A PRIMAL-DUAL ALGORITHM FOR UNFOLDING NEUTRON ENERGY SPECTRUM FROM MULTIPLE ACTIVATION FOILS

YU-HONG DAI
LESC, ICMSEC, Academy of Mathematics and Systems Science
Chinese Academy of Sciences, Beijing 100190, China
School of Mathematical Sciences, University of Chinese Academy of Sciences
Beijing 100049, China

ZHOUHONG WANG
School of Science, Beijing Jiaotong University
Beijing, 100044, China

FENGMIN XU
School of Economics and Finance, Xian Jiaotong University
Xi’an, 710061, China

(Communicated by Tamás Terlaky)

Abstract. In this paper we propose a robust and efficient primal-dual interior-point method for a nonlinear ill-conditioned problem with associated errors which are arising in the unfolding procedure for neutron energy spectrum from multiple activation foils. Based on the maximum entropy principle and Boltzmann’s entropy formula, the discrete form of the unfolding problem is equivalent to computing the analytic center of the polyhedral set \( P = \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \} \), where the matrix \( A \in \mathbb{R}^{m \times n} \) is ill-conditioned, and both \( A \) and \( b \) are inaccurate. By some derivations, we find a new regularization method to reformulate the problem into a well-conditioned problem which can also reduce the impact of errors in \( A \) and \( b \). Then based on the primal-dual interior-point methods for linear programming, we propose a hybrid algorithm for this ill-conditioned problem with errors. Numerical results on a set of ill-conditioned problems for academic purposes and two practical data sets for unfolding the neutron energy spectrum are presented to demonstrate the effectiveness and robustness of the proposed method.

1. Introduction. The knowledge on neutron spectrometry is imperative to various applications and research activities related to the neutron irradiation in nuclear physics. A number of neutron detection systems and spectrometers such as proton recoil scintillators, Bonner sphere systems, foil activation detectors, etc., have been employed in general to detect neutrons and measure their energy distributions.

2010 Mathematics Subject Classification. Primary: 65K05, 90C51; Secondary: 65F22.
Key words and phrases. neutron spectrum unfolding, Boltzmann’s entropy, ill-condition, numerical stability, interior point algorithm, analytic center.

This work was partly supported by the Chinese NSF grants (nos. 11631013, 11991021, 11971372 and 11991020) and partly supported by the CSC scholarship and Beijing Academy of Artificial Intelligence (BAAI). The authors are grateful to the editor and the referees for their valuable comments and suggestions.

* Corresponding author: Zhouhong Wang.
These methods usually lead to an ill-posed or ill-conditioned inverse problem. The interest in neutron spectrometry has stimulated the development of several deconvolution procedures like SAND-II, SPECTRA, BUNKI, FRUIT, MAXED, GRAVEL, HPRO, BONDI-97, ANDI-03, etc. (see, e.g., [18, 23, 28] and references therein). These procedures use least square methods, Monte Carlo methods, artificial neural network algorithms and genetic algorithms to unfold the neutron energy spectrum. Since the unfolding problem is usually ill-posed or ill-conditioned, we require some kind of regularization method to make the solution unique and less sensitive to data errors. Various regularization methods have been proposed for these problems, such as Tikhonov regularization methods [2, 8], maximum entropy methods [32, 14, 24], Fisher regularization methods [9], trust region regularization methods [31, 30], $l_1$ regularization (see, e.g., [6], which is usually used in signal processing) and exact regularization methods (see, e.g., [10, 7]), etc. However, since the resulted unfolding problem is usually ill-conditioned with associated errors, the numerical results we get may have extremely large variances. Due to these uncertainties, quantifying the confidence of the unfolding methodology still remains a complex task, as pointed out in [18, 27].

The multiple foil activation technique made itself widely used in the measurement of neutron spectrum due to its low interference, measurability in pulse condition, insensitivity to other radiation components, and the relative easiness in constructing the response matrix for the unfolding purposes among many other advantages. For the fewer number of activation foils available and the more number of energy group required, it will lead to an underdetermined system of equations with associated errors, which usually has many different solutions, or has no solution at all due to the errors. In order to overcome these difficulties, the maximum entropy regularization methods are favored in [32, 14, 24] (and references therein), where the Poisson statistics of neutron detection and Shannon’s information entropy are used. The advantages of the method is that it could be used without any a priori information about neutrons, and be used to derive a priori distribution for other unfolding procedures. This method is both consistent and unbiased in inference, and can guarantee the positiveness of the solution which is a conditio sine qua non for all particle energies.

In the maximum entropy regularization methods, another possible way is to use the Boltzmann’s entropy formula instead of Shannon’s information entropy formula, which has been noticed in the dissertation [41]. By using Boltzmann’s entropy formula, the discrete form of the maximum entropy regularization method for unfolding the neutron energy spectrum can be described as

$$\min - \sum_{j=1}^{n} \ln x_j$$
$$s.t. \quad Ax = b,$$
$$x_j > 0, \quad j = 1, \ldots, n,$$

(1)

where $\sum_{j=1}^{n} \ln x_j$ is proportional to the discrete entropy function for the neutron energy spectrum according to the Boltzmann’s entropy formula in statistical mechanics, i.e., “$S = k \ln W$”, where $k$ is the Boltzmann constant and $W$ is the number of real microstates corresponding to the macrostate of the gas. The vector $x \in \mathbb{R}^n$ is a column vector, whose components are variables $x_j (j = 1, \ldots, n)$, which is the number of neutrons in the $j$-th energy group. The $i$-th element $b_i$ of vector $b \in \mathbb{R}^m$
is the observed mononuclear saturation activity of the \(i\)-th detector \((i = 1, \ldots, m)\). The element \(a_{ij}\) in the matrix \(A = (a_{ij})_{m \times n}\) is proportional to the nuclear reaction microscopic cross section of the \(j\)-th energy group in the \(i\)-th detector. Due to the cost and technology, the number \(m\) cannot be too big, which means that \(m \ll n\). In addition, the inherent errors of the instrument and observation in the process of experiment will result in inaccuracy of vector \(b\). According to our knowledge, model (1) has never been studied before for unfolding the neutron spectrum in nuclear science. Our numerical results in Section 4 show that the new method for unfolding neutron spectrum is indeed promising. Comparing with the mathematical model obtained by Shannon’s information entropy (where the objective would be \(-\sum_{j=1}^{n} x_j \ln x_j\) as studied in \([32, 14, 24]\)), model (1) has certain geometrical meaning.

The unique optimal solution \(x^*\) of (1) is just the analytic center of the polytope \(\{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}\) if it exists, which is independent of affine changes of coordinates and is invariant under positive scalings of the constraints. Another advantage of model (1) is that many efficient primal-dual methods for computing the analytic center of a polyhedral set have been studied in the literature of interior point methods for linear programming (LP) and convex optimization (see, e.g., \([1, 26, 37, 5]\) and references therein).

However we met some numerical difficulties when we use model (1) to unfold the neutron energy spectrum for practical problems. Firstly, in order to get one accurate vector \(b\), the instrument called multi-foil activation spectrometer is required to be highly precise and highly sensitive. Because of the inherent error of the instrument itself and inappropriate operations, there is always some errors in vector \(b\). The errors in vector \(b\), which are usually much bigger than we desire, may lead to an infeasible problem. Secondly the matrix \(A\) is usually ill-conditioned by the discretization methods, which may cause serious numerical difficulties when we use existing primal-dual interior point method to compute the corresponding analytic center directly, because a linear system \(AD^2A^T y = h\) or some equivalent systems will be formed with ill-conditioned matrices \(A\) and \(D\) in the process of computation. Although M. H. Wright \([33]\) and S. J. Wright \([34]\) have shown that the ill-conditioning of the matrix \(D\) does not noticeably impair the accuracy of the computed primal-dual steps in certain cases, the ill-conditioning of the matrix \(A\) really causes serious numerical difficulties in practical computations. Despite a solution can be gotten by some numerical techniques for an ill-conditioned linear system (see, e.g., \([40]\)), the solution obtained is usually not the one we need and may damage the convergence properties of the algorithm, let alone the errors in \(A\) and \(b\). The computational difficulties caused by the ill-conditioning of the matrix \(A\) will be shown by some numerical experiments for academic purposes in Section 4. In fact, we have tried the techniques proposed in \([13]\), and use some software such as LIPSOL \([40]\) and IPOPT \([29]\) to solve two practical problems of the form (1) presented in Section 4, but their numerical performances are unsatisfactory due to the ill-conditioning of \(A\) and the errors in \(A\) and \(b\). Hence a more robust and efficient method for the computation of problem (1) is needed.

For this purpose, we propose a numerically stable primal-dual interior point method for solving the nonlinear problem (1) in this paper, where the matrix \(A\) is ill-conditioned, and both \(A\) and \(b\) are inaccurate. We firstly derive some regularization results for the ill-conditioned problem (1) and transform it into a problem which is
equivalent to finding the analytic center of the optimal solution set of a usually well-conditioned linear programming problem. The reformulation can not only reduce the impact of errors in $A$ and $b$ but can also improve the condition number. Then we propose a hybrid primal-dual interior point algorithm for the reformulated problem. Finally, some numerical experiments are presented to demonstrate the effectiveness and robustness of the new algorithm.

2. Regularization and transformation. Here we firstly give some conventions and assumptions, which will be used throughout this paper. For any system of (linear) equalities and (linear) inequalities, if a feasible point $x$ strictly satisfies all inequality constraints in the system, we will call the point $x$ as an interior point of the system. For a vector $x \in \mathbb{R}^n$, we will always denote its $j$-th component by $x_j$, and $x \geq (> \geq 0$ means $x_j \geq (> 0$ for every $j = 1, \ldots, n$. The superscript $^T$ means the transpose of a vector or matrix. The symbol $e$ denotes the vector of all ones, whose dimension can be determined by the context. Given a vector $x$, the corresponding upper case symbol $X$ denotes the diagonal matrix defined by the vector. We shall denote component-wise operations on vectors by the usual notations for real numbers. This means, given two vectors $u, v$ of the same dimension, $uv, u/v$, etc., are the vectors with components $u_i v_i, u_i/v_i$, etc.

According to the practical background of problem (1), we make the following assumption throughout this paper.

**Assumption 2.1.** In problem (1) with $A = (a_{ij})_{m \times n}, b = (b_1, \ldots, b_m)^T$, we have $m \ll n$, and $a_{ij} \geq 0$ for $i = 1, \ldots, m, j = 1, \ldots, n$, $b_i > 0$ for $i = 1, \ldots, m$, and $A^T e > 0$ where $e$ is the vector of ones in $\mathbb{R}^m$.

Denote the feasible solution set of problem (1) as:

$$P_1 = \{x \in \mathbb{R}^n \mid Ax = b, x \geq 0\}.$$

As pointed above, the optimal solution $x^*$ of (1) if exists is just the analytic center of the polyhedral set $P_1$. For completeness, we present the definition of analytic center here, which can be found in [26].

**Definition 2.1 (Analytic center).** Let the nonempty and bounded set $T$ be the intersection of an affine space in $\mathbb{R}^n$ with the nonnegative orthant of $\mathbb{R}^n$. Define the support $\sigma(T)$ of $T$ as the subset of the full index set $\{1, 2, \ldots, n\}$ given by $\sigma(T) = \{i : \exists x \in T$ such that $x_i > 0\}$. The analytic center of $T$ is defined as the zero vector if $\sigma(T)$ is empty; otherwise it is the vector in $T$ that maximizes the product

$$\prod_{i \in \sigma(T)} x_i, x \in T.$$

By the definition of the analytic center, we have the following result.

**Lemma 2.2.** If $P_1$ has a nonempty interior, problem (1) has a unique optimal solution which is just the analytic center of the polyhedral set $P_1$.

Since the matrix $A$ is ill-conditioned, which means that the matrix $AA^T$ is nearly singular, we will usually meet some numerical difficulties if we solve problem (1) directly. Furthermore, the inaccuracy of $b$ usually lead problem (1) to be infeasible.

---

1The definition of analytic center for the set $S = \{x \in \mathbb{R}^n \mid Ax \leq b\}$ is a little different, which maximize $\prod_{i=1}^{m} (b_i - a_i^T x)$ where $[a_i, b_i]$ is the $i$-th row of $[A, b]$ if the interior of $S$ is not empty, see [37, 5].
Hence some reformulations are needed. Based on the fact that the right hand side \( b \) is always bigger than desired in practical computations, we propose the following penalty model for the computation of problem (1).

\[
\begin{align*}
\min & \quad - \sum_{j=1}^{n} \ln x_j + \rho \sum_{i=1}^{m} y_i \\
\text{s.t.} & \quad Ax + y = b, \\
& \quad x_j \geq 0, \quad j = 1, \ldots, n, \\
& \quad y_i \geq 0, \quad i = 1, \ldots, m
\end{align*}
\]

(2)

where \( \rho > 0 \) is the penalty parameter, the variable \( y \) can be seen as a correction of the errors in vector \( b \). With Assumption 2.1, problem (2) is always well defined even when there are big errors in vector \( b \) which may result in an infeasible linear system \( "Ax = b, x \geq 0" \). Furthermore, the matrix \([A, I]\) is well-conditioned if \( m \ll n \) and we scale \( A \) in advance, where \( I \in \mathbb{R}^{m \times m} \) is the identity matrix. In fact, we have the following results.

**Lemma 2.3.** Suppose that Assumption 2.1 is satisfied and \( \sum_{j=1}^{n} a_{ij} \leq 1 \) for every \( i = 1, \ldots, m \). Denote \( B = [A, I] \). Then the condition number \( \kappa_2(B) \) of the matrix \( B \) satisfies

\[
\kappa_2(B) \leq \sqrt{m + 1}.
\]

**Proof.** Since the matrix \( B \) has full row rank, by the definition of the condition \( \kappa_2(B) \) (see, e.g., Chapter 5 in [11]) we have

\[
\kappa_2(B) = \frac{\sigma_{\max}(B)}{\sigma_{\min}(B)} = \frac{\lambda_{\max}(BB^T)}{\lambda_{\min}(BB^T)},
\]

where \( \sigma_{\max}(B) \) and \( \sigma_{\min}(B) \) are the largest and smallest singular values of the matrix \( B \) respectively, and \( \lambda_{\max}(BB^T) \) and \( \lambda_{\min}(BB^T) \) are the largest and smallest eigenvalues of the matrix \( BB^T \) respectively. Since \( BB^T = I + AA^T \) and \( AA^T \) is positive semidefinite, we have

\[
\lambda_{\min}(BB^T) \geq 1.
\]

By Assumption 2.1 and \( \sum_{j=1}^{n} a_{ij} \leq 1 \) for \( i = 1, \ldots, m \), we have \( \sum_{k=1}^{n} a_{ik}a_{jk} \leq (\sum_{k=1}^{n} a_{ik})(\sum_{k=1}^{n} a_{jk}) \leq 1 \). Then by the Gerschgorin circles theorem we get

\[
\lambda_{\max}(BB^T) \leq \max_{1 \leq i \leq m} \left\{ 1 + \sum_{j=1}^{m} \sum_{k=1}^{n} a_{ik}a_{jk} \right\} \leq 1 + m.
\]

Thus we obtain \( \kappa_2(B) = \frac{\lambda_{\max}(BB^T)}{\lambda_{\min}(BB^T)} \leq \sqrt{m + 1} \). \( \square \)

**Remark 2.1.** Since in practical computations we have \( m \ll n \) and \( m \) is usually in the interval \([20, 30]\), by Lemma 2.3 we know that the matrix \([A, I]\) is usually well-conditioned if we scale \( A \) in advance such that \( \sum_{j=1}^{n} a_{ij} \leq 1 \) for every \( i = 1, \ldots, m \).

Denote the feasible solution set of problem (2) as:

\[
P_2 = \{(x, y) \in \mathbb{R}^n \times \mathbb{R}^m \mid Ax + y = b, x \geq 0, y \geq 0\}.
\]

**Lemma 2.4.** The set \( P_2 \) is a compact set with a nonempty interior, and the optimal solution set of problem (2) is nonempty and compact.
Proof. By Assumption 2.1, we can find a positive integer \( p \) large enough such that 
\[ A \left( \frac{e}{y} \right) < b. \]
Let \( x_0 = \frac{e}{y}, y_0 = b - Ax_0 > 0 \), we know that \((x_0, y_0)\) is an interior point of the set \( P_2 \). For all \((x, y) \in P_2\), we have \( 0 \leq y \leq b, Ax \leq b, x \geq 0 \). By Assumption 2.1 we have \( a_{ij} \geq 0 \) and there exists at least one \( i_j \in \{1, \ldots, m\} \) such that \( a_{i_jj} > 0 \) for every \( j = 1, \ldots, n \). Therefore we have
\[
0 \leq a_{i_jj}x_j \leq \sum_{k=1}^n a_{i_jk}x_k \leq b_{i_j},
\]
which means \( 0 \leq x_j \leq \frac{b_{i_j}}{a_{i_jj}} \). Hence the set \( P_2 \) is bounded. Since \( P_2 \) is also closed, we know the set \( P_2 \) is compact.

Since the set \( P_2 \) has a nonempty interior, the optimal objective value of problem (2) is bounded above. Furthermore, by the fact that \( P_2 \) is compact, the objective values of problem (2) on \( P_1 \) is bounded below. Hence problem (2) has a finite optimal objective value and the corresponding optimal solution set is nonempty and compact.

Based on the theory for the exact penalization of convex programs (see, e.g., [3, 10, 7]), we have the following result.

**Theorem 2.5.** Suppose that the set \( P_1 \) has a nonempty interior. Then problem (1) has a unique optimal solution \( x > 0 \). Moreover, there exists a constant \( c > 0 \) such that when \( \rho > c \), the solution \( x = \bar{x}, y = 0 \) is the unique optimal solution of problem (2).

**Proof.** The first part of the proposition is implied by Lemma 2.2. We only need to prove the second part. Assume that the unique solution of problem (1) is \( \bar{x} \) with \( \bar{x}_j > 0, j = 1, \ldots, n \). Denote \( f(x) = -\sum_{j=1}^n \ln x_j \). By Lemma 2.2, the fact that the set \( P_1 \) has a nonempty interior and Theorem 28.2 in [25], there exist some Lagrange multipliers \( \bar{w} = (\bar{w}_1, \ldots, \bar{w}_m)^T \in \mathbb{R}^m \) such that
\[
f(\bar{x}) = \inf_{x > 0} \{ f(x) - \bar{w}^T (Ax - b) \}.
\]
(3)

Let \( c = \max\{|w_i| : i = 1, \ldots, m\} \) and \( \rho > c \). For any feasible point \((x, y)\) of problem (2), by (3) we have
\[
f(\bar{x}) \leq f(x) - \bar{w}^T (Ax - b) \leq f(x) + \rho \sum_{i=1}^m y_i,
\]
since \( y = b - Ax \geq 0 \). Hence \( x = \bar{x}, y = 0 \) is an optimal solution of problem (2). On the contrary, suppose \((\bar{x}, \bar{y})\) is an optimal solution of problem (2). Then \( \bar{y} \geq 0 \). If \( e^T \bar{y} > 0 \), denoting \( \bar{y} = 0 \) and by (3) we get
\[
f(\bar{x}) + \rho \sum_{i=1}^m \bar{y}_i > f(\bar{x}) - \bar{w}^T \bar{y} \geq f(\bar{x}) - \bar{w}^T \bar{y} = f(\bar{x}) + \rho \sum_{i=1}^m \bar{y}_i.
\]
This contradicts the optimality of the point \((\bar{x}, \bar{y})\). Hence we must have \( e^T \bar{y} = 0 \), which means \( \bar{y} = 0 \). Then \( \bar{x} \) is an optimal solution of problem (1), and by the uniqueness of the optimal solution of problem (1) we get \( \bar{x} = \bar{x} \).

**Remark 2.2.** According to Theorem 2.5, problem (2) has a unique solution when the penalty parameter \( \rho \) is big enough and problem (1) has a nonempty interior,
even though the objective function of problem (2) is not strictly convex. This is somewhat surprising.

It is obvious that problem (2) is equivalent to the following problem

$$\begin{align*}
\min & -\mu \sum_{j=1}^{n} \ln x_j + \sum_{i=1}^{m} y_i \\
\text{s.t.} & \quad Ax + y = b, \\
& \quad x_j \geq 0, \quad j = 1, \ldots, n, \\
& \quad y_i \geq 0, \quad i = 1, \ldots, m,
\end{align*}$$

where $\mu = \frac{1}{\rho} > 0$, and $\mu \downarrow 0$ as $\rho \uparrow +\infty$. The problem (4) is in fact some kind of logarithmic barrier penalty model (without barricading the variable $y$) for the following linear programming problem

$$\begin{align*}
\min & \quad \sum_{i=1}^{m} y_i \\
\text{s.t.} & \quad Ax + y = b, \\
& \quad x_j \geq 0, \quad j = 1, \ldots, n, \\
& \quad y_i \geq 0, \quad i = 1, \ldots, m,
\end{align*}$$

where $x = (x_1, \ldots, x_n)^T \in \mathbb{R}^n$, $y = (y_1, \ldots, y_m)^T \in \mathbb{R}^m$.

It is surprising to find that problem (5) is the same as the first-phase problem in the two-phase simplex methods for linear programming. In fact, when the polytope $P_1$ is nonempty, the analytic center of the optimal solution set of the problem (5) is the same as that of the polytope $P_1$ by Definition 2.1.

**Theorem 2.6.** The optimal solution set $S$ of problem (5) is nonempty and compact. Moreover, if the polytope $P_1$ is nonempty, the analytic center of the optimal solution set $S$ is the same as that of the polytope $P_1$ in $x$-part, and its $y$-part is always zero.

**Proof.** By Lemma 2.4 we know that problem (5) has a compact feasible set with a nonempty interior and its objective function is continuous on the compact feasible set. Hence the optimal solution set $S$ of problem (5) is nonempty and compact. When $P_1 \neq \emptyset$, for every $(\bar{x}, \bar{y}) \in S$ we must have that $\bar{y} = 0$ and $\bar{x}$ is just a feasible solution of in $P_1$ by the definition of problem (5). Hence $\sigma(P_1) = \sigma(S)$ by Definition 2.1, and the analytic center of $S$ is the same as that of the polytope $P_1$ in $x$-part by definition, with its $y$-part being zero. \qed

**Remark 2.3.** When the polyhedral set $P_1 = \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \}$ has a nonempty interior, we have $\sigma(P_1) = \{ 1, 2, \ldots, n \}$ by Definition 2.1. Hence problem (1) is equivalent to finding the analytic center of the optimal solution set of problem (5) in this case. When the polyhedral set $P_1$ is nonempty but has no interior point, or $P_1 = \emptyset$, problem (1) will have no solution. However, by Lemma 2.4 problem (5) always has a nonempty interior and its optimal solution set is nonempty and compact by Theorem 2.6. Then by Corollary II.11 in [26], we know that the dual problem of problem (5) has a nonempty interior. Hence we can always solve problem (5) by some primal-dual interior point method even when the set $P_1$ has no feasible interior point, or $P_1 = \emptyset$ due to big errors in vector $b$. It should be more advisable to compute the analytic center of the optimal solution set $S$ of problem (5) instead of solving problem (1) directly for numerical stability, since the matrix $[A, I]$ is well-conditioned even when matrix $A$ is very ill-conditioned (here we assume that the matrix $A$ is scaled in advance).
For the general case, we have the following result.

**Theorem 2.7.** Let $S$ be the optimal solution set of problem (5), and $\sigma_x(S) = \{ i : \exists (x, y) \in S \text{ such that } x_i > 0 \}$, and $\sigma_y(S) = \{ j : \exists (x, y) \in S \text{ such that } y_j > 0 \}$. Then the analytic center of $S$ is the unique point $(\bar{x}, \bar{y}) \in S$ which maximize $\prod_{i \in \sigma_x(S)} x_i \prod_{j \in \sigma_y(S)} y_j$ with $(x, y) \in T$ (here define $\prod_{i \in I} x_i = 1$ if $I = \emptyset$), and the point $\bar{x}$ is the unique analytic center of the polytope $P_{\bar{y}} = \{ x \mid Ax = b - \bar{y}, x \geq 0 \}$.

**Proof.** The first part of the proposition is obtained by definition. For every $x \in P_{\bar{y}}$, it is easy to see that $(x, \bar{y}) \in S$, and the second part follows.

**Remark 2.4.** By Theorem 2.7, we can always solve problem (1) by finding the analytic center $(\bar{x}, \bar{y})$ of the optimal solution set $S$ of problem (5) even when the polyhedral set $P_1 = \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \}$ is empty or has no interior points. When the set $P_1$ is nonempty, we always have $\bar{y} = 0$ by Theorem 2.6, and $\bar{x}$ is the analytic center of the set $P_1$. When the set $P_1$ is empty, we will have $\bar{y} \neq 0$ and $\bar{x}$ is the analytic center of the set $P_{\bar{y}} = \{ x \mid Ax = b - \bar{y}, x \geq 0 \}$ by Theorem 2.7. In this case, the introduced variable $y$ can be seen as a correction of the errors in vector $b$. Hence we would rather compute the analytic center of the optimal solution set of problem (5) than solve problem (1) directly for numerical stability.

3. **The primal-dual unfolding procedure.** In this section we discuss how to compute the analytic center of the optimal solution set of problem (5) efficiently and robustly. It is fortunate that the theory of interior point methods for linear programming (see, e.g., [26, 37, 5] and references therein) has established abundant results on the computation of the analytic center of a polytope, especially primal-dual methods (see, e.g., [35]). The basic primal-dual interior algorithm was proposed by Kojima, Mizuno and Yoshise [16]. The most efficient implementation of interior point methods at the moment seems to be some variant of Mehrotra’s infeasible predictor-corrector primal-dual interior point method [19]. However, it is not guaranteed in theory that the iteration sequence generated by the algorithm will converge to the analytic center of the optimal solution set. Soon after that, Mizuno, Todd, and Ye [22] proposed another predictor-corrector method with good theoretical properties, which will be called as “MTY algorithm” in the rest of the paper. Ye et al. [36] and Mehrotra [20] independently proved that the sequence generated by the MTY algorithm is quadratically convergent in duality gap. Bonnans and Gonzaga [4], Gonzaga and Tapia [12] proved that the iteration sequence generated by the MTY algorithm converges to the analytic center of the optimal solution set. González-Lima, Tapia, and Potra [13] also proposed a variant of Kojima-Mizuno-Yoshise primal-dual algorithm which can guarantee the sequence generated converges to the analytic center based on the results established in [38]. Since we need to compute the analytic center of the solution set, and the quartadratic convergence property is important for numerical stability when the problem to be solved is difficult, the MTY algorithm would be a good choice for problem (1). However, in our computation we found out that it is very difficult to get an initial interior point for the MTY algorithm to start, and the techniques proposed in [13] failed in our computation. Hence a more robust and efficient algorithm is needed. For this purpose, we will adopt the infeasible-interior-point (IIP) techniques to compute an initial point for the MTY algorithm to start, and then use the MTY algorithm to compute the analytic center of the optimal solution set of problem (5).
Many papers have analysed the properties of IIP methods. The IIP methods were firstly proposed in [19, 17]. Then Kojima, Megiddo and Mizuno [15] proved the global convergence of a primal-dual IIP algorithm for LP. Zhang [39] proved the worst case iteration complexity $O(n^2L)$ of a primal-dual IIP algorithm. Soon after that, Mizuno [21] also proved that a modification of the algorithm in [15] has $O(n^2L)$ iteration complexity, and proposed a predictor-corrector IIP algorithm with $O(nL)$ iteration complexity. The book [35] gave a detailed description of IIP methods. Zhang [40] discussed an implementation of a variant of Mehrotra’s predictor-corrector primal-dual infeasible interior point method under MATLAB environment.

Now we describe some basic concepts in the interior point algorithms for linear programming for completeness. The dual problem of (5) is

$$
\begin{align*}
\max_{w,r,s} \quad & b^T w \\
\text{s.t.} \quad & A^T w + r = 0, \\
& w + s = e, \\
& r \geq 0, s \geq 0.
\end{align*}
$$

(6)

By Assumption 2.1, it is easy to see that the point $w^{(0)} = -e, s^{(0)} = 2e > 0, r^{(0)} = -A^T w^{(0)} = A^T e > 0$ is an interior point of the dual problem (6). Hence the interior point sets of problems (5) and (6) both are nonempty. Then by the well-known interior point condition, for every $\mu > 0$, the following system

$$
\begin{align*}
Ax + y &= b, \\
A^T w + r &= 0, \\
w + s &= e, \\
xr &= \mu e, r \geq 0, x \geq 0, \\
sy &= \mu e, s \geq 0, y \geq 0,
\end{align*}
$$

(7)

has a unique solution $z(\mu) = (x(\mu), y(\mu), w(\mu), r(\mu), s(\mu))$, and the curve $z(\mu) = (x(\mu), y(\mu), w(\mu), r(\mu), s(\mu))$ for $\mu > 0$ is called as the central path for problem (5) and its dual (6). When $\mu = 0$, system (7) is just the KKT system of problem (5) and its dual (6). Hence system (7) is also called as the perturbed KKT system of problem (5).

In order to solve (7) by Newton’s method, the following linear system is formed at a positive point $z = (x, y, w, r, s) > 0$ (which may be feasible or infeasible) for computing the search direction $\Delta z = (\Delta x, \Delta y, \Delta w, \Delta r, \Delta s)$.

$$
\begin{align*}
A\Delta x + \Delta y &= -r_b, \\
A^T \Delta w + \Delta r &= -r_c, \\
\Delta w + \Delta s &= -r_e, \\
R\Delta x + X\Delta r &= \sigma \mu e - x r, \\
Y \Delta s + S \Delta y &= \sigma \mu e - s y.
\end{align*}
$$

(8)

where $r_b = Ax + y - b, r_c = A^T w + r, r_e = w + s - e$. The parameter $\sigma \in [0, 1]$ is a factor to decrease the duality gap, and $R, X, Y, S$ are diagonal matrices of vectors $r, x, y, s$, respectively.
For any primal-dual feasible interior point \( z = (x, y, w, r, s) \), we define its proximity to the central path for some parameter \( \mu > 0 \) as follows
\[
\delta(z, \mu) \equiv \delta(x, y, w, r, s, \mu) \triangleq \| \begin{bmatrix} \frac{x}{\mu} \\ \frac{y}{\mu} \end{bmatrix} - e \|,
\]
and denote
\[
\mu(z) \equiv \mu(x, y, w, r, s) \triangleq \frac{x^T r + y^T s}{n + m}, \quad (10)
\]
\[
\delta(z) \equiv \delta(x, y, w, r, s) \triangleq \delta(x, y, w, r, s, \mu(x, y, w, r, s)).
\]
Two kinds of neighborhoods of the central path are usually used in interior point methods, which are defined as:
\[
\mathcal{N}_2(\beta) \triangleq \{ z = (x, y, w, r, s) \mid \delta(z) \leq \beta \}, \quad (12)
\]
\[
\mathcal{N}_\infty(\gamma) \triangleq \{ z = (x, y, w, r, s) \mid (x, y, r, s) > 0, x^T \geq \gamma^\mu e, y^T \geq \gamma^\mu e \}, \quad (13)
\]
The sets \( \mathcal{N}_2(\beta) \) and \( \mathcal{N}_\infty(\gamma) \) are usually called as the narrow neighborhood and the wide neighborhood of the central path respectively.

Based on the primal-dual IIP methods for linear programming and the MTY algorithm, we design a hybrid algorithm for computing the analytic center of the optimal solution set \( \mathcal{S} \) of problem (5). Firstly we modify the IIP techniques proposed in [19, 17, 15, 40] (see also [35]) to compute a primal-dual feasible interior point for the MTY algorithm to start. Then we call the MTY algorithm to compute the analytic center of the solution set.

In the following algorithm, we denote the residuals \( r_b^{(k)} = Ax^{(k)} + y^{(k)} - b, r_c^{(k)} = A^T w^{(k)} + r^{(k)}, r_e^{(k)} = w^{(k)} + s^{(k)} - e \). Since \( z^{(k+1)} = z^{(k)} + \alpha_k \Delta^{(k)} \), by (8) we get
\[
\begin{align}
r_b^{(k+1)} &= (1 - \alpha_k) r_b^{(k)} = \nu_{k+1} r_b^{(0)}, \quad (14) \\
r_c^{(k+1)} &= (1 - \alpha_k) r_c^{(k)} = \nu_{k+1} r_c^{(0)}, \quad (15) \\
r_e^{(k+1)} &= (1 - \alpha_k) r_e^{(k)} = \nu_{k+1} r_e^{(0)}. \quad (16)
\end{align}
\]
where
\[
\nu_k = (1 - \alpha_{k-1}) \nu_{k-1} = \prod_{i=0}^{k-1}(1 - \alpha_i), k = 1, 2, \ldots, \quad (17)
\]
and \( \nu_0 = 1 \).

Let \( (x^{(0)}, y^{(0)}) \) be a positive feasible point of \( Q \), and \( w^{(0)} = 0, r^{(0)} = e, s^{(0)} = e \). Then \( z^{(0)} = (x^{(0)}, y^{(0)}, w^{(0)}, r^{(0)}, s^{(0)}) \) is an initial infeasible interior point of system (7) and we have:
\[
\begin{align}
r_b^{(0)} &= Ax^{(0)} + y^{(0)} - b = 0, \\
r_c^{(0)} &= A^T w^{(0)} + r^{(0)} = e, \quad (18) \\
r_e^{(0)} &= w^{(0)} + s^{(0)} - e = 0.
\end{align}
\]
By (14)–(18), at the \( k \)-th iteration we will have
\[
r_b^{(k)} = 0, r_c^{(k)} = \nu_k e, r_e^{(k)} = 0, \quad k = 1, 2, \ldots, \quad (19)
\]
Hence we have \( \| r_c^{(k)} \|_\infty = \nu_k \), and the point \( z^{(k)} \) would be approximately feasible when \( \nu_k \leq \varepsilon \).

The details of our method are presented in the following algorithm.

Algorithm 3.1 A Primal-Dual Unfolding Procedure for Problem (1) (PDUP)
Step 0. Scale the matrix \( A = (a_{ij})_{m \times n} \), i.e., for \( i = 1, \ldots, m \), compute \( t_i = \sum_{j=1}^{n} a_{ij} \). If \( t_i > 1 \), set \( a_{ij} \leftarrow a_{ij}/t_i \) for \( j = 1, \ldots, n \), and \( b_i \leftarrow b_i/t_i \). Set the convergence tolerance parameter \( \varepsilon > 0 \), the parameter \( \beta \in [0.1, 0.26] \).

Compute an interior point \( (x^{(0)}, y^{(0)}) \) of the set \( P_2 \) by the way described in the proof of Lemma 2.4, i.e., find a positive proper real number \( h_0 \) such that \( x^{(0)} = h_0 \varepsilon \) and \( y^{(0)} = b - Ax^{(0)} > 0 \). Let \( w^{(0)} = 0, r^{(0)} = e, s^{(0)} = e \). Set \( r_b^{(0)} = 0, r_c^{(0)} = A^T w^{(0)} + r^{(0)} = e, r_e^{(0)} = 0, \sigma_0 = 1 \). Compute \( \mu_0 = \mu(z^{(0)}) = (z^{(0)})^T (z^{(0)}) + (s^{(0)})^T s^{(0)} \) according to formula (10) and \( \delta_0 = \delta(z^{(0)}) \) according to formula (11). Set \( \gamma \in [10^{-8}, 10^{-3}], \mu = \min\{0.1\mu_0, 1\} \). Set \( \hat{\mu} = \mu_0 \) and \( k = 0 \).

Step 1. Use the following IIP method which is modified from IIP techniques described in [19, 17, 15, 35, 40] to generate iteration sequence until we get a point \( z^{(k)} = (x^{(k)}, y^{(k)}, w^{(k)}, r^{(k)}, s^{(k)}) \) such that \( |r_c^{(k)}|_\infty \leq \varepsilon \) and \( \delta_k \triangleq \delta(z^{(k)}) \leq \beta \), where \( r_c^{(k)} = A^T w^{(k)} + r^{(k)} \) is the residual defined in (8), and \( \delta_k = \delta(z^{(k)}) \) is defined in (11).

Step 1.1 If \( \delta_k \leq \delta(z^{(k)}) \leq \beta \), then we can use backtracking technique to compute the smallest nonnegative integer \( p \) with \( \alpha_k = \theta^p \alpha_{max} \) (where \( \theta < 1 \)) such that

\[
\begin{align*}
\zeta &= \frac{1}{\min(t, -0.5)}, \\
\alpha_{max} &= \min\{1, 0.9995 \times \zeta\}.
\end{align*}
\]

Step 1.2 Let \( \zeta = z^{(k)}, \sigma = \sigma_k, \mu = \hat{\mu}, r_b = 0, r_c = r_c^{(k)}, r_e = 0 \) and solve (8) to get the Newton search direction

\[
\Delta_k = (\Delta x^{(k)}, \Delta y^{(k)}, \Delta w^{(k)}, \Delta r^{(k)}, \Delta s^{(k)}).
\]

Step 1.3 If \( \alpha_{max} = 1 \), set \( z^{(k+1)} = z^{(k)} + \Delta_k \). Otherwise use backtracking technique to compute the smallest nonnegative integer \( p \) with \( \alpha_k = \theta^p \alpha_{max} \) (where \( \theta < 1 \)) such that

\[
\begin{align*}
\hat{\mu} \leq \mu(z^{(k+1)}),
\end{align*}
\]

Moreover, if \( \delta_k > \beta \) with \( \|r_c^{(k)}\|_\infty = \nu_k \leq \varepsilon \), the new point \( z^{(k+1)} \) should also satisfy

\[
G(z^{(k+1)}) \leq (1 - 0.01\alpha_k)G(z^{(k)}),
\]

where \( G(z) = \|v - \mu(z)e\| \) with \( v = \begin{bmatrix} x^r \\ y^s \end{bmatrix} \).

Step 1.4 Compute \( \nu_{k+1} = (1 - \alpha_k)\nu_k, r_c^{(k+1)} = \nu_{k+1} e \). Compute \( \mu_{k+1} = \mu(z^{(k+1)}) \) and \( \delta_{k+1} = \delta(z^{(k+1)}) \) according to (10)-(11).

If \( \nu_k \geq \tau_0 \) and \( \nu_{k+1} < \tau_0 \), set \( \hat{\mu} = \max\{\mu_{k+1}, 2\hat{\mu}\} \).

If \( \nu_{k+1} < \tau_0 \), set \( \hat{\mu} = \hat{\mu}, \sigma_{k+1} = 1 \). Otherwise, if \( \mu_{k+1} > 10\hat{\mu} \) and \( \alpha_k > 0.5 \) with \( \nu_{k+1} \geq \tau_0 \), set \( \sigma_{k+1} = 0.2, \hat{\mu} = \mu_{k+1} \); otherwise \( \mu_{k+1} \leq 10\hat{\mu} \) or \( \alpha_k \leq 0.5 \) with \( \nu_{k+1} \geq \tau_0 \), set \( \sigma_{k+1} = 1, \hat{\mu} = \max\{\mu_{k+1}, 2\hat{\mu}\} \).

Set \( k \leftarrow k + 1 \) and go to step 1.

Step 2. Do the MTY predictor-corrector algorithm as follows.

Step 2.1 (Predictor) Set \( \gamma = 0, z = z^{(k)} = (x^{(k)}, y^{(k)}, w^{(k)}, r^{(k)}, s^{(k)}) \) and solve (8) to get the search direction \( \Delta z = (\Delta x, \Delta y, \Delta w, \Delta r, \Delta s) \).
Step 2.2 Compute the largest \( \bar{\alpha} \in (0, 1] \) such that \( \hat{z} = z + \bar{\alpha}\Delta z \in \mathcal{N}(2\beta) \), which can be done by solving two quadratic equations in one variable.

Step 2.3 Set \( \mu_k \leftarrow (1 - \bar{\alpha})\mu_k \).

Step 2.4 Set \( \gamma = 1, \mu = \mu_k, z = \hat{z} \) and solve (8) to get the search direction \( \Delta z = (\Delta x, \Delta y, \Delta w, \Delta r, \Delta s) \), and set \( \bar{z} = z + \Delta z \).

Step 2.5 If \( (n + m)\mu < \varepsilon \), stop and return the solution; Otherwise, set \( k \leftarrow k + 1, z^{(k)} = \bar{z}, \mu_k = \mu \) and go to Step 2.1.

Remark 3.1. In Step 1 of Algorithm 3.1, we use a modified IIP method to compute a primal-dual feasible point whose proximity to the central path is suitable for the MTY algorithm to start. Since the main purpose of Step 1 is to compute an initial point for the MTY algorithm, we will keep the duality gap \( \mu_k \) bigger than a positive constant \( \bar{\mu} \) in practical implementation, i.e., when \( \mu_k \leq \bar{\mu} \), we will stop decreasing the duality gap \( \mu_k \). Since we always have \( \sigma_{k+1}\mu \geq 2\bar{\mu} \) according to Step 1.4, by the techniques established in [15, 39, 21], there will always exist some \( \alpha_k \geq \bar{\alpha} \) (where \( \bar{\alpha} \) is a positive constant) such that formulae (20)–(21) are satisfied. It is easy to see that when \( \|r^{(k)}\|_{\infty} = \nu_k \leq \varepsilon \), the parameter \( \sigma_{k}\mu_k \) is fixed to \( \bar{\mu} \geq 2\bar{\mu} \) according to Step 1.4. In this case we are in fact using Newton method to solve nonlinear system (7) with fixed \( \mu \). With the help of formula (21), we will eventually have \( \delta_k \to 0 \) as \( k \to +\infty \). Hence the sub-loops in Step 1 will terminate finitely according to the theory for IIP described in [39, 21, 35]. Another advantage for fixing the parameter \( \mu \) to a positive constant in Step 1 is that we can effectively avoid numerical difficulties caused by the ill-conditioning of the matrices formed in primal-dual interior point methods. After that we call the MTY algorithm to compute the analytic center of the optimal solution set of problem (5). Step 2.1 is a predictor step when the current point is near the central path, and leave away the central path to decrease the duality gap in Step 2.2– Step 2.3. Step 2.4 is the corrector step in the MTY algorithm, which can guarantee the iteration point to get close to the central path enough again. In practical computation, we usually set \( \beta = 0.26 \), since with bigger \( \beta \), we usually get better performance. When \( \beta < 0.1 \), we find that the duality gap parameter \( \mu_k \) will decrease very slow before the quadratic convergence is observed in some computations.

According to the convergence theory for the MTY algorithm established in [36, 20, 4, 12], we know that the iteration sequence generated in Step 2 of Algorithm 3.1 will converge to the analytic center of the optimal solution set of problem (5), which is just the unique solution of problem (1) when it has a nonempty interior by Lemma 2.2 and Theorem 2.6, or the unique corrected analytic center as described in Theorem 2.7. Moreover, the worst case complexity of the algorithm is polynomial, and the local convergence rate in the duality gap is quadratic.

4. Numerical results. In this section, we will present some numerical results obtained by Algorithm 3.1. All the numerical tests are done under the MATLAB environment in Windows 7 system on a laptop computer. Numerical results obtained by Algorithm 3.1 will be denoted as “PDUP” in the following paragraphs and figures for convenience, and the parameters of Algorithm 3.1 are set as \( \varepsilon = 10^{-6}, \beta = 0.26 \) for all test problems.

Firstly we construct some test problems for academic purposes by using the well-known Hilbert matrices, which are seriously ill-conditioned as the dimension increases. Letting \( H_m \) be an \( m \times m \) Hilbert matrix, i.e., we have \( H = \left( \frac{1}{i+j-1} \right)_{m \times m} \).
Consider the following ill-conditioned polytope
\[ P_m = \{ x \in \mathbb{R}^n \mid Ax = b, x \geq 0 \}, \]
where \( A = A_m \triangleq [H_m, H_m], b = H_m e \), and \( n = 2m \). It is easy to check that
the analytic center of \( P_m \) is \( \bar{x} = (0.5, \ldots, 0.5) \), and the condition number of \( A_m \)
increases as \( m \) increases.

For comparison, we will use the interior point method used in MATLAB’s optimization tool \textit{linprog} to solve the following problem:
\[
\min_{x} 0^T x \\
\text{s.t.} \quad A_m x = b, \quad x \geq 0. \tag{22}
\]
The corresponding numerical results is denoted by “linprog-1”. We also use the interior point method in \textit{linprog} to solve problem \( (2) \) with \( A = A_m \), and denote corresponding numerical results by “linprog-2”. We use the function “optimoptions” in MATLAB to set the options of the tool \textit{linprog}:
\[
\text{options=optimoptions('linprog','Algorithm','interior-point', ...} \\
\quad \text{'Display','iter','OptimalityTolerance',1e-6,'maxiter',1000);}
\]
Then we call the optimization tool \textit{linprog} in MATLAB by using the parameter \textit{options}.

We also use the interior methods in MATLAB’s optimization tool \textit{fmincon} to compute the following problem
\[
\min_{x} -\sum_{j=1}^{n} \ln x_j \\
\text{s.t.} \quad Ax = b, \quad x \geq 0. \tag{23}
\]
The corresponding numerical results will be denoted by “fmincon”. The options for \textit{fmincon} is set as follows:
\[
\text{options=optimoptions(@fmincon,'SpecifyObjectiveGradient',true, ...} \\
\quad \text{'Display','iter', 'Algorithm','interior-point',...} \\
\quad \text{'OptimalityTolerance', 1e-6, 'maxiter', 1000);}
\]
which means that \textit{fmincon} will use its interior-point algorithm to compute.

Numerical results for computing the analytic center of \( P_m \) are collected in Table 1. The column “m” in the table is the dimension of the Hilbert matrix \( H_m \). The column “Algo.” indicates which method is used, which are “lingprog-1”, “lingprog-2”, “fmincon”, and “PDUP” respectively. The columns \( f \) and \( g \) are the function values of \( f(x) = -\sum_{j=1}^{n} \ln x_j \) and \( g(x) = \|Ax-b\| \) at the numerical solution \( \bar{x} \) obtained by the corresponding algorithm respectively. The function value \( f(\bar{x}) \) indicates whether the solution \( \bar{x} \) is the analytic center, and \( g(\bar{x}) \) indicates whether the solution \( \bar{x} \) is feasible. In these two columns, the symbol “-” means that the corresponding method failed to obtain a solution. In column “\( f \)”, the symbol “+\( \infty \)” means that there exists at least one \( x_j \) whose value is 0 in the obtained solution \( \bar{x} \), which means that the solution obtained cannot be the analytic center. The column “No.” represents the iteration number of the corresponding algorithm needed to obtain the solution \( \bar{x} \). The column “Opt.” indicates whether the corresponding method stops with its optimality measure being satisfied, with the symbol “Y” meaning “Yes”,

\[
\begin{array}{|c|c|c|c|c|}
\hline
\text{m} & \text{Algo.} & \text{f} & \text{g} & \text{No.} & \text{Opt.} \\
\hline
\end{array}
\]
the symbol “N” meaning “No”, and the symbol “P” meaning “possible optimal” (Under MATLAB, the situation in indicated by the parameter “exitflag”). The column “Time” is the CPU time (in seconds) needed by the corresponding algorithm to compute the problem.

Table 1. Numerical Results for $A_m = [H_m, H_m]$, $b = H_m e$

| $m$ | Algo.   | No. | $f$      | $g$          | Opt. | Time(s) |
|-----|---------|-----|----------|--------------|------|---------|
| 10  | linprog-1 | 6   | $+\infty$ | $4.110e-07$ | Y    | 0.235   |
|     | linprog-2 | 5   | $+\infty$ | $1.421e-09$ | Y    | 0.243   |
|     | fmincon  | 39  | 13.863   | 1.286e-13   | Y    | 1.392   |
|     | PDUP     | 11  | 13.863   | 8.674e-09   | Y    | 0.071   |
| 20  | linprog-1 | 1001| —        | —            | N    | 0.285   |
|     | linprog-2 | 5   | $+\infty$ | $1.101e-08$ | Y    | 0.266   |
|     | fmincon  | 15  | 27.726   | 1.528e-13   | Y    | 0.763   |
|     | PDUP     | 13  | 27.726   | 3.589e-11   | Y    | 0.092   |
| 50  | linprog-1 | 1001| —        | —            | N    | 0.820   |
|     | linprog-2 | 6   | $+\infty$ | $6.305e-08$ | Y    | 0.241   |
|     | fmincon  | 14  | 69.315   | 2.336e-13   | Y    | 1.021   |
|     | PDUP     | 16  | 69.315   | 1.780e-10   | Y    | 0.127   |
| 100 | linprog-1 | 337 | —        | —            | N    | 1.062   |
|     | linprog-2 | 5   | $+\infty$ | $1.299e-07$ | Y    | 0.246   |
|     | fmincon  | 13  | 138.629  | 1.483e-13   | Y    | 1.117   |
|     | PDUP     | 18  | 138.629  | 1.096e-10   | Y    | 0.390   |
| 300 | linprog-1 | 1001| —        | —            | N    | 41.905  |
|     | linprog-2 | 19  | —        | —            | N    | 1.401   |
|     | fmincon  | 156 | 415.888  | 3.855e-13   | N    | 166.530 |
|     | PDUP     | 22  | 415.888  | 2.379e-11   | Y    | 3.600   |
| 500 | linprog-1 | 676 | —        | —            | N    | 149.613 |
|     | linprog-2 | 8   | —        | —            | N    | 3.548   |
|     | fmincon  | 77  | 693.147  | 6.652e-13   | P    | 385.377 |
|     | PDUP     | 24  | 693.147  | 6.288e-11   | Y    | 11.828  |

By the results in Table 1, we can see that only for $m = 10$, the method “linprog-1” succeeded in obtaining the optimal solution among all the test problems. For other problems it failed, and at the end of its computation it displayed either

*Solver stopped prematurely. Linprog stopped because it exceeded the iteration limit, options.MaxIterations = 1000.*

when the iteration No. = 1000, or

*Linprog stopped because it was unable to find a point that satisfies the constraints within the default value of the constraint tolerance.*

when the iteration No. < 1000. By these numerical results, we can see that the ill-conditioning of matrix $A$ indeed play an important role in the computation of interior point methods.

The method “linprog-2” succeeded in obtaining optimal solutions according to its optimization measure for $m = 10, 20, 50, 100$. But it failed for $m = 300, 500$, and displayed similar messages as the method “linprog-1”. Comparing numerical
results obtained by “linprog-1” with “linprog-2”, we can see that problem (2) is much more stable than problem (22), as indicated by Lemma 2.3. By numerical results obtained by “linprog-2”, we can see that problem (2) is much more stable than problem (22), as indicated by Lemma 2.3. By numerical results obtained by “linprog-2”, we can see that the interior method used in linprog, which is based on LIPSOL [40], cannot guarantee the iteration sequence generated converges to the analytic center of the optimal solution set.

When we used the tool fmincon to solve problem (23), it also met some numerical difficulties during computation and displayed the following message

Warning: Matrix is close to singular or badly scaled. Results may be inaccurate. RCOND = 1.013015e-016.

or something alike. However, the method fmincon still obtained optimal solutions for \( m = 10, 20, 50, 100, 300 \) according to its optimization measure, i.e., stopped with the “exitflag = 1”. For \( m = 500 \), the method fmincon stopped with the “exitflag = 2”, and displayed the following message:

Local minimum possible. Constraints satisfied.
Change in x less than options.StepTolerance and maximum constraint violation less than options.ConstraintTolerance.

For all test problems, the method fmincon got the approximate analytic center according to the column \( f \) in Table 1. However, its CPU time is much more than our method “PDUP” for all test problems. For every test problem, our method “PDUP” terminated with the optimal rule being satisfied, and got the approximate analytic center with the least CPU time (in seconds). From the numerical results in Table 1, we can see that the performance of our method “PDUP” is the most effective and robust one.

Now we try to use our method to unfold neutron energy spectrum on two data sets which are obtained from Northwest Institute of Nuclear Technology of P. R. China. The first data set is named as “data1” which consists of matrix \( A = (a_{ij})_{23 \times 640} \), vector \( b \in \mathbb{R}^{23} \), vector \( E \in \mathbb{R}^{640} \), and vector \( \Delta E \in \mathbb{R}^{640} \). The second data set is named as “data2” which includes matrix \( A = (a_{ij})_{21 \times 640} \), vector \( b \in \mathbb{R}^{21} \), vector \( E \in \mathbb{R}^{640} \), and vector \( \Delta E \in \mathbb{R}^{640} \). In both of the data sets, matrix \( A = (a_{ij})_{m \times n} \) represents the nuclear reaction microscopic cross sections and vector \( b \in \mathbb{R}^{m} \) represents the mononuclear saturation activities, where \( m \) is the number of foils which we call nuclides in neutron physics and \( n \) is the number of energy groups which are determined by a standard procedure. Vector \( E \) is a column vector, whose component \( E_j \) (\( j = 1, \ldots, 640 \)) is the energy corresponding to the \( j \)-th energy group. The \( j \)-th element of vector \( \Delta E \) is the energy gap between adjacent energy groups.

The data set “data1” is obtained by applying the principle of neutron activation, where the standard neutron spectrum is employed which is a piecewise function made up of the Maxwell thermal spectrum, the \( 1/E \) spectrum and the Watt fission spectrum. It is formulated as:

\[
\varphi_1(E_j) = \begin{cases} 
1 \times 10^6 \cdot E_j / (0.025 \cdot 0.025) \cdot e^{-E_j / 0.025}, & E_j \leq 5 \times 10^{-2} eV; \\
5.4 \times 10^5 / E_j, & 5 \times 10^{-2} eV \leq E_j \leq 5 \times 10^5 eV; \\
1.5 \cdot \exp (-E_j) \cdot \sinh(\sqrt{2E_j}), & E_j \geq 5 \times 10^5 eV.
\end{cases}
\]

From the standard procedure, we can calculate the nuclear reaction microscopic cross section to obtain the matrix \( A \), and the vector \( b \) is obtained by computing \( Ax \) in the data set “data1”. Namely, we have already known the theoretical neutron spectral distribution in “data1”. However, the data set “data2” comes from a real experiment where the matrix \( A \) is calculated and the vector \( b \) is measured. The
inherent error of the instrument and the test error in the process of experiment will result in the inaccuracy of the vector $b$. For both data sets, the $i$-th row of the matrix $A$, which is the microscopic cross sections’ values on different energy groups, can oscillate sharply within a range of $10^{-15}$ to $10^5$. The 2-norm condition number of the matrix $A$ are $1.7455 \times 10^7$ and $1.2718 \times 10^7$ in “data1” and “data2” respectively. Hence the resulting matrix $A$ is ill-conditioned in both data sets. The vector $b \in \mathbb{R}^m$ represents the mononuclear saturation activities on $m$ nuclides, whose elements vary from $7.72 \times 10^{-3}$ to $3.34 \times 10^{12}$ and $6.64 \times 10^6$ to $5.92 \times 10^{11}$ in “data1” and “data2”, respectively. In order to get one accurate vector $b$, the instrument called multi foil activation spectrometer is required to be highly precise and highly sensitive. However, because of the inherent error of the instrument itself and inappropriate operations, there is always a deviation between the vector $b$ measured by this instrument and the true values. This deviation may cause problem (1) to be infeasible or a large bias from the optimal solution. By introducing the error variable $y$, we eventually eliminate the effect of measurement error of the vector $b$ on problem (1).

Actually, the phenomenon discussed above is common in the process of unfolding spectrum. Hence the introduction of the variable $y$ into problem (2) as a correction of the errors in the vector $b$ should reasonable and meaningful in practice.

The solution of problem (1) is not the neutron spectrum which we utilize for manufacturing devices in neutron physics. The component $x_i$ is the integral of neutron spectrum in a definite energy interval, and $\Delta E_i$ is the integral injection rate of neutrons. The number $x_i/\Delta E_i$ ($i = 1, \ldots, n$) is a discrete approximation of the neutron spectrum defined in $n$ energy groups, which is called as the differential injection rate of neutrons. Therefore, we choose $x_i/\Delta E_i$ as the ordinate and $E_i$ as the abscissa to expose the distribution of neutron spectrum in the following figures. Since the variation range of energy and the spectrum magnitude is very large, we will utilize double logarithm coordinate system in following figures to present the experimental results, which would be more visually in details.

In Figure 1 we denote the standard theoretical neutron spectrum as “XOPT” and the solution obtained by Algorithm 3.1 as “PDUP”, which are depicted as two curves. The abscissa is the energy vector $E$ in logarithm whose elements correspond to the 640 energy groups, and the ordinate is the vector $X = x./\Delta E \in \mathbb{R}^n$ in logarithm which approximates the neutron spectrum defined in the 640 energy groups. The curve PDUP overlaps with the curve XOPT almost completely in the energy interval $[10^{-4}, 10^{-1}]$ (which is in the low energy region), and in the energy interval $[10^5, 10^7]$ (which belongs to high energy region). But the curve PDUP is more volatile compared with the curve XOPT in the energy interval $[10^1, 10^4]$, which is a portion of intermediate energy region and corresponds to 200-th to 400-th energy groups. The matrix $A$ also oscillates sharply in the energy interval $[10^1, 10^4]$, which may cause the solution fluctuates in the intermediate energy region.

Figure 2 presents similar results for the experimental data set “data2”. Although we have no idea about the standard spectrum on the data set “data2”, researchers can employ the solution obtained by the algorithm SAND-II which is used extensively in practice. The algorithm SAND-II establish objective function based on the least squares principle and then minimize $||Ax-b||^2$ iteratively until the termination conditions are satisfied, which utilize all the information available on the spectrum. Hence the algorithm SAND-II requires a priori knowledge on the spectrum, which is called as the initial spectrum and has a key impact on the results obtained by the
software SAND-II. The procedure to obtain the initial spectrum is rather complicated in operations. In Figure 2 we denote the solution obtained by the algorithm SAND-II as “SAND-II”, the initial spectrum for the algorithm SAND-II as “SAND-II-initial”, and the solution obtained by our method—Algorithm 3.1 as “PDUP”. Similar as the data set “data1”, the curve PDUP reflects the distribution of neutron spectrum which is consistent with the curve SAND-II in general. But the deviation between PDUP and SAND-II is not small in low energy region and the shapes of two curves is also distinct. Besides, there is a large oscillation of PDUP in local intermediate energy region. The oscillation of PDUP in intermediate energy region is also due to the sharp oscillations of the matrix $A$ in the corresponding energy region. The SAND-II-initial contains a lot of a priori information about neutrons which determines the shape of the curve SAND-II in low energy region. Our algorithm makes no use of any such information, so the curve PDUP has different structure in the low energy region which is from $10^{-4}$ to $10^{-1}$.

By the results presented above, we can see that our algorithm is effective and robust for the ill-conditioned problem (1) which has associated errors in the right hand side $b$. The above results show that our method is promising. The advantage of our unfolding procedure proposed above is that it does not dependent on any initial neutron spectrum, and the results obtained by our algorithm PDUP in general reflects the neutron spectrum despite some fluctuations which may need some further studies.

Figure 1. The standard spectrum and the spectrum solved by PDUP for “data1” in logarithmic coordinates.
5. Conclusions. In this paper, we have proposed a stable numerical method for the computation of problem (1) which arises in the unfolding procedure for neutron energy spectrum from multiple activation foils. Problem (1) is built up based on the maximum entropy principle of the thermodynamic entropy theory and Boltzmann’s entropy formula. In problem (1), the matrix $A \in \mathbb{R}^{m \times n}$ is ill-conditioned and the right hand side $b$ is inaccurate. We have tried techniques proposed in [13] and some software such as LIPSOL [40] and IPOPT [29] to solve the two problems of the form (1) presented in Section 4, but the numerical results obtained by them are poor due to the ill-conditioning of the matrix $A$ and the associated errors in the vector $b$. After several attempts, we finally derived an elegant reformulation of problem (1). We first introduced a variable $y$ to tackle the errors in the right hand side $b$ by the exact penalty method. Then we analysed the properties of the transformed problem, and proved that the condition number of the coefficients matrix $[A, I]$ is bounded by $\sqrt{m}$. We finally found out that we only need compute the analytic center of the optimal solution set of the well-conditioned linear programming problem (5), which is much easier to tackle than problem (1). Then based on the primal-dual infeasible interior point methods and the MTY algorithm for linear programming, we proposed a robust hybrid primal-dual interior-point algorithm for the ill-conditioned problem (1) with errors in the vector $b$. Some numerical test results were presented and compared with SAND-II in Section 4. The numerical results in general coincide with the results obtained by SAND-II well despite some fluctuations, which showed the effectiveness and robustness of our Algorithm 3.1 for problem (1). The new method
we got can also be used to compute the analytic center of general ill-conditioned polytopes, which is an important problem in some applications.

Our work provided a new computation tool for the neutron energy spectrum unfolding problem, which is important in nuclear science and high energy physics. The advantage of our method is that we do not need any a priori knowledge about the initial spectrum, and it can be used to infer the true neutron energy spectrum with other unfolding methods. Although there are still some fluctuations in the numerical results which may be caused by the sharp oscillations of the matrix $A$, the numerical results presented in section 4 showed that the new method is indeed worth further studies.

REFERENCES

[1] D. S. Atkinson and P. M. Vaidya, A scaling technique for finding the weighted analytic center of a polytope, Mathematical Programming, 57 (1992), 163–192.

[2] E. A. Belogorlov and V. P. Zhigunov, Interpretation of the solution to the inverse problem for the positive function and the reconstruction of neutron spectra, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 235 (1985), 146–163.

[3] D. P. Bertsekas, Necessary and sufficient conditions for a penalty method to be exact, Mathematical Programming, 9 (1975), 87–99.

[4] J. F. Bonnans and C. C. Gonzaga, Convergence of interior point algorithms for the monotone linear complementarity problem, Mathematics of Operations Research, 21 (1996), 1–25.

[5] S. Boyd and L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, 2004.

[6] E. J. Candes, J. Romberg and T. Tao, Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information, IEEE Transactions on Information Theory, 52 (2006), 489–509.

[7] X. Chen, Z. Lu and T. K. Pong, Penalty methods for a class of non-lipschitz optimization problems, SIAM Journal on Optimization, 26 (2016), 1465–1492.

[8] G. Cowan, A Survey of Unfolding Methods for Particle Physics, 2002. Available from: https://www.ippp.dur.ac.uk/old/Workshops/02/statistics/proceedings/cowan.pdf.

[9] F. Z. Dehimi, A. Seghour and S. E. H. Abaidia, Unfolding of neutron energy spectra with fisher regularisation, IEEE Transactions on Nuclear Science, 57 (2010), 768–774.

[10] M. P. Friedlander and P. Tseng, Exact regularization of convex programs, SIAM Journal on Optimization, 18 (2008), 1326–1350.

[11] G. H. Golub and C. F. Van Loan, Matrix Computations, 3rd edition, The Johns Hopkins University Press, Baltimore, 1996.

[12] C. C. Gonzaga and R. A. Tapia, On the convergence of the Mizuno-Todd-Ye algorithm to the analytic center of the solution set, SIAM Journal on Optimization, 7 (1997), 47–65.

[13] M. D. González-Lima, R. A. Tapia and F. A. Potra, On effectively computing the analytic center of the solution set by primal-dual interior-point methods, SIAM Journal on Optimization, 8 (1998), 1–25.

[14] S. Itoh and T. Tsunoda, Neutron spectra unfolding with maximum entropy and maximum likelihood, Journal of Nuclear Science and Technology, 26 (1989), 833–843.

[15] M. Kojima, N. Megiddo and S. Mizuno, A primal—dual infeasible-interior-point algorithm for linear programming, Mathematical Programming, 61 (1993), 263–280.

[16] M. Kojima, S. Mizuno and A. Yoshise, A primal-dual interior point algorithm for linear programming, in Progress in Mathematical Programming: Interior-Point and Related Methods (ed. N. Megiddo), Springer, New York, (1989), 29–47.

[17] I. J. Lustig, R. E. Marsten and D. F. Shanno, On implementing Mehrotra’s predictor-corrector interior-point method for linear programming, SIAM Journal on Optimization, 2 (1992), 435–449.

[18] M. Matzke, Propagation of uncertainties in unfolding procedures, Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment, 476 (2002), 230–241.
[19] S. Mehrotra, On the implementation of a primal-dual interior point method, *SIAM Journal on Optimization*, 2 (1992), 575–601.

[20] S. Mehrotra, Quadratic convergence in a primal-dual method, *Mathematics of Operations Research*, 18 (1993), 741–751.

[21] S. Mizuno, Polynomiality of infeasible-interior-point algorithms for linear programming, *Mathematical Programming*, 67 (1994), 109–119.

[22] S. Mizuno, M. J. Todd and Y. Ye, On adaptive-step primal-dual interior-point algorithms for linear programming, *Mathematics of Operations Research*, 18 (1993), 964–981.

[23] B. Mukherjee, A high-resolution neutron spectra unfolding method using the genetic algorithm technique, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476 (2002), 247–251.

[24] M. Reginatto, P. Goldhagen and S. Neumann, Spectrum unfolding, sensitivity analysis and propagation of uncertainties with the maximum entropy deconvolution code maxed, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 476 (2002), 242–246.

[25] C. Roos, T. Terlaky and J.-P. Vial, *Interior Point Methods for Linear Optimization*, 2nd edition, Springer, Berlin, 2005.

[26] V. Suman and P. Sarkar, Neutron spectrum unfolding using genetic algorithm in a Monte Carlo simulation, *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 280 (1989), 103–112.

[27] M. Wright, Ill-conditioning and computational error in interior methods for nonlinear programming, *SIAM Journal on Optimization*, 9 (1998), 84–111.

[28] S. Wright, Effects of finite-precision arithmetic on interior-point methods for nonlinear ill-posed inverse problems, *Inverse Problems*, 21 (2005), 821–838.

[29] Y. Zhang, On the convergence of a class of infeasible interior-point methods for the horizontal linear complementarity problem, *SIAM Journal on Optimization*, 4 (1994), 208–227.

[30] Y. Zhang, Solving large-scale linear programs by interior-point methods under the matlab environment, *Optimization Methods and Software*, 10 (1998), 1–31.
[41] Y. Zhangsun, *Unfolding Method Based on Entropy Theory for the Determination of Neutron Spectrum (in Chinese)*, Master's thesis, Northwest Institute of Nuclear Technology, Xi'an, Shanxi, P. R. China, 2015.

Received February 2018; revised August 2019.

E-mail address: dyb@lsec.cc.ac.cn
E-mail address: wangzhh@bjtu.edu.cn
E-mail address: fengminxu@mail.xjtu.edu.cn