehat{Z}, \hat{w})$.
(ii) *Every element in* $\hat{\partial}^2\theta_k(\hat{y})$ *is symmetric and negative definite.*
(iii) *The operator*

$$\mathcal{V}_0 = -\lambda_k\left(\mathcal{A}\mathcal{V}_1^1\mathcal{A}^* + B^T\mathcal{V}_2^0 B\right)$$

is symmetric and negative definite, where $\mathcal{V}_1^1$ *is the same as* $\mathcal{V}_1^0$ *in* (38) *except when in the case of* $\|\widehat{Y}\|_* = 1$, *the operator is defined by* (33) *with* $N = \mathrm{rank}(\widehat{Y})$.

Proof Assume the SVD of $\widehat{Z}$ and the index sets κ_1, κ_2 are given in Proposition 4.2.

"(i) $\Rightarrow$ (ii)". Let $\mathcal{V}$ be an arbitrary element of $\hat{\partial}^2\theta_k(\hat{y})$. Then there exist $\mathcal{V}_1$ and $\mathcal{V}_2$ in $\partial\Pi_{\mathcal{B}}(\widehat{Y})$ and $\partial\Pi_{\mathcal{Q}^*}[w^k - \lambda_k(B\hat{y} - b)]$, respectively, such that

$$\mathcal{V} = -\lambda_k[\mathcal{A}\mathcal{V}_1\mathcal{A}^* + B^T\mathcal{V}_2 B].$$

Recall [29, Proposition 1]: for any $x \in \mathcal{X}$ and $V \in \partial\Pi_C(x)$, V is self-adjoint and it holds

$$\langle Vd,\ d - Vd\rangle \geq 0,\ \forall d \in \mathcal{X}, \tag{47}$$

where $C \subset \mathcal{X}$ is a closed convex set. Therefore, $\mathcal{V}_1$ and $\mathcal{V}_2$ are self-adjoint, and so is $\mathcal{V}$. Moreover, we know from (47) that for any $h \in \Re^p$,

$$\begin{aligned}\langle h,\ \mathcal{V}h\rangle &= -\lambda_k\langle h,\ \mathcal{A}\mathcal{V}_1\mathcal{A}^*h\rangle - \lambda_k\langle h,\ B^T\mathcal{V}_2 Bh\rangle\\ &= -\lambda_k\langle \mathcal{A}^*h,\ \mathcal{V}_1\mathcal{A}^*h\rangle - \lambda_k\langle Bh,\ \mathcal{V}_2 Bh\rangle\\ &\leq -\lambda_k\langle \mathcal{V}_1\mathcal{A}^*h,\ \mathcal{V}_1\mathcal{A}^*h\rangle - \lambda_k\langle \mathcal{V}_2 Bh,\ \mathcal{V}_2 Bh\rangle\ \leq\ 0,\end{aligned}$$

which implies that $\mathcal{V}$ is negative semidefinite. To proceed the proof of this part, we next show that $\mathcal{V}$ is negative definite for any $\mathcal{V}_1 \in \partial_B\Pi_{\mathcal{B}}(\widehat{Y})$ and $\mathcal{V}_2 \in \partial\Pi_{\mathcal{Q}^*}[w^k - \lambda_k(B\hat{y} - b)]$. Consider the following linear system

$$\mathcal{V}h = 0, \quad \text{or equivalently,} \quad \mathcal{V}_1\mathcal{A}^*h = 0,\ \mathcal{V}_2 Bh = 0. \tag{48}$$

To prove $h = 0$, we consider the following two cases.

Case 1: $\|\widehat{Y}\|_* < 1$. In this case, since $\widehat{Z} = \Pi_{\mathcal{B}}(\widehat{Y})$, we have $\|\widehat{Z}\|_* < 1$ and $\mathcal{V}_1 = \mathcal{I}$. Then, it follows from (48) that

$$\mathcal{A}^*h = 0,\ B_{\kappa_1}h = 0, \tag{49}$$

which, together with the constraint nondegeneracy assumption (42), implies $h = 0$.

Case 2: $\|\widehat{Y}\|_* \geq 1$ and thus $\|\widehat{Z}\|_* = 1$. We first show the nonsingularity of $\mathcal{V}$ for the choice that $\mathcal{V}_1 = \mathcal{I}$. In this situation, (49) still holds and hence by taking $k = 0$ in (43), we know that $\mathcal{V}$ is nonsingular and hence negative definite. Next, we turn to the case in which $\mathcal{V}_1$ is another element selected from $\partial_B\Pi_{\mathcal{B}}(\widehat{Y})$. We consider two sub-cases.

Case 2.1: $\|\widehat{Y}\|_* = 1$. Let $H = \mathcal{A}^*h$. In view of the analysis in the previous subsection, we know from $\mathcal{V}_1 H = 0$ that

$$\begin{aligned}0 = U^\infty\left[\begin{bmatrix}\mathbf{1}_{N\times N} & \mathbf{1}_{r\times(m-N)} & \\ \mathbf{1}_{(m-N)\times r} & \Omega^\infty_{\alpha_2\alpha_3} & \\ & (\Omega^\infty_{\alpha_2\alpha_3})^T & 0\end{bmatrix}\circ S(\widetilde{H}_1) \quad \begin{bmatrix}\mathbf{1}_{r\times(n-m)}\\ \Upsilon^\infty_{\alpha_2}\mathbf{1}^T_{n-m}\\ 0\end{bmatrix}\circ\widetilde{H}_2\right](V^\infty)^T\\ + U^\infty\left(\begin{bmatrix}\mathbf{1}_{r\times r} & \mathbf{1}_{r\times(m-r)} & \\ \mathbf{1}_{(m-r)\times r} & \Gamma^\infty_{\alpha_2\alpha_2} & \Gamma^\infty_{\alpha_2\alpha_3}\\ & (\Gamma^\infty_{\alpha_2\alpha_3})^T & 0\end{bmatrix}\circ T(\widetilde{H}_1) - \frac{\mathrm{Tr}(\widetilde{H}_{11})}{N}\begin{bmatrix}I_N & 0\\ 0 & 0\end{bmatrix}\right)\left(V^\infty_1\right)^T,\end{aligned}$$

where $V_1^\infty \in \Re^{m\times m}$, $V_2^\infty \in \Re^{m\times(n-m)}$ and $V^\infty := [V_1^\infty\ V_2^\infty]$, the index sets $\alpha_1, \alpha_2, \alpha_3$ and α_4 are defined as in (32). This implies that

$$(U_{\alpha_1}^\infty)^T(H)V_{\alpha_1}^\infty = \frac{\mathrm{Tr}(\widetilde{H}_{11})}{N} I_r, \quad (U_{\alpha_1}^\infty)^T(H)V_{\alpha_2\cup\alpha_3\cup\alpha_4}^\infty = 0, \quad (U_{\alpha_2\cup\alpha_3}^\infty)^T(H)V_{\alpha_1}^\infty = 0. \tag{50}$$

By Proposition 2.4, there exist orthogonal matrices $Q \in \Re^{r\times r}$, $Q' \in \Re^{(m-r)\times(m-r)}$ and $Q'' \in \Re^{(n-r)\times(n-r)}$ such that

$$U_{\alpha_1}^\infty = U_{\alpha_1} Q, \quad U_{\alpha_2\cup\alpha_3}^\infty = U_{\alpha_2\cup\alpha_3} Q', \quad V_{\alpha_1}^\infty = Q V_{\alpha_1}, \quad V_{\alpha_2\cup\alpha_3\cup\alpha_4}^\infty = Q'' V_{\alpha_2\cup\alpha_3\cup\alpha_4}. \tag{51}$$

Moreover, we know from $\mathcal{V}_2 h = 0$ that

$$B_{\kappa_1} h = 0. \tag{52}$$

Combining (43), (50), (51) with (52), we deduce $h = 0$ and hence $\mathcal{V}$ is nonsingular and hence negative definite.

Case 2.2: $\|\widehat{Y}\|_* > 1$. The proof of the negative definiteness of $\mathcal{V}$ is similar to that of Case 2.1, with the equality (34) replacing (33).

By taking the convex hull of $\partial_B \Pi_{\mathcal{B}}(\widehat{Y})$, we complete the proof of the first part.

"(ii) $\Rightarrow$ (iii)". This is trivial since $\mathcal{V}_0 \in \hat{\partial}^2\theta_k(\hat{y})$.

"(iii) $\Rightarrow$ (i)". Assume the contrary that the constraint nondegeneracy condition fails to hold at $(\widehat{Z}, \hat{w})$. Again, we consider two cases.

Case 1: $\|\widehat{Y}\|_* < 1$ and hence $\|\widehat{Z}\|_* < 1$. By assumption, there exists a $z \neq 0$ such that

$$\mathcal{A}^* z = 0, \quad B_{\kappa_1} z = 0.$$

This means that $\mathcal{V}_0$ is singular, which contradicts to (iii).

Case 2: $\|\widehat{Y}\|_* \geq 1$ and hence $\|\widehat{Z}\|_* = 1$. By assumption, there exist $k \in \Re$ and $z \neq 0$ such that

$$B_{\kappa_1} z = 0, \quad (U_1)^T(\mathcal{A}^* z)V_1 = k I_r, \quad (U_1)^T(\mathcal{A}^* z)V_2 = 0, \quad (U_2)^T(\mathcal{A}^* z)V_1 = 0. \tag{53}$$

Substituting the equalities above into (38) and (39), we deduce that $\mathcal{V}_0 z = 0$. This contradicts to the statement (iii). The proof is completed. □

4.3 A semismooth Newton-CG algorithm

In this subsection, we briefly describe the semismooth Newton-CG algorithm for solving (22). The basic template of the algorithm is given as follows. For simplicity, we drop the outer iteration index k.

Algorithm 4.1 (A Semismooth Newton-CG Method)

Step 0. *Given* $\varsigma \in (0, 0.5)$, $\delta_1, \delta_2, \bar{\eta}, \rho \in (0, 1)$ *and* $\tau \in (0, 1]$. *Choose* $y^0 \in \Re^p$.

Step 1. *For* $j = 0, 1, 2, \ldots,$

Step 1.1. *Apply the preconditioned conjugate gradient (*PCG*) method to find an approximation solution* d^j *to*

$$(\mathcal{V}_j - \epsilon_j I)d = -\nabla\theta_k(y^j) \tag{54}$$

satisfying the residual condition

$$\|(\mathcal{V}_j - \epsilon_j I)d^j + \nabla\theta_k(y^j)\| \le \eta_j := \min\{\bar{\eta}, \|\nabla\theta_k(y^j)\|^{1+\tau}\}. \tag{55}$$

Here, $\mathcal{V}_j$ *is an element of* $\hat{\partial}^2\theta_k(y^j)$ *and* $\epsilon_j = \delta_1 \min\{\delta_2, \|\nabla\theta_k(y^j)\|\}$.

Step 1.2. *Let* l_j *be the smallest nonnegative integer* l *satisfying*

$$\theta_k(y^j + \rho^l d^j) - \theta_k(y^j) \ge \varsigma\rho^l \langle \nabla\theta_k(y^j), d^j\rangle.$$

Set $\alpha_j := \rho^{l_j}$ *and* $y^{j+1} := y^j + \alpha_j d^j$.

From the structures of $\mathcal{V}_1^0$ and $\mathcal{V}_2^0$, we know that $\mathcal{V}_j$ is always negative semidefinite. Hence $\mathcal{V}_j - \epsilon_j I$ is always negative definite as long as $\nabla_y\theta_k(y^j) \neq 0$. So, it is reasonable for us to apply the PCG method to solve (54). Furthermore, by noting the strong semismoothness of $\Pi_{\mathcal{B}}(\cdot)$ and $\Pi_{\mathcal{Q}^*}(\cdot)$, and using the proof similar to [47, Theorem 3.4], we can easily derive the following convergence results for Algorithm 4.1.

Theorem 4.1 *Suppose that the Slater condition holds for* (21). *Then the semismooth Newton-CG algorithm* 4.1 *is well defined and any accumulation point* $\hat{y}$ *of* $\{y^j\}$ *generated by Algorithm* 4.1 *is an optimal solution to the inner subproblem* (22).

Theorem 4.2 *Assume that the Slater condition holds for* (21). *Let* $\hat{y}$ *be an accumulation point of the infinite sequence* $\{y^j\}$ *generated by the semismooth Newton-CG algorithm for solving* (22). *Suppose that at each step* $j \ge 0$, *the residual condition (55) is satisfied. Assume that the constraint nondegeneracy condition* (41) *holds at* $(\hat{Z}, \hat{w})$, *where* $\hat{Z} := \Pi_{\mathcal{B}}(Z^k - \lambda_k(\mathcal{A}^*(\hat{y}) - A_0))$ *and* $\hat{w} := \Pi_{\mathcal{Q}^*}(w^k - \lambda_k(B\hat{y} - b))$. *Then the whole sequence* $\{y^j\}$ *converges to* $\hat{y}$ *and*

$$\|y^{j+1} - \hat{y}\| = O\left(\|y^j - \hat{y}\|^{1+\tau}\right).$$

4.4 Numerical issues

In applying the semismooth Newton-CG method to solve the inner problem (22), the most expensive step is to compute the Newton direction from the linear equation

involving the operator $\mathcal{V}_j$ in (54). As is well known, the basic operation in implementing the PCG method is to calculate the multiplication $\mathcal{V}_j y$ for any given $y \in \Re^p$. From the analysis in Sect. 4.1, the computation appears to require the full SVD of an $m \times n$ matrix. For a problem in which m is moderate but n is large, the full SVD computation would be expensive and huge memory space is also needed to store the large and dense matrix V. To alleviate this difficulty, the authors in [21] suggested computing the full SVD indirectly via a reduced SVD and a QR factorization by Householder transformations. But in fact, we can completely avoid the computation of V_2 by carefully analyzing the structure of (38) as shown next. The part $\mathrm{Diag}\begin{bmatrix} \Upsilon_{\gamma_1} \\ 0 \end{bmatrix} \widetilde{H}_2 V_2^T$ in (38) is given as follows:

$$\begin{aligned} \mathrm{Diag}\begin{bmatrix} \Upsilon_{\gamma_1} \\ 0 \end{bmatrix} \widetilde{H}_2 V_2^T &= \begin{bmatrix} \mathrm{Diag}(\Upsilon_{\gamma_1}) U_{\gamma_1}^T H (V_2 V_2^T) \\ 0 \end{bmatrix} \\ &= \begin{bmatrix} \mathrm{Diag}(\Upsilon_{\gamma_1}) U_{\gamma_1}^T H (I - V_1 V_1^T) \\ 0 \end{bmatrix}. \end{aligned} \tag{56}$$

From (56), it is clear that V_2 can be avoided when evaluating $\mathcal{V}_0^1 H$ in (38). Given the economical SVD of $Z^k - \lambda_k(\mathcal{A}^* y - A_0)$, we can compute $\mathcal{V}_0^1 H$ in at most $k_1(\sigma)(18mn + 4m^2)$ flops for any given $H \in \Re^{m \times n}$. The above complexity shows that our algorithm is able to utilize any low rank or flat rectangular structure of a matrix to reduce the computational cost.

Next, we introduce two diagonal preconditioners which may help to accelerate the convergence of the CG method applied to solve the linear system (54). Let $\mathbf{A}$ and $\mathbf{V}$ be the matrix representations of the linear mappings $\mathcal{A}$ and $\mathcal{V}_1^0$, respectively. Then the coefficient matrix in (54) has the following form

$$W = -\lambda \mathbf{A}\mathbf{V}\mathbf{A}^T - \lambda B_{\mathcal{J}_1}^T B_{\mathcal{J}_1} - \epsilon I.$$

Note that we have omitted the iteration index for brevity. Let the standard basis in $\Re^{m \times n}$ be $\{E_{ij} \in \Re^{m \times n} : 1 \le i \le m,\ 1 \le j \le n\}$, where E_{ij} is the matrix whose (i, j)-th entry is one and zero otherwise. The diagonal element of $\mathcal{V}_1^0$ with respect to the standard basis is given by

$$\begin{aligned} \mathbf{V}_{(i,j),(i,j)} = &\left((U \circ U)\Lambda^\infty (V^T \circ V^T)\right)_{ij} - \frac{1}{k_1(\sigma)} \left((U_1' V_1'^T) \circ (U_1' V_1'^T)\right)_{ij} \\ &+ \frac{1}{2} \left\langle H_{ij} \circ H_{ij}^T,\ \Omega^\infty - \Gamma^\infty \right\rangle, \end{aligned} \tag{57}$$

where

$$\Lambda^\infty = \begin{bmatrix} \frac{1}{2}(\Omega^\infty + \Gamma^\infty) & \left(\Upsilon^\infty \mathbf{1}^T\right)_{k_1(\sigma) \times (n-m)} \\ \mathbf{0}_{(m-k_1(\sigma)) \times (n-m)} \end{bmatrix}, \quad H_{ij} = U^T E_{ij} V_1 \in \Re^{m \times m}$$

and U_1' and V_1' are the matrices formed by the first $k_1(\sigma)$ columns of U and V, respectively. To avoid excessive computational cost, one would naturally drop the last

term in (57). However, it turns out that the first two terms often do not provide a good approximation of $\mathbf{V}_{(i,j),(i,j)}$. In practice, we find that replacing Λ^∞ by $\widehat{\Lambda}^\infty$ (which is the same as Λ^∞ with Γ^∞ replaced by Ω^∞) in the first term often gives a better approximation, i.e., we consider

$$\mathbf{D}_{(i,j),(i,j)} = \left((U \circ U)\widehat{\Lambda}^\infty \left(V^T \circ V^T\right)\right)_{ij} - \frac{1}{k_1(\sigma)} \left(\left(U_1' V_1'^T\right) \circ \left(U_1' V_1'^T\right)\right)_{ij} \quad (58)$$

as an approximation of (57). Thus we propose the following diagonal preconditioner for the coefficient matrix:

$$M = \lambda \mathrm{Diag}\left(\mathbf{A}\mathbf{D}\mathbf{A}^T + B_{\mathcal{J}_1}^T B_{\mathcal{J}_1}\right) + \epsilon I. \quad (59)$$

Clearly, to use the preconditioner above, we need the explicit form of V, which may lead to memory difficulty when n is large. Thus when n is too large for V to be stored explicitly, we just use the following simple diagonal preconditioner

$$M' = \lambda \mathrm{Diag}(\mathbf{A}\mathbf{A}^T + B_{\mathcal{J}_1}^T B_{\mathcal{J}_1}) + \epsilon I. \quad (60)$$

5 Numerical results

In this section, we will apply our algorithm SNDPPA (semismooth Newton-CG based dual PPA) to solve four different types of the MNA problems. For the purpose of numerical comparison, we will also report the performance of an alternating direction method of multipliers (ADMM) for solving the same set of problems. All the codes are written in MATLAB 7.11 and run an Intel Xeon 2.80 GHz (quadcore) PC with 24 GB memory.

The ADMM was first introduced in [12,14]. Since then a number of variants have been studied theoretically and employed in many applications due to its great simplicity, see, e.g., [4,10,13,17,45]. Here we shall employ the classical ADMM [12,14] to solve the MNA problems. Note that problem (1) can be expressed in the following equivalent form:

$$\min\left\{\|X\|_2 \mid \mathcal{A}^* y + X = A_0,\ By - b = z,\ z \in \mathcal{Q}\right\}. \quad (61)$$

The augmented Lagrangian function associated with (61) is given by

$$\begin{aligned}\mathcal{L}_\beta(y, X, z; Z, w) := \|X\|_2 - \langle Z, \mathcal{A}^* y + X - A_0\rangle - \langle w, By - b - z\rangle \\ + \frac{\beta}{2}\|\mathcal{A}^* y + X - A_0\|^2 + \frac{\beta}{2}\|By - b - z\|^2, \end{aligned} \quad (62)$$

where Z and w are Lagrangian multipliers, and $\beta > 0$ is the penalty parameter. Given $X^0, Z^0 \in \Re^{m\times n}$, $z^0, w^0 \in \Re^{n_1+n_2}$, and $\beta_0 > 0$, the k-th iteration of the ADMM for

(61) is given as follows:

$$\begin{aligned} y^{k+1} &= \arg\min\{\mathcal{L}_{\beta_k}(y, X^k, z^k; Z^k, w^k) \mid y \in \Re^p\},\\ (X^{k+1}, z^{k+1}) &= \arg\min\{\mathcal{L}_{\beta_k}(y^{k+1}, X, z; Z^k, w^k) \mid (X, z) \in \Re^{m\times n} \times \mathcal{Q}\},\\ Z^{k+1} &= Z^k - \varrho\beta_k(\mathcal{A}^* y^{k+1} + X^{k+1} - A_0),\\ w^{k+1} &= w^k - \varrho\beta_k(By^{k+1} - b - z^{k+1}), \end{aligned}$$

where $\varrho \in (0, \frac{1+\sqrt{5}}{2})$. It is easy to see that the minimizer y^{k+1} is the solution of the following linear system of equations:

$$(\mathcal{A}\mathcal{A}^* + B^T B)y^{k+1} = \mathcal{A}(A_0 - X^k + Z^k/\beta_k) + B^T(b + z^k + w^k/\beta_k). \quad (63)$$

Since $\mathcal{L}_{\beta_k}(y^{k+1}, X, z; Z^k, w^k)$ is separable in X and z, simple algebraic manipulations then give

$$\begin{aligned} X^{k+1} &= A_0 - \mathcal{A}^* y^{k+1} + Z^k/\beta_k - \Pi_{\mathcal{B}_{1/\beta_k}}(A_0 - \mathcal{A}^* y^{k+1} + Z^k/\beta_k),\\ z^{k+1} &= \Pi_{\mathcal{Q}}(By^k - b - w^k/\beta_k). \end{aligned}$$

As analyzed before, X^{k+1} can be computed analytically, and z^{k+1} is just a simple projection over $\mathcal{Q}$.

We use R_p, R_d and gap to denote, respectively, the primal infeasibility, dual infeasibility and primal-dual relative gap, namely

$$\begin{aligned} R_p &= \frac{\|[\mathcal{A}^* y + X - A_0;\, \Pi_{\mathcal{Q}^*}(b - By)]\|}{1 + \|[A_0; b]\|},\ R_d = \frac{\|\mathcal{A}Z + B^T w\|}{1 + \|[\mathcal{A}; B^T]\|},\\ \text{gap} &= \frac{|\text{pobj} - \text{dobj}|}{1 + |\text{pobj}| + |\text{dobj}|}, \end{aligned}$$

where pobj and dobj are the primal and dual objective values, respectively.

In our numerical experiments, we start the ADMM from the point $(X, y, z, Z, w) = (0, 0, 0, 0, 0)$ and terminate it when

$$\max\{R_p, R_d\} \le 10^{-6} \quad (64)$$

or the maximum number of iterations exceeds 2,000. The penalty parameter β in the ADMM is adjusted dynamically: starting from the initial value of 10, we adjust β at every fifth step as follows such that β_k cannot go to extremely large or small, and the primal and dual infeasibilities are well balanced:

$$\beta_{k+1} = \begin{cases} \min(10^3, 2\beta_k), & \text{if } R_p^k/R_d^k > 10,\\ \max(10^{-2}, 0.5\beta_k), & \text{if } R_p^k/R_d^k < 0.1,\\ \beta_k, & \text{otherwise.} \end{cases} \quad (65)$$

For the SNDPPA, we use the ADMM to generate an initial point which satisfies that $\max\{R_p, R_d\} \le 5 \times 10^{-3}$. The total number of the ADMM steps for this initialization

is capped at 50. We terminate the SNDPPA when the condition (64) is met. For each PPA iteration, we cap the number of Newton-CG iterations for solving an inner subproblem to 40. In solving the linear system associated with the Newton direction, the maximal number of PCG steps is set as 500. As the parameter λ plays a critical role in the convergence speed of a PPA-based algorithm, we need to handle it with care. In our implementation, the parameter λ is initialized as 10 and increased according to the following empirical rule in order to accelerate the convergence of the outer loop of the PPA and also balance the primal and dual infeasibilities:

$$\lambda_{k+1} = \begin{cases} 3\lambda_k, & R_p^{k+1}/R_p^k > 0.5 \text{ and } R_p^{k+1} > 10^{-4}, \\ 2\lambda_k, & R_p^{k+1}/R_p^k > 0.5 \text{ and } R_p^{k+1} < 10^{-4}, \\ \lambda_k, & \text{otherwise}, \end{cases} \tag{66}$$

where λ_k denotes the penalty parameter value at the kth PPA iteration.

5.1 Random matrix norm approximation

We first consider randomly generated MNA problems with/without constraints. In the experiments, the matrices $A_0, A_1, \ldots, A_p$ are generated independently from the multivariate uniform distribution on $[0, 1]^{m\times n}$.

In Table 1, we report the numerical performance of the SNDPPA and the ADMM for solving different random matrix approximation instances without constraints. The number of outer iterations (iter), primal infeasibility (R_p), dual infeasibility (R_d), primal objective value (pobj), relative gap (gap), and the CPU time (time) taken are listed in the table. To better understand the performance of the SNDPPA, we also report the number of Newton systems solved (itersub) and the average number PCG steps (pcg) taken to solve each of the systems.

Table 1 Results for unconstrained random matrix norm approximation problems

$p \mid m \mid n$	Algo.	it (itersub|pcg)	pobj|gap	$R_p \mid R_d$	Time
300|300|300	PPA	14 (15|4.1)	9.44515934 0|2.8–6	4.4–7|3.2–8	8.3
	ADMM	300	9.44520938 0|4.2–6	9.7–7|2.5–7	26.0
500|500|500	PPA	17 (18|4.2)	1.22905150 1|3.7–6	4.3–7|2.3–8	42.4
	ADMM	619	1.22905586 1|1.9–5	6.8–7|9.9–7	232.4
100|100|3,000	PPA	16 (18|4.0)	1.83807818 1|8.3–6	9.4–7|4.1–8	13.1
	ADMM	821	1.83807914 1|6.5–6	9.9–7|1.7–7	100.5
100|100|5,000	PPA	16 (17|4.0)	2.31039070 1|5.6–6	9.3–7|4.0–8	20.8
	ADMM	443	2.31040515 1|3.2–6	9.9–7|9.7–7	93.1
100|100|10,000	PPA	18 (19|4.0)	3.16771120 1|2.8–6	5.4–7|1.1–7	46.2
	ADMM	740	3.16774836 1|7.4–6	9.5–7|9.9–7	300.3
100|100|20,000	PPA	16 (17|4.0)	4.37704442 1|1.2–7	7.3–7|2.7–9	97.7
	ADMM	654	4.37704413 1|9.9–6	4.6–7|9.9–7	668.2

As can be observed in Table 1, both the ADMM and the SNDPPA are able to solve the unconstrained random matrix approximation problems to relatively high accuracy. The SNDPPA substantially outperforms the ADMM in terms of the CPU time taken to solve the problems. For example, the ADMM takes about 35 min to solve the last instance while our SNDPPA solves it in 4.5 min and with a better accuracy in dual infeasibility and objective gap. Note that the random matrix approximation problems can also be solved by the interior point package SDPT3 [38] via the SDP reformulation (6). However, the interior solver may encounter computational and memory difficulties when solving large problems. For example, SDPT3 takes about 5 min to solve the problem with $(m, n, p) = (300, 300, 300)$ while our SNDPPA solves it in only 8 s. For the larger instance with $(m, n, p) = (100, 5{,}000, 100)$, SDPT3 is not able to solve the problem due to excessive computer memory required. It is worth noting that for the instances with $(p, m) = (100, 100)$, the CPU time taken by each iteration of the SNDPPA and the ADMM increases almost linearly with n. But for a solver (say the algorithm in [47]) that attempts to solve (1) via the SDP reformulation (6), the cost per iteration would grow at least quadratically in n. This observation is consistent with the fact mentioned in the previous section that our SNDPPA is capable of exploiting the flat rectangular structure of the matrices involved.

Next, we test our SNDPPA on the MNA problems with constraints. A simple example is to find a convex combination of given matrices $A_0, A_1, \ldots, A_p$ having the minimal spectral norm, i.e.,

$$\min \left\{ \|A_0 - \mathcal{A}^* y\|_2 \mid \sum\nolimits_{i=1}^{p} y_i = 1, \quad y \geq 0 \right\}. \tag{67}$$

In what follows, we investigate the performance of the SNDPPA and the ADMM applied to (67) where the matrices $A_1, \ldots, A_p$ are randomly generated as before. Table 2 lists the numerical results obtained by the SNDPPA and the ADMM. For this collection of problems, we can easily see the superiority of the SNDPPA over the first order algorithm ADMM. While our SNDPPA solves all the tested instances to the accuracy of 10^{-6} within 36 semismooth Newton-CG iterations, the ADMM fails to achieve the required accuracy even after 2,000 iterations. For the instance with $(m, n, p) = (100, 2{,}0000, 100)$, the ADMM fails to achieve the accuracy of 10^{-6} after running for 35 min while our SNDPPA is able to solve the problem in about 2 min. As one may deduce from the results in Table 2, the ADMM may encounter both computational and accuracy difficulties even only simple constraints are imposed on y.

5.2 Chebyshev polynomials of matrices

In this subsection, we apply the proposed SNDPPA to compute the Chebyshev polynomials of a given matrix $A \in \Re^{n\times n}$. Since the power basis $I, A, \ldots, A^t$ is usually highly ill conditioned, in [40] the authors suggested replacing this basis by a better-conditioned alternative $Q_1, Q_2, \ldots, Q_{t+1}$ and consider the resulting problem

$$\min_{y\in\Re^t} \|Q_{t+1} - \sum\nolimits_{i=1}^{t} y_i Q_i\|_2. \tag{68}$$

Table 2 Results for the matrix norm approximation problem (67)

$p \mid m \mid n$	Algo.	it (itersub | pcg)	pobj | gap	$R_p \mid R_d$	Time
300|300|300	PPA	17 (31|12.8)	9.59410207 0|2.6–5	7.7–7|3.6–8	21.8
	ADMM	2,000	9.59309978 0|3.8–6	4.2–6|6.8–6	175.5
500|500|500	PPA	19 (36|14.8)	1.24537522 1|5.4–5	7.6–7|7.2–7	131.7
	ADMM	2,000	1.24556416 1|1.2–4	1.1–5|1.1–5	777.9
100|100|3,000	PPA	20 (27|5.0)	1.83873278 1|3.7–6	5.4–7|2.6–8	19.5
	ADMM	2,000	1.83863304 1|1.6–4	7.1–6|6.7–6	246.7
100|100|5,000	PPA	18 (24|5.3)	2.31091411 1|3.5–6	4.2–7|2.1–8	29.2
	ADMM	2,000	2.31077010 1|1.0–4	8.3–6|2.8–6	423.4
100|100|10,000	PPA	19 (25|4.9)	3.16803834 1|6.1–6	8.7–7|2.7–8	59.4
	ADMM	2,000	3.16798808 1|1.0–5	3.0–6|9.9–7	885.2
100|100|20,000	PPA	21 (25|4.0)	4.37736739 1|2.3–5	7.8–7|6.6–7	125.2
	ADMM	2,000	4.37716525 1|6.2–5	4.5–6|2.1–6	2,139.9

From the solution of (68), one can easily compute the coefficients of the Chebyshev polynomials via Theorem 2 in [40]. In our experiments, the test examples are taken from Section 6 in [40] and $Q_1, Q_2, \ldots, Q_{t+1}$ is the orthogonal basis corresponding to the power basis of A.

Table 3 shows that for most of the test instances, both the SNDPPA and the ADMM are capable of achieving the accuracy of less than 10^{-6}. However, for examples such as Bidiag ($n = 1{,}000$) and Diag ($n = 500$), the ADMM fails to solve them within 2,000 iterations while the SNDPPA succeeds in achieving the required accuracy for all the instances. This illustrates that our SNDPPA performs much more robustly than the ADMM. Moreover, the SNDPPA is much more efficient than the ADMM in terms of computing time. Specifically, the former is about 5–10 times faster than the latter. Also, the performance of our SNDPPA is superior to the interior-point solver SDPT3 [38] applied to the SDP reformulation in terms of CPU time. For example, SDPT3 takes about 2 min to achieve a solution with an accuracy of 10^{-6} for the problem Rand ($n = 500, t = 50$) while our SNDPPA solves it in 13 s. This is not surprising since for most instances, the SNDPPA takes less than 20 semismooth Newton-CG iterations to generate a highly accurate solution and the average number of PCG steps needed to solve each of the Newton systems is less than 10.

5.3 FMMC/FDLA

In this subsection, we investigate the numerical performance of the two algorithms for solving the FMMC problem (4) and the FDLA problem (5). The tested graphs are taken from the sparse matrix collection [7] but some are slightly modified to make them connected. The data set is available at http://www2.research.att.com/~gyifanhu/GALLERY/GRAPHS/search.html.

Table 3 Chebyshev polynomials of matrices

Problem	Algo.	$n \mid t$	it (itersub | pcg)	pobj | gap	$R_p \mid R_d$	Time
Rand	PPA	500|50	16 (18|7.9)	2.19977218-1|3.7–7	3.5–7|1.8–7	12.4
		1,000|100	14 (15|9.2)	1.84595016-1|1.5–7	2.5–7|1.6–7	72.0
	ADMM	500|50	354	2.19977287-1|6.2–8	3.0–7|8.7–7	45.6
		1,000|100	661	1.84594835-1|3.1–7	1.1–7|9.1–7	518.8
Randtri	PPA	500|50	6 (9|9.5)	4.14987261-1|3.4–7	2.2–7|1.6–7	8.2
		1,000|100	9 (11|12.8)	3.56509526-1|1.5–7	1.8–7|3.6–7	60.9
	ADMM	500|50	663	4.14987166-1|1.4–6	7.1–8|8.8–7	85.1
		1,000|100	786	3.56509344-1|1.1–6	7.3–8|9.1–7	585.9
Diag	PPA	500|50	15 (26|8.9)	7.20405153-2|5.8–7	3.0–8|5.3–7	9.1
		1,000|100	13 (24|10.3)	4.85094484-2|5.1–7	3.1–7|1.3–7	49.1
	ADMM	500|50	2,000	7.20772064-2|3.0–4	3.0–5|3.6–4	94.2
		1,000|100	396	4.85093725-2|2.7–7	4.6–7|8.7–7	108.3
Bidiag	PPA	500|50	11 (41|20.8)	1.90877129-1|2.5–7	2.5–7|2.1–7	30.1
		1,000|100	17 (79|36.9)	1.38036272-1|1.3–6	4.5–7|7.6–7	386.1
	ADMM	500|50	1,482	1.90877146-1|2.2–7	2.0–7|8.8–7	162.8
		1,000|100	2,000	1.38036596-1|5.4–7	1.0–6|9.1–7	1,231.5
Ellipse	PPA	500|50	9 (14|4.1)	5.51257423-2|5.1-10	1.5–7|1.9–9	6.2
		1,000|100	13 (21|4.8)	3.90141290-2|4.2–9	3.7–7|1.2–7	53.8
	ADMM	500|50	269	5.51257424-2|4.4–7	4.7–7|8.7–7	28.3
		1,000|100	370	3.90141292-2|3.6–7	4.8–7|5.5–7	219.4
Grcar	PPA	500|50	17 (42|10.1)	7.19041068-2|2.8–7	3.8–7|2.0–7	24.5
		1,000|100	11 (25|9.6)	5.07326772-2|2.0–7	4.0–7|4.0–7	94.8
	ADMM	500|50	2,000	7.19051700-2|2.2–6	2.4–6|7.3–7	276.5
		1,000|100	865	5.07326793-2|9.0–7	2.6–7|9.1–7	656.5
Lemniscate2	PPA	500|50	17 (65|8.7)	8.09230480-2|1.7–6	2.0–7|7.3–7	37.8
		1,000|100	20 (78|19.9)	3.33334342-2|3.3–6	3.5–7|2.5–7	357.9
	ADMM	500|50	1,104	8.09229960-2|3.1–8	2.1–7|8.7–7	145.7
		1,000|100	1,154	3.33337454-2|1.8–6	4.9–7|8.6–7	818.9
Wilkinson	PPA	500|50	14 (25|5.2)	2.02888114-1|4.4–6	1.0–7|6.8–7	12.0
		1,000|100	13 (30|6.9)	1.92544544-1|7.5–7	1.6–7|2.3–7	63.8
	ADMM	500|50	859	2.02888114-1|1.2–6	5.0–7|8.5–7	85.9
		1,000|100	1,723	1.92544547-1|9.2–8	5.8–8|9.1–7	986.5
Chebyshev	PPA	500|50	9 (14|5.1)	2.24960561-1|3.5–7	2.0–7|8.1–8	8.4
		1,000|100	12 (22|6.4)	2.06618495-1|5.6–9	2.1–7|2.9–8	92.8
	ADMM	500|50	788	2.24960549-1|1.3–6	3.4–7|8.7–7	84.6
		1,000|100	2,000	2.06629894-1|2.7–4	4.8–6|2.4–5	1,142.7

Table 4 shows that our SNDPPA is able to achieve the required accuracy of less than 10^{-6} for all the test examples. However, by comparing the results for FMMC/FDLA with those for the random matrix approximation and Chebyshev polynomial problems,

Table 4 Performance of the SNDPPA and the ADMM for FMMC/FDLA problems on connected graphs

Problem	$p \mid n$	Algo.	it (itersub | pcg)	pobj | gap	$R_p \mid R_d$	Time
FDLA-Cage	2562|366	PPA	9 (11|2.0)	4.58547216-1|2.8–6	5.5–7|1.0–9	3.5
		ADMM	2,000	4.75216058-1|3.4-1	1.1–3|1.3–3	92.2
FMMC-Cage	2562|366	PPA	5 (6|2.6)	4.58545022-1|5.7–7	4.1–8|7.6-10	2.1
		ADMM	2,000	5.87384195-1|4.1-1	7.0–5|1.5–3	95.8
FDLA-Erdos981	1381|485	PPA	6 (6|3.2)	1.00000000 0|2.9-14	0.0–8|0.0-16	2.9
		ADMM	20	1.00000000 0|3.7–5	7.1–7|3.0–8	2.2
FMMC-Erdos981	1381|485	PPA	7 (10|6.0)	1.00000000 0|4.1-14	0.0–9|1.5–7	4.6
		ADMM	23	1.00000000 0|7.5–6	7.4–7|1.5–7	2.5
FDLA-G3	19176|800	PPA	11 (17|31.7)	2.40597954-1|1.6–4	6.3–7|4.3–7	48.0
		ADMM	2,000	2.41026286-1|3.9–4	1.1–4|5.3–6	756.8
FMMC-G3	19176|800	PPA	17 (27|37.8)	2.40914549-1|4.9–8	3.6–7|1.7–9	85.7
		ADMM	2,000	2.41009134-1|8.7–4	5.4–6|3.0–5	769.5
FDLA-NotreDame_yeast	2203|2114	PPA	12 (17|4.6)	1.00000000 0|6.1-12	0.0–9|0.0-16	408.4
		ADMM	97	1.00000000 0|1.9-11	5.8–7|1.3–8	653.7
FMMC-NotreDame_yeast	2203|2114	PPA	8 (8|3.4)	1.00000000 0|3.1-13	0.0–8|0.0-16	209.1
		ADMM	26	1.00000000 0|3.7–6	8.1–7|2.0–8	147.1
FDLA-G46	9990|1,000	PPA	11 (32|33.4)	4.17339208-1|7.8–6	1.9–7|1.8–7	111.6
		ADMM	2,000	4.17421122-1|3.4–4	4.5–6|1.8–5	1,307.0
FMMC-G46	9990|1,000	PPA	11 (24|26.8)	4.19936658-1|3.0–7	8.6–7|4.2–8	78.3
		ADMM	2,000	4.21142429-1|9.8–4	1.2–4|6.2–5	1,310.2
FDLA-G15	4661|800	PPA	13 (57|50.5)	7.31899971-1|7.9–6	4.5–7|1.9–7	138.9
		ADMM	1,122	7.31899758-1|4.1–4	3.7–7|9.9–7	426.4
FMMC-G15	4661|800	PPA	12 (57|72.7)	7.85243183-1|6.5–5	3.0–7|3.4–7	159.7
		ADMM	2,000	7.85529701-1|2.9–3	1.3–6|5.7–6	749.8
FDLA-G54	5916|1,000	PPA	15 (49|50.6)	7.32247725-1|2.7–4	6.2–7|5.3–7	191.8
		ADMM	2,000	7.33611791-1|2.0–4	7.8–5|2.9–5	1,424.3
FMMC-G54	5916|1,000	PPA	14 (73|91.4)	7.86519818-1|4.5–6	8.9–7|1.2–7	404.3
		ADMM	2,000	7.88923019-1|1.7–3	4.2–6|3.5–5	1,413.3
FDLA-G43	9990|1,000	PPA	12 (41|29.6)	4.21305462-1|7.9–6	5.3–7|8.8–8	133.5
		ADMM	2,000	4.21415022-1|5.7–4	2.3–6|2.3–5	1,333.3
FMMC-G43	9990|1,000	PPA	18 (48|48.7)	4.25983862-1|1.3–5	7.3–7|3.2–8	198.6
		ADMM	2,000	4.26209610-1|8.5–4	6.3–6|3.3–5	1,325.2

we see that the SNDPPA is slower for the former cases. This behavior is understandable because for FMMC/FDLA problems, the average PCG steps taken to compute the Newton directions and the total number of semismooth Newton-CG iterations are significantly larger. It is also not surprising that the ADMM fails to obtain solutions with the desired accuracy after 2,000 iterations for most of the instances. In fact, the

ADMM can only obtain an approximate solution with the primal-dual accuracy in the order 10^{-4}–10^{-5} for about 50 % of the instances and the objective gap more than 10^{-4} for about 80 % of the instances. The performance of the ADMM is especially poor for the problems `FDLA-Cage` and `FMMC-Cage`.

6 Conclusion

In this paper, we proposed a semismooth Newton-CG based dual proximal point algorithm (SNDPPA) to solve large scale matrix spectral norm approximation problems. In each iteration, the dual PPA solves the subproblems by a semismooth Newton-CG method and the Newton direction is computed inexactly by a PCG solver. Theoretical results to guarantee the global convergence and local superlinear convergence of the dual PPA are established based on the classical analysis of proximal point algorithms. Capitalizing on the recent advances on nonseparable spectral operators [8] and related perturbation analysis, we also characterize the nonsingularity of the semismooth Newton systems. The latter property is an important condition for the fast convergence of the semismooth Newton-CG method. Extensive numerical experiments on problems arising from different areas are conducted to evaluate the performance of the SNDPPA against the ADMM. The numerical results show that the SNDPPA, which is warm-started with an initial point obtained from the ADMM, is very efficient and robust, and it substantially outperforms the pure ADMM.

Appendix: Proof of Proposition 4.1

Let $\{Y^i\}_{i\geq 1}$ be a sequence converging to Y such that every element $Y^i \in \mathcal{D}_{\Pi_{\mathcal{B}}}$. This, by Proposition 2.3(i), implies that $\|Y^i\|_* \neq 1$ for each $i \geq 1$. Let the SVD of Y^i be $Y^i = U^i[\mathrm{Diag}(\sigma^i)\ 0](V^i)^T$. We consider the following three cases.

(i) $\|Y\|_* < 1$. In this case, $\Pi_{\mathcal{B}}(\cdot)$ is continuously differentiable at Y and its generalized Jacobian is a singleton consisting of the identity operator from $\Re^{m\times n}$ to itself.

(ii) $\|Y\|_* = 1$. Since Y can be approximated by a sequence in the interior of $\mathcal{B}$, it follows that the identity operator $\mathcal{I}$ is always an element of $\partial_B \Pi_{\mathcal{B}}(Y)$. To obtain the remaining elements, we consider the case in which $\{Y^i\}$ has an infinite subsequence outside $\mathcal{B}$. Without loss of generality, we assume that $\|Y^i\|_* > 1$ for all i. By passing to a subsequence if necessary, we know that there exists a positive integer $N \in [r,\ m]$ such that $N = k_1(\sigma^i)$ for each i. Therefore, one has

$$g_k^i := (\Pi_{\mathbb{B}}(\sigma^i))_k \ = \ \begin{cases} \sigma_k^i - \frac{1}{N}\Big(\sum_{j=1}^N \sigma_j^i - 1\Big), & 1 \leq k \leq N, \\ 0, & \text{otherwise} \end{cases}$$

and

$$\Pi'_{\mathbb{B}}(\sigma^i) = \begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix} - \frac{1}{N}\begin{bmatrix} \mathbf{1}_{N\times N} & 0 \\ 0 & 0 \end{bmatrix}.$$

For each i, it holds that

$$\big(\Omega(\sigma^i)\big)_{kj} = \begin{cases} 1, & \text{if } k, j \in \alpha_1 \cup \alpha_2, \\ \frac{g_k^i}{\sigma_k^i - \sigma_j^i}, & \text{if } k \in \alpha_1 \cup \alpha_2,\ j \in \alpha_3, \\ \frac{g_j^i}{\sigma_j^i - \sigma_k^i}, & \text{if } k \in \alpha_3,\ j \in \alpha_1 \cup \alpha_2, \\ 0, & \text{if } k,\ j \in \alpha_3, \end{cases}$$

$$\big(\Gamma(\sigma^i)\big)_{kj} = \begin{cases} \frac{g_k^i + g_k^j}{\sigma_k^i + \sigma_j^i}, & \text{if } k, j \in \alpha_1 \cup \alpha_2, \\ \frac{g_k^i}{\sigma_k^i + \sigma_j^i}, & \text{if } k \in \alpha_1 \cup \alpha_2,\ j \in \alpha_3, \\ \frac{g_j^i}{\sigma_k^i + \sigma_j^i}, & \text{if } k \in \alpha_3,\ j \in \alpha_1 \cup \alpha_2, \\ 0, & \text{if } k,\ j \in \alpha_3 \end{cases}$$

and

$$\big(\mathcal{F}(\sigma^i)\big)_{kj} = \begin{cases} -\frac{1}{N}, & \text{if } k, j \in \alpha_1 \cup \alpha_2, \\ 0, & \text{otherwise}, \end{cases} \quad \big(\Upsilon(\sigma^i)\big)_k = \begin{cases} \frac{g_k^i}{\sigma_k^i}, & \text{if } k \in \alpha_1 \cup \alpha_2, \\ 0, & \text{if } k \in \alpha_3. \end{cases}$$

Now from Proposition 2.3(i), we know that for any given $H \in \Re^{m\times n}$,

$$\Pi'_{\mathcal{B}}(Y^i)H = U^i \left[W^i - \frac{\mathrm{Tr}(\widetilde{H}_{11}^i)}{N} \begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix},\ \mathrm{Diag}\big(\Upsilon(\sigma^i)\big)\widetilde{H}_2^i \right] (V^i)^T, \tag{69}$$

where the matrix $W^i \in \Re^{m\times m}$ is defined by

$$W^i = \Omega(\sigma^i) \circ S(\widetilde{H}_1^i) + \Gamma(\sigma^i) \circ T(\widetilde{H}_1^i)$$

with $\widetilde{H}_1^i \in \Re^{m\times m}$, $\widetilde{H}_2^i \in \Re^{m\times(n-m)}$, $[\widetilde{H}_1^i\ \ \widetilde{H}_2^i] = (U^i)^T H V^i$ and $\widetilde{H}_{11}^i$ being the matrix extracted from the first N columns and rows of $\widetilde{H}_1^i$. By simple algebraic computations, we have

$$\begin{aligned} &\lim_{i\to\infty} \big(\Omega(\sigma^i)\big)_{\alpha_1\alpha_3} = \lim_{i\to\infty} \big(\Omega(\sigma^i)\big)^T_{\alpha_3\alpha_1} = \mathbf{1}_{r\times(m-N)}, \\ &\lim_{i\to\infty} \big(\Gamma(\sigma^i)\big)_{\alpha_1(\alpha_1\cup\alpha_2\cup\alpha_3)} = \big(\Gamma(\sigma^i)\big)^T_{(\alpha_1\cup\alpha_2\cup\alpha_3)\alpha_1} = \mathbf{1}_{r\times m}, \\ &\lim_{i\to\infty} \big(\Upsilon(\sigma^i)\big)_{\alpha_1} = \mathbf{1}_r. \end{aligned}$$

Note that $\left\{\left(\big(\Omega(\sigma^i)\big)_{\alpha_2\alpha_3}, \big(\Gamma(\sigma^i)\big)_{\alpha_2\alpha_2}, \big(\Gamma(\sigma^i)\big)_{\alpha_2\alpha_3}, \big(\Upsilon(\sigma^i)\big)_{\alpha_2}\right)\right\}_{i\ge 1}$ is bounded and $\mathcal{S}_N$ is the set of cluster points associated with the sequence. By taking limits on both sides of (69), we are able to establish the conclusion that for any $\mathcal{V}(\neq \mathcal{I}) \in \partial_B \Pi_{\mathcal{B}}(Y)$, there exist an integer $N \in [r, m]$,

$(\Omega^{\infty}_{\alpha_2\alpha_3}, \Gamma^{\infty}_{\alpha_2\alpha_2}, \Gamma^{\infty}_{\alpha_2\alpha_3}, \Upsilon^{\infty}_{\alpha_2}) \in \mathcal{S}_N$ and singular vector matrices U^{∞}, V^{∞} of Y such that for any $H \in \Re^{m\times n}$, (33) is valid.

(iii) $\|Y\|_* > 1$. Taking a subsequence if necessary, we know that there exists a positive integer $N \in [k_1(\sigma), k_2(\sigma)]$ such that $N = k_1(\sigma^i)$ for each i. Therefore,

$$g^i_k := (\Pi_{\mathbb{B}}(\sigma^i)_k = \begin{cases} \sigma^i_k - \frac{1}{N}\Big(\sum_{j=1}^N \sigma^i_j - 1\Big), & 1 \le k \le N, \\ 0, & \text{otherwise} \end{cases}$$

and

$$\Pi'_{\mathbb{B}}(\sigma^i) = \begin{bmatrix} I_N & 0 \\ 0 & 0 \end{bmatrix} - \frac{1}{N}\begin{bmatrix} \mathbf{1}_{N\times N} & 0 \\ 0 & 0 \end{bmatrix}.$$

For each i, it holds that

$$\big(\Omega(\sigma^i)\big)_{kj} = \begin{cases} 1, & \text{if } k, j \in \gamma_1, \\ \frac{g^i_k}{\sigma^i_k - \sigma^i_j}, & \text{if } k \in \gamma_1,\ j \in \gamma_2, \\ \frac{g^i_j}{\sigma^i_j - \sigma^i_k}, & \text{if } k \in \gamma_2,\ j \in \gamma_1, \\ 0 & \text{if } k,\ j \in \gamma_2, \end{cases}$$

$$\big(\Gamma(\sigma^i)\big)_{kj} = \begin{cases} \frac{g^i_k + g^i_j}{\sigma^i_k + \sigma^i_j}, & \text{if } k, j \in \gamma_1, \\ \frac{g^i_k}{\sigma^i_k + \sigma^i_j}, & \text{if } k \in \gamma_1,\ j \in \gamma_2, \\ \frac{g^i_j}{\sigma^i_k + \sigma^i_j}, & \text{if } k \in \gamma_2,\ j \in \gamma_1, \\ 0, & \text{if } k,\ j \in \gamma_2 \end{cases}$$

and

$$\big(\mathcal{F}(\sigma^i)\big)_{kj} = \begin{cases} -\frac{1}{N}, & \text{if } k, j \in \gamma_1, \\ 0, & \text{otherwise}, \end{cases} \quad \big(\Upsilon(\sigma^i)\big)_k = \begin{cases} \frac{g^i_k}{\sigma^i_k}, & \text{if } k \in \gamma_1, \\ 0, & \text{otherwise}. \end{cases}$$

Then the equality (69) is also valid. Simple calculations show that

$$\begin{aligned} &\lim_{i\to\infty} \big(\Omega(\sigma^i)\big)_{\beta_1\gamma_2} = \lim_{i\to\infty} \big(\Omega(\sigma^i)\big)^T_{\gamma_2\beta_1} = \Omega_{\beta_1\gamma_2}, \\ &\lim_{i\to\infty} \big(\Omega(\sigma^i)\big)_{\beta_2\beta_4} = \lim_{i\to\infty} \big(\Omega(\sigma^i)\big)^T_{\beta_4\beta_2} = \Omega_{\beta_2\beta_4}, \\ &\lim_{i\to\infty} \big(\Gamma(\sigma^i)\big)_{\gamma_1(\gamma_1\cup\gamma_2)} = \lim_{i\to\infty} \big(\Gamma(\sigma^i)\big)^T_{(\gamma_1\cup\gamma_2)\gamma_1} = \Gamma_{\gamma_1(\gamma_1\cup\gamma_2)}, \\ &\lim_{i\to\infty} \big(\Upsilon(\sigma^i)\big)_{\gamma_1} = \Upsilon_{\gamma_1}. \end{aligned}$$

Note that $\Big\{\big(\Omega(\sigma^i)\big)_{\beta_2\beta_3}\Big\}$ is bounded and $\mathcal{S}_N$ is the set of cluster points associated with the sequence. By taking limits on both sides of (69), we have the conclusion

that for any $\mathcal{V} \in \partial_B \Pi_{\mathcal{B}}(Y)$, there exist an integer $N \in [k_1(\sigma),\ k_2(\sigma)]$, $\Omega^{\infty}_{\beta_2\beta_3} \in \mathcal{S}_N$ and singular vector matrices U^{∞}, V^{∞} of Y such that for any $H \in \Re^{m\times n}$, (34) holds. This completes the proof.

References

1. Bonnans, J.F., Shapiro, A.: Perturbation Analysis of Optimization Problems. Springer, New York (2000)
2. Boyd, S., Diaconis, P., Sun, J., Xiao, L.: Fastest mixing Markov chain on a path. Am. Math. Mon. **113**, 70–74 (2006)
3. Boyd, S., Diaconis, P., Xiao, L.: Fastest mixing Markov chain on a graph. SIAM Rev. **46**, 667–689 (2004)
4. Chen, C.H., He, B.S., Yuan, X.M.: Matrix completion via alternating direction methods. IMA J. Numer. Anal. **32**, 227–245 (2012)
5. Chen, X., Qi, H.-D., Qi, L., Teo, K.-L.: Smooth convex approximation to the maximum eigenvalue function. J. Glob. Optim. **30**, 253–270 (2004)
6. Clarke, F.H.: Optimization and Nonsmooth Analysis, 2nd edn. SIAM, Philadelphia (1990)
7. Davis, T.A., Hu, Y.: University of Florida sparse matrix collection (2009)
8. Ding, C.: An introduction to a class of matrix optimization problems. Ph.D. thesis, National University of Singapore, Singapore (2012)
9. Ding, C., Sun, D.F., Toh, K.-C.: An introduction to a class of matrix cone programming. Math. Program. Ser. A **144**, 141–179 (2014)
10. Eckstein, J., Bertsekas, D.P.: On the Douglas–Rachford splitting method and the proximal point algorithm for maximal monotone operators. Math. Program. Ser. A **55**, 293–318 (1992)
11. Facchinei, F., Pang, J.-S.: Finite-Dimensional Variational Inequalities and Complementarity Problems, vol. II. Springer, New York (2003)
12. Gabay, D., Mercier, B.: A dual algorithm for the solution of nonlinear variational problems via finite element approximations. Comput. Math. Appl. **2**, 17–40 (1976)
13. Glowinski, R., Tallec, P.L.: Augmented Lagrangian and Operator Splitting Methods in Nonlinear Mechanics. SIAM, Philadelphia (1989)
14. Glowinski, R., Marrocco, A.: Sur l'approximation, par éléments finis d'ordre un, et la résolution, par pénalisation-dualité, d'une classe de problèmes de Dirichlet non-linéaires. Rev. Française Autom. Inf. Recherche Opérationnelle Anal. Numér. **9**, 41–76 (1975)
15. Greenbaum, A., Trefethen, L.N.: GMRES/CR and Arnoldi/Lanczos as matrix approximation problems. SIAM J. Sci. Comput. **15**, 359–368 (1994)
16. Ha, C.D.: A generalization of the proximal point algorithm. SIAM J. Control Optim. **28**, 503–512 (1990)
17. He, B.S., Liao, L.-Z., Han, D.R., Yang, H.: A new inexact alternating directions method for monotone variational inequalities. Math. Program. Ser. A **92**, 103–118 (2002)
18. Held, M., Wolfe, P., Crowder, H.P.: Validation of subgradient optimization. Math. Program. **6**, 62–88 (1974)
19. Hiriart-Urruty, J.-B., Lemaréchal, C.: Convex Analysis and Minimization Algorithms, vols. I and II. Springer, Berlin (1993)
20. Hiriart-Urruty, J.-B., Strodiot, J.-J., Nguyen, V.H.: Generalized Hessian matrix and second-order optimality conditions for problems with $C^{1,1}$ data. Appl. Math. Optim. **11**, 43–56 (1984)
21. Jiang, K.F.: Algorithms for large scale nuclear norm minimization and convex quadractic semidefinite programming problems. Ph.D. thesis. National University of Singapore, Singapore (2012)
22. Jiang, K.F., Sun, D.F., Toh, K.C.: A partial proximal point algorithm for nuclear norm regularized matrix least squares problems. Math. Program. Comput. **6**, 281–325 (2014)
23. Jiang, K.F., Sun, D.F., Toh, K.C.: Solving nuclear norm regularized and semidefinite matrix least squares problems with linear equality constraints. In: Bezdek, K., Ye, Y., Deza, A. (eds.) Fields Institute Communications Series on Discrete Geometry and Optimization. Springer, Switzerland (2013)
24. Lemaréchal, C., Sagastizábal, C.: Practical aspects of the Moreau–Yosida regularization: theoretical preliminaries. SIAM J. Optim. **7**, 367–385 (1997)

25. Lewis, A.S., Sendov, H.S.: Nonsmooth analysis of singular values. II: Applications. Set-Valued Anal. **13**, 243–264 (2005)
26. Liese, J., Tichý, P.: On best approximations of polynomials in matrices in the matrix 2-norm. SIAM J. Matrix Anal. Appl. **31**, 853–863 (2009)
27. Faber, V., Liesen, J., Tichý, P.: On Chebyshev polynomials of matrices. SIAM J. Matrix Anal. Appl. **31**, 2205–2221 (2010)
28. Liu, Y.-J., Sun, D.F., Toh, K.-C.: An implementable proximal point algorithmic framework for nuclear norm minimization. Math. Program. Ser. A **133**, 399–436 (2012)
29. Meng, F.W., Sun, D.F., Zhao, G.Y.: Semismoothness of solutions to generalized equations and the Moreau–Yosida regularization. Math. Program. Ser. B **104**, 561–581 (2005)
30. Moreau, J.J.: Proximite et dualite dans un espace hilbertien. Bull. Soc. Math. France **93**, 273–299 (1965)
31. Qi, L.Q., Sun, J.: A nonsmooth version of Newton's method. Math. Program. Ser. A **58**, 353–367 (1993)
32. Robinson, S.M.: Local structure of feasible sets in nonlinear programming. II: Nondegeneracy. Math. Program. Stud. **22**, 217–230 (1984)
33. Rockafellar, R.T.: Augmented Lagrangians and applications of the proximal point algorithm in convex programming. Math. Oper. Res. **1**, 97–116 (1976)
34. Rockafellar, R.T.: Convex Analysis. Princeton University Press, Princeton (1970)
35. Rockafellar, R.T.: Monotone operators and the proximal point algorithm. SIAM J. Control Optim. **14**, 877–898 (1976)
36. Singer, I.: Best Approximation in Normed Linear Spaces by Elements of Linear Subspaces. Springer, Berlin (1970)
37. Sturm, J.F.: Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. Optim. Methods Softw. **11–12**, 625–653 (1999)
38. Toh, K.-C.: Solving large scale semidefinite programs via an iterative solver on the augmented systems. SIAM J. Optim. **14**, 670–698 (2003)
39. Toh, K.-C., Todd, M.J., Tütüncü, R.H.: SDPT3—a MATLAB software package for semidefinite programming, version 1.3. Optim. Methods Softw. **11–12**, 545–581 (1999)
40. Toh, K.-C., Trefethen, L.N.: The Chebyshev polynomials of a matrix. SIAM J. Matrix Anal. Appl. **20**, 400–419 (1999)
41. von Neumann, J.: Some matrix inequalities and metrization of metric spaces. Tomsk. Univ. Rev. **1**, 286–300 (1937)
42. Wang, C.J., Sun, D.F., Toh, K.-C.: Solving log-determinant optimization problems by a Newton-CG primal proximal point algorithm. SIAM J. Optim. **20**, 2994–3013 (2010)
43. Watson, G.A.: Characterization of the subdifferential of some matrix norms. Linear Algebra Appl. **170**, 33–45 (1992)
44. Xiao, L., Boyd, S.: Fast linear iterations for distributed averaging. Syst. Control Lett. **53**, 65–78 (2004)
45. Yang, J.F., Zhang, Y.: Alternating direction algorithms for ℓ_1-problems in compressive sensing. SIAM J. Sci. Comput. **33**, 250–278 (2011)
46. Yosida, K.: Functional Analysis. Springer, Berlin (1964)
47. Zhao, X.-Y., Sun, D.F., Toh, K.-C.: A Newton-CG augmented Lagrangian method for semidefinite programming. SIAM J. Optim. **20**, 1737–1765 (2010)
48. Zietak, K.: Properties of linear approximations of matrices in the spectral norm. Linear Algebra Appl. **183**, 41–60 (1993)