A polynomial predictor-corrector trust-region algorithm for linear programming

Guanghui Lan* Renato D.C. Monteiro† Takashi Tsuchiya‡

May 31, 2007 (revised: August 8, 2008)

Abstract

In this paper we present a scaling-invariant interior-point predictor-corrector type algorithm for linear programming (LP) whose iteration-complexity is polynomially bounded by the dimension and the logarithm of a certain condition number of the LP constraint matrix. At the predictor stage, the algorithm either takes the step along the standard affine scaling (AS) direction or a new trust-region type direction, whose construction depends on a scaling-invariant bipartition of the variables determined by the AS direction. This contrasts with the layered least squares direction introduced in [29], whose construction depends on multiple-layered partitions of the variables that are not scaling-invariant. Moreover, it is shown that the overall arithmetic complexity of the algorithm (weakly) depends on the right hand side and the cost of the LP in view of the work involved in the computation of the trust region steps.

Keywords: Interior-point algorithms, primal-dual algorithms, path-following, trust-region, central path, layered steps, condition number, polynomial complexity, predictor-corrector, affine scaling, strongly polynomial, linear programming.

AMS 2000 subject classification: 65K05, 68Q25, 90C05, 90C51, 90C60

1 Introduction

We consider the linear programming (LP) problem

$$\begin{array}{ll} \text{minimize}_x & c^T x \\ \text{subject to} & Ax = b,\ x \geq 0 \end{array} \tag{1}$$

and its associated dual problem

$$\begin{array}{ll} \text{maximize}_{(y,s)} & b^T y \\ \text{subject to} & A^T y + s = c,\ s \geq 0 \end{array} \tag{2}$$

*School of ISyE, Georgia Institute of Technology, Atlanta, Georgia 30332, USA. This author was supported by NSF Grant CCR-0430644 and ONR Grants N00014-05-1-0183 and N00014-08-1-0033. (Email: glan@isye.gatech.edu).

†School of ISyE, Georgia Institute of Technology, Atlanta, Georgia 30332, USA. This author was supported in part by NSF Grants CCR-0430644 and CCF-0808863 and ONR Grant N00014-05-1-0183 and N00014-08-1-0033. (Email: monteiro@isye.gatech.edu).

‡The Institute of Statistical Mathematics, 4-6-7 Minami-Azabu, Minato-Ku, Tokyo, 106-8569, Japan. (Email: tsuchiya@sun312.ism.ac.jp).

where $A \in \Re^{m\times n}, c \in \Re^n$, and $b \in \Re^m$ are given, and the vectors $x, s \in \Re^n$ and $y \in \Re^m$ are the unknown variables.

Karmarkar in his seminal paper [5] proposed the first polynomially convergent interior-point method with an $\mathcal{O}(nL)$ iteration-complexity bound, where L is the size of the LP instance (1). The first path-following interior-point algorithm was proposed by Renegar in his breakthrough paper [18]. Renegar's method closely follows the primal central path and exhibits an $\mathcal{O}(\sqrt{n}L)$ iteration-complexity bound. The first path-following algorithm which simultaneously generates iterates in both the primal and dual spaces has been proposed by Kojima, Mizuno and Yoshise [6] and Tanabe [20], based on ideas suggested by Megiddo [8]. In contrast to Renegar's algorithm, Kojima et al.'s algorithm has an $\mathcal{O}(nL)$ iteration-complexity bound. A primal-dual path-following with an $\mathcal{O}(\sqrt{n}L)$ iteration-complexity bound was subsequently obtained by Kojima, Mizuno and Yoshise [7] and Monteiro and Adler [12, 13] independently. Following these developments, many other primal-dual interior-point algorithms for linear programming have been proposed.

An outstanding open problem in optimization is whether there exists a strongly polynomial algorithm for linear programming, that is one whose complexity is bounded by a polynomial of m and n only. A major effort in this direction is due to Tardos [21] who developed a polynomial-time algorithm whose complexity is bounded by a polynomial of m, n and L_A, where L_A denotes the size of A. Such an algorithm gives a strongly polynomial method for the important class of linear programming problems where the entries of A are either 1, -1 or 0, e.g., LP formulations of network flow problems. Tardos' algorithm consists of solving a sequence of "low-sized" LP problems by a standard polynomially-convergent LP method and using their solutions to obtain the solution of the original LP problem.

The development of a method which works entirely in the context of the original LP problem and whose complexity is also bounded by a polynomial of m, n and L_A is due to Vavasis and Ye [29]. Their method is a primal-dual path-following interior-point algorithm similar to the ones mentioned above except that it uses once in a while a crucial step, namely the least layered square (LLS) direction. They showed that their method has an $O(n^{3.5}(\log \bar{\chi}_A + \log n))$ iteration-complexity bound, where $\bar{\chi}_A$ is a condition number associated with A having the property that $\log \bar{\chi}_A = O(L_A)$. The number $\bar{\chi}_A$ was first introduced implicitly by Dikin [2] in the study of primal affine scaling (AS) algorithms, and was later studied by several researchers including Vanderbei and Lagarias [28], Todd [22] and Stewart [19]. Properties of $\bar{\chi}_A$ are studied in [4, 26, 27].

The complexity analysis of Vavasis and Ye's algorithm is based on the notion of crossover event, a combinatorial event concerning the central path. Intuitively, a crossover event occurs between two variables when one of them is larger than the other at a point in the central path and then becomes smaller asymptotically as the optimal solution set is approached. Vavasis and Ye showed that there can be at most $n(n-1)/2$ crossover events and that a distinct crossover event occurs every $O(n^{1.5}(\log \bar{\chi}_A + \log n))$ iterations, from which they deduced the overall $O(n^{3.5}(\log \bar{\chi}_A + \log n))$ iteration-complexity bound. In [11], an LP instance is given where the number of crossover events is $\Theta(n^2)$.

One difficulty of Vavasis and Ye's method is that it requires the explicit knowledge of $\bar{\chi}_A$ in order to determine a partition of the variables into layers used in the computation of the LLS step. This difficulty was remedied in a variant proposed by Megiddo, Mizuno and Tsuchiya [9] which does not require the explicit knowledge of the number $\bar{\chi}_A$. They observed that at most n types of partitions arise as $\bar{\chi}_A$ varies from 1 to ∞, and that one of these can be used to compute the LLS step. Based on this idea, they developed a variant which computes the LLS steps for all these partitions and picks

the one that yields the greatest duality gap reduction at the current iteration. Another approach that also remedies the above difficulty was proposed by Monteiro and Tsuchiya [15]. Their algorithm computes only one LLS step per iteration without any explicit knowledge of $\bar{\chi}_A$. This method is a predictor-corrector type algorithm like the one described in [10] except that at the predictor stage it takes a step along either the primal-dual AS step or the LLS step. In contrast to the LLS step used in Vavasis and Ye's algorithm, the partition of variables used for computing the LLS step is constructed from the information provided by the AS direction and hence does not require any knowledge on $\bar{\chi}_A$. Both of these variants ([9], [15]) have exactly the same overall complexity as Vavasis and Ye's algorithm.

Another disadvantage associated with Vavasis and Ye's algorithm, as well as its variants in [9] and [15], is that they are not scaling-invariant under the change of variables $(x, y, s) = (D\tilde{x}, \tilde{y}, D^{-1}\tilde{s})$, where D is a positive diagonal matrix. Hence, when these algorithms are applied to the scaled pair of LP problems, the number of iterations performed by it generally changes and is now bounded by $O(n^{3.5}\log(\bar{\chi}_{AD}+n))$, as AD is the coefficient matrix for the scaled pair of LP problems. On the other hand, using the notion of crossover events, LLS steps and a few other nontrivial ideas, Monteiro and Tsuchiya [16] have shown that, for the Mizuno-Todd-Ye predictor-corrector (MTY P-C) algorithm, the number of iterations needed to approximately traverse the central path from μ_0 to μ_f is bounded by $\mathcal{O}(n^{3.5}\log(\bar{\chi}_A^* + n) + T(\mu_0/\mu_f))$, where $\bar{\chi}_A^*$ is the infimum of $\bar{\chi}_{AD}$ as D varies over the set of positive diagonal matrices and $T(t) \equiv \min\{n^2\log(\log t), \log t\}$ for all $t > 0$. The condition number $\bar{\chi}_A^*$ is clearly scaling-invariant and the ratio $\bar{\chi}_A^*/\bar{\chi}_A$, as a function of A, can be arbitrarily small (see [16]). Hence, while the iteration-complexity obtained in [16] for the MTY P-C algorithm has the extra term $T(\mu_0/\mu_f)$, its first term can be considerably smaller than the bound obtained by Vavasis and Ye. Also note that, as μ_0/μ_f grows to ∞, the iteration-complexity bound obtained in [16] is smaller than the classical iteration-complexity bound of $\mathcal{O}(\sqrt{n}\log(\mu_0/\mu_f))$ established in [10] for the MTY P-C algorithm.

An interesting open problem is whether one can develop a scaling-invariant interior-point algorithm for linear programming whose iteration-complexity and arithmetic-complexity are bounded by a polynomial of n and $\log\bar{\chi}_A^*$. In this paper, we partially answer the above question by presenting a predictor-corrector type algorithm, referred to as the predictor-corrector trust-region (PC-TR) algorithm, which has $\mathcal{O}(n^{3.5}\log(\bar{\chi}_A^* + n))$ iteration-complexity bound. It is a predictor-corrector algorithm similar to the one developed in [10] except that, at the predictor stage, it takes a step along either the AS direction or a trust-region (TR) type step. Unlike the LLS direction used in the predictor-corrector algorithm of [15], the TR direction depends on a scaling-invariant bipartition of the variables and hence it is a scaling-invariant direction. Its iteration can be briefly described as follows. First, the AS direction is computed and a test involving this direction is performed to determine whether the TR step is needed. If the TR direction is not needed, a step along the AS direction, followed by a standard corrector step, is taken as usual. Otherwise, the AS direction determines a scaling-invariant bipartition of the variables which allows to construct a pair of primal and dual trust region subproblems whose optimal solutions yield the TR direction. Then the algorithm takes a step along either the AS or the TR direction whichever yields the largest duality gap reduction. Moreover, we show that the overall arithmetic complexity of the PC-TR algorithm (weakly) depends also on b and c due to work involved in the computation of the trust region steps.

The organization of the paper is as follows. Section 2 consists of six subsections. In Subsection 2.1, we review the notion of the primal-dual central path and its associated two norm neighborhoods. Subsection 2.2 introduces the notion of the condition number $\bar{\chi}_A$ of a matrix A and describes the

properties of $\bar{\chi}_A$ that will be useful in our analysis. Subsection 2.3 reviews the AS step and the corrector (or centrality) step which are the basic ingredients of several well-known interior-point algorithms. Subsection 2.4 motivates and formally introduces the TR step. Subsection 2.5 describes an interior-point algorithm based on these TR steps, which we refer to as the predictor-corrector trust-region (PC-TR) algorithm, and states one of main results of this paper which gives an upper bound on the iteration-complexity of the PC-TR algorithm. Subsection 2.6 introduces a variant of the PC-TR algorithm with the same iteration-complexity as the latter one and discusses a procedure for computing the TR steps used by this variant. It also states the other main result of this paper regarding the overall arithmetic complexity of the above variant of the PC-TR algorithm. Section 3, which consists of three subsections, introduces some basic tools which are used in our convergence analysis. Subsection 3.1 discusses the notion of crossover events. Subsection 3.2 introduces the LLS direction and states an approximation result that provides an estimation of the closeness between the AS direction and the LLS direction. Subsection 3.3 reviews an important result, which basically provides sufficient conditions for the occurrence of crossover events. Section 4 is dedicated to the proof of the main result stated in Subsection 2.5. Section 5 provides the proof of the other main result stated in Subsection 2.6 regarding the arithmetic complexity of the variant of the PC-TR algorithm. Finally, the Appendix gives the proof of an important lemma used in Subsection 2.4 to motivate the definition of the TR step.

The following notation is used throughout our paper. We denote the vector of all ones by e. Its dimension is always clear from the context. The symbols $\Re^n$, $\Re^n_+$ and $\Re^n_{++}$ denote the n-dimensional Euclidean space, the nonnegative orthant of $\Re^n$ and the positive orthant of $\Re^n$, respectively. The set of all $m \times n$ matrices with real entries is denoted by $\Re^{m\times n}$. If J is a finite index set then $|J|$ denotes its cardinality, that is the number of elements of J. For $J \subseteq \{1,\ldots,n\}$ and $w \in \Re^n$, we let w_J denote the subvector $[w_i]_{i\in J}$; moreover, if E is an $m \times n$ matrix then E_J denotes the $m \times |J|$ submatrix of E corresponding to J. For a vector $w \in \Re^n$, we let: $\max(w)$ and $\min(w)$ denote the largest and the smallest component of w, respectively; $\mathrm{Diag}(w)$ denote the diagonal matrix whose i-th diagonal element is w_i for $i = 1,\ldots,n$, and; for an arbitrary $\alpha \in \Re$, w^α denote the vector $[\mathrm{Diag}(w)]^\alpha e$ whenever it is well-defined. For two vectors $u, v \in \Re^n$, uv denotes their Hadamard product, i.e. the vector in $\Re^n$ whose ith component is $u_i v_i$. The Euclidean norm, the 1-norm and the ∞-norm are denoted by $\|\cdot\|$, $\|\cdot\|_1$ and $\|\cdot\|_\infty$, respectively. For a matrix E, $\mathrm{Im}(E)$ denotes the subspace generated by the columns of E and $\mathrm{Ker}(E)$ denotes the subspace orthogonal to the rows of E. The superscript ^{T} denotes transpose.

2 The problem and Algorithm

In this section we propose a predictor-corrector primal-dual interior-point algorithm with trust-region steps for solving linear programming (1) and (2). We also present the main convergence results for the algorithm. One result establishes a polynomial iteration-complexity bound, namely, $O(n^{3.5}\log(\bar{\chi}_A^*+n+\varepsilon_0^{-1}))$, where ε_0 is a constant and $\bar{\chi}_A^*$ is a certain scaling-invariant condition number associated with the constraint matrix A, and the other result establishes a polynomial arithmetic complexity bound for the algorithm.

2.1 The central path

In this subsection we introduce the pair of primal and dual linear programs and the assumptions used in our development. We also describe the associated primal-dual central path and its corresponding two-norm neighborhoods.

Given $A \in \Re^{m\times n}$, $c \in \Re^n$ and $b \in \Re^m$, consider the pairs of linear programs (1) and (2), where $x \in \Re^n$ and $(y, s) \in \Re^m \times \Re^n$ are their respective variables. The set of strictly feasible solutions for these problems are

$$\begin{aligned} \mathcal{P}^{++} &\equiv \{x \in \Re^n : Ax = b,\ x > 0\}, \\ \mathcal{D}^{++} &\equiv \{(y, s) \in \Re^{m\times n} : A^T y + s = c,\ s > 0\}, \end{aligned}$$

respectively. Throughout the paper we make the following assumptions on the pair of problems (1) and (2).

A.1 $\mathcal{P}^{++}$ and $\mathcal{D}^{++}$ are nonempty.

A.2 The rows of A are linearly independent.

Under the above assumptions, it is well-known that for any $\nu > 0$ the system

$$\begin{aligned} xs &= \nu e, && (3)\\ Ax &= b, \quad x > 0, && (4)\\ A^T y + s &= c, \quad s > 0, && (5) \end{aligned}$$

has a unique solution (x, y, s), which we denote by $(x(\nu), y(\nu), s(\nu))$. The central path is the set consisting of all these solutions as ν varies in $(0, \infty)$. As ν converges to zero, the path $(x(\nu), y(\nu), s(\nu))$ converges to a primal-dual optimal solution (x^*, y^*, s^*) for problems (1) and (2). Given a point $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$, its duality gap and its normalized duality gap are defined as $x^T s$ and $\mu = \mu(x, s) \equiv x^T s/n$, respectively, and the point $(x(\mu), y(\mu), s(\mu))$ is said to be the central point associated with w. Note that $(x(\mu), y(\mu), s(\mu))$ also has normalized duality gap μ. We define the proximity measure of a point $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ with respect to the central path by

$$\eta(w) \equiv \|xs/\mu - e\|.$$

Clearly, $\eta(w) = 0$ if and only if $w = (x(\mu), y(\mu), s(\mu))$, or equivalently w coincides with its associated central point. The two-norm neighborhood of the central path with opening $\beta > 0$ is defined as

$$\mathcal{N}(\beta) \equiv \{w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++} : \eta(w) \le \beta\}.$$

Finally, for any point $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$, we define

$$\delta(w) \equiv s^{1/2} x^{-1/2} \in \Re^n. \tag{6}$$

The following propositions provide important estimates which are used throughout our analysis.

Proposition 2.1 *For every $0 < \nu_1 \le \nu_2$, we have*

$$s(\nu_1) \le n s(\nu_2) \quad \text{and} \quad x(\nu_1) \le n x(\nu_2). \tag{7}$$

Proof. Please refer to Lemma 16 of Vavasis and Ye [29]. ∎

Proposition 2.2 *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1)$ be given and define $\delta \equiv \delta(w)$. Let $w(\mu) = (x(\mu), y(\mu), s(\mu))$ be the central point associated with w. Then:*

$$\frac{1-\beta}{(1+\beta)^{1/2}}\,\delta \le \frac{s(\mu)}{\sqrt{\mu}} \le \frac{(1+\beta)^{1/2}}{1-\beta}\,\delta.$$

Proof. This result is summarized in Proposition 2.1 in [15]. ∎

2.2 Condition number

In this subsection we define a certain condition number associated with the constraint matrix A and state the properties of $\bar{\chi}_A$ which will play an important role in our analysis.

Let $\mathcal{D}$ denote the set of all positive definite $n \times n$ diagonal matrices and define

$$\begin{aligned} \bar{\chi}_A &\equiv \sup\left\{\|A^T(A\tilde{D}A^T)^{-1}A\tilde{D}\| \ : \tilde{D} \in \mathcal{D}\right\} \qquad (8)\\ &= \sup\left\{\frac{\|A^T y\|}{\|c\|} : \ y = \operatorname{argmin}_{\tilde{y}\in\Re^n}\|\tilde{D}^{1/2}(A^T\tilde{y} - c)\| \text{ for some } 0 \neq c \in \Re^n \text{ and } \tilde{D} \in \mathcal{D}\right\}. \end{aligned}$$

The parameter $\bar{\chi}_A$ plays a fundamental role in the complexity analysis of algorithms for linear programming and least square problems (see [29] and references therein). Its finiteness has been firstly established by Dikin [2]. Other authors have also given alternative derivations of the finiteness of $\bar{\chi}_A$ (see for example Stewart [19], Todd [22] and Vanderbei and Lagarias [28]).

We summarize in the next proposition a few important facts about the parameter $\bar{\chi}_A$.

Proposition 2.3 *Let $A \in \Re^{m\times n}$ with full row rank be given. Then, the following statements hold:*

a) $\bar{\chi}_{GA} = \bar{\chi}_A$ for any nonsingular matrix $G \in \Re^{m\times m}$;

b) $\bar{\chi}_A = \max\{\|G^{-1}A\| : G \in \mathcal{G}\}$ where $\mathcal{G}$ denote the set of all $m \times m$ nonsingular submatrices of A;

c) if the $m \times m$ identity matrix is a submatrix of A and $\tilde{A}$ is an $r \times n$ submatrix of A then $\|\tilde{G}^{-1}\tilde{A}\| \le \bar{\chi}_A$ for every $r \times r$ nonsingular submatrix $\tilde{G}$ of $\tilde{A}$.

Proof. Statement a) readily follows from the definition (8). The inequality $\bar{\chi}_A \ge \max\{\|G^{-1}A\| : G \in \mathcal{G}\}$ is established in Lemma 3 of [29] while the proof of the reverse inequality is given in [22] (see also Theorem 1 of [23]). Hence, b) holds. A proof of c) can be found in [15]. ∎

The condition number $\bar{\chi}_A^*$, defined by taking the infimum of the condition number $\bar{\chi}_{AD}$ as D varies over the set of positive diagonal matrices, that is, $\bar{\chi}_A^* \equiv \inf\{\bar{\chi}_{AD} : D \in \mathcal{D}\}$, also plays an important role in the convergence analysis for our algorithm. Note that by definition, $\bar{\chi}_A^*$ is a scaling-invariant quantity.

2.3 Predictor-corrector step

In this subsection we describe the well-known predictor-corrector (P-C) iteration which is used by several interior-point algorithms (see for example Mizuno et al. [10]). We also describe the properties of this iteration which will be used in our analysis.

The P-C iteration consists of two steps, namely the predictor (or AS) step and the corrector (or centrality) step. The search direction used by either step from a current point in $(x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ is the solution of the following linear system of equations

$$\begin{aligned} S\Delta x + X\Delta s &= \sigma\mu e - xs, \\ A\Delta x &= 0, \\ A^T\Delta y + \Delta s &= 0, \end{aligned} \tag{9}$$

where $\mu = \mu(x, s)$ and $\sigma \in \Re$ is a pre-specified parameter, commonly referred to as the centrality parameter. When $\sigma = 0$, we denote the solution of (9) by $(\Delta x^{\rm a}, \Delta y^{\rm a}, \Delta s^{\rm a})$ and refer to it as the primal-dual affine scaling direction at w; it is the direction used in the predictor step of the P-C iteration. When $\sigma = 1$, we denote the solution of (9) by $(\Delta x^{\rm c}, \Delta y^{\rm c}, \Delta s^{\rm c})$ and refer to it as the corrector direction at w; it is the direction used in the corrector step of the P-C iteration.

We are now ready to describe the entire predictor-corrector iteration. Suppose that a constant $\beta \in (0, 1/4]$ and a point $w = (x, y, s) \in \mathcal{N}(\beta)$ is given. The P-C iteration generates another point $(x^+, y^+, s^+) \in \mathcal{N}(\beta)$ as follows. It first moves along the direction $(\Delta x^{\rm a}, \Delta y^{\rm a}, \Delta s^{\rm a})$ until it hits the boundary of the enlarged neighborhood $\mathcal{N}(2\beta)$. More specifically, it computes the point $w^{\rm a} = (x^{\rm a}, y^{\rm a}, s^{\rm a}) \equiv (x, y, s) + \alpha_{\rm a}(\Delta x^{\rm a}, \Delta y^{\rm a}, \Delta s^{\rm a})$ where

$$\alpha_{\rm a} \equiv \sup\{\alpha \in [0, 1] : (x, y, s) + \alpha(\Delta x^{\rm a}, \Delta y^{\rm a}, \Delta s^{\rm a}) \in \mathcal{N}(2\beta)\}. \tag{10}$$

Next, the P-C iteration generates a point inside the smaller neighborhood $\mathcal{N}(\beta)$ by taking a unit step along the corrector direction $(\Delta x^{\rm c}, \Delta y^{\rm c}, \Delta s^{\rm c})$ at the point $w^{\rm a}$, that is, it computes the point $(x^+, y^+, s^+) \equiv (x^{\rm a}, y^{\rm a}, s^{\rm a}) + (\Delta x^{\rm c}, \Delta y^{\rm c}, \Delta s^{\rm c}) \in \mathcal{N}(\beta)$. The successive repetition of this iteration leads to the so called Mizuno-Todd-Ye (MTY) predictor-corrector algorithm (see [10]).

Our method is very similar to the algorithm of [10] except that it sometimes replaces the AS step by the trust-region step described in the next subsection. The insertion of the trust region step in the above MTY predictor-corrector algorithm guarantees that the modified method has the finite termination property. The trust-region step is only taken when it yields a point with a smaller duality gap than the one obtained from the AS step as described above.

In the remaining part of this subsection, we discuss some properties of the P-C iteration and the primal-dual AS direction. For a proof of the next two propositions, we refer the reader to [10].

Proposition 2.4 *(***Predictor step***) Suppose that $w = (x, y, s) \in \mathcal{N}(\beta)$ for some constant $\beta \in (0, 1/2]$. Let $\Delta w^{\rm a} = (\Delta x^{\rm a}, \Delta y^{\rm a}, \Delta s^{\rm a})$ denote the affine scaling direction at $w^{\rm a}$ and let $\alpha_{\rm a}$ be the step-size computed according to (10). Then the following statements hold:*

a) the point $w + \alpha\Delta w^{\rm a}$ has normalized duality gap $\mu(\alpha) = (1 - \alpha)\mu$ for all $\alpha \in \Re$;

b) $\alpha_{\rm a} \geq \sqrt{\beta/n}$ and hence $\mu(\alpha_{\rm a})/\mu \leq 1 - \sqrt{\beta/n}$.

Proposition 2.5 *(***Corrector step***) Suppose that $w = (x, y, s) \in \mathcal{N}(2\beta)$ for some constant $\beta \in (0, 1/4]$ and let $(\Delta x^{\rm c}, \Delta y^{\rm c}, \Delta s^{\rm c})$ denote the corrector step at w. Then, $w + \Delta w^{\rm c} \in \mathcal{N}(\beta)$. Moreover, the (normalized) duality gap of $w + \Delta w^{\rm c}$ is the same as that of w.*

For the sake of future usage, we mention the following alternative characterization of the primal-dual AS direction whose verification is straightforward:

$$\Delta x^{\mathrm{a}} \equiv \operatorname{argmin}_{p\in\Re^n}\left\{\|\delta(x+p)\|^2 : Ap=0\right\}, \tag{11}$$

$$(\Delta y^{\mathrm{a}},\Delta s^{\mathrm{a}}) \equiv \operatorname{argmin}_{(r,q)\in\Re^m\times\Re^n}\left\{\|\delta^{-1}(s+q)\|^2 : A^Tr+q=0\right\}, \tag{12}$$

where $\delta\equiv\delta(w)$. For a search direction $(\Delta x,\Delta y,\Delta s)$ at a point (x,y,s), the quantity

$$(Rx(w),Rs(w)) \equiv \left(\tfrac{\delta(x+\Delta x)}{\sqrt{\mu}}\,,\,\tfrac{\delta^{-1}(s+\Delta s)}{\sqrt{\mu}}\right) = \left(\tfrac{x^{1/2}s^{1/2}+\delta\Delta x}{\sqrt{\mu}}\,,\,\tfrac{x^{1/2}s^{1/2}+\delta^{-1}\Delta s}{\sqrt{\mu}}\right) \tag{13}$$

appears quite often in our analysis. We refer to it as the *residual* of $(\Delta x,\Delta y,\Delta s)$. Note that if $(Rx^{\mathrm{a}}(w),Rs^{\mathrm{a}}(w))$ is the residual of $(\Delta x^{\mathrm{a}},\Delta y^{\mathrm{a}},\Delta s^{\mathrm{a}})$ then

$$Rx^{\mathrm{a}}(w)=-\frac{1}{\sqrt{\mu}}\delta^{-1}\Delta s^{\mathrm{a}},\quad Rs^{\mathrm{a}}(w)=-\frac{1}{\sqrt{\mu}}\delta\Delta x^{\mathrm{a}}, \tag{14}$$

and

$$Rx^{\mathrm{a}}(w)+Rs^{\mathrm{a}}(w)=\frac{x^{1/2}s^{1/2}}{\sqrt{\mu}}, \tag{15}$$

due to the fact that $(\Delta x^{\mathrm{a}},\Delta y^{\mathrm{a}},\Delta s^{\mathrm{a}})$ satisfies the first equation in (9) with $\sigma=0$. The following quantity is used in the test to determine when the trust-region step should be used in place of the AS step:

$$\varepsilon^{\mathrm{a}}_{\infty}(w)\equiv\max_i\left\{\min\left\{|Rx^{\mathrm{a}}_i(w)|\,,\,|Rs^{\mathrm{a}}_i(w)|\right\}\right\}. \tag{16}$$

We end this section by providing some estimates involving the residual of the AS direction.

Lemma 2.6 *Suppose that $w=(x,y,s)\in\mathcal{N}(\beta)$ for some $\beta\in(0,1/4]$. Then, for all $i=1,\ldots,n$, we have*

$$\max\left\{|Rx^{\mathrm{a}}_i(w)|,|Rs^{\mathrm{a}}_i(w)|\right\}\geq\frac{\sqrt{1-\beta}}{2}\geq\frac{1}{4}.$$

Proof. Assume for a contradiction that for some $i\in\{1,\ldots,n\}$, $\max\{|Rx^{\mathrm{a}}_i(w)|,|Rs^{\mathrm{a}}_i(w)|\}<\sqrt{1-\beta}/2$. Then, using (15), we obtain the following contradiction:

$$\frac{x_i^{1/2}s_i^{1/2}}{\sqrt{\mu}}=Rx^{\mathrm{a}}_i(w)+Rs^{\mathrm{a}}_i(w)\leq|Rx^{\mathrm{a}}_i(w)|+|Rs^{\mathrm{a}}_i(w)|<\sqrt{1-\beta}\leq\frac{x_i^{1/2}s_i^{1/2}}{\sqrt{\mu}}.$$

■

2.4 Trust region step

In this subsection we introduce a new type of search step, namely, the trust-region (TR) step, and describe some properties about it.

The definition of the TR step is motivated by the following result regarding the duality gap reduction obtained by moving along a search direction satisfying certain conditions. This result can be viewed as a generalization of Lemma 4.6 in [15] and its proof is given in the Appendix.

Lemma 2.7 *Let $w \in \mathcal{N}(\beta)$ with $\beta \in (0, 1/2]$ and a direction $\Delta w = (\Delta x, \Delta y, \Delta s)$ satisfying $A\Delta x = 0$ and $A^T\Delta y + \Delta s = 0$ be given. Then, for any positive scalar γ satisfying*

$$\left(4\sqrt{2} + \sqrt{2(1+\beta)}\right)\gamma \leq \frac{\beta - 2\beta^2}{1+2\beta} \tag{17}$$

and for any bipartition (B, N) of $\{1, 2, ..., n\}$, the condition

$$\max\left\{\frac{\|\delta_B \Delta x_B\|}{\sqrt{\mu}}, \frac{\|\delta_N^{-1}\Delta s_N\|}{\sqrt{\mu}}\right\} \leq \gamma \tag{18}$$

implies that

$$\frac{\mu(w + \alpha_\tau \Delta w)}{\mu(w)} \leq \frac{\sqrt{1+\beta} + \gamma}{2\gamma}\max\{\|Rx_N\|, \|Rs_B\|\}, \tag{19}$$

where $\alpha_\tau \equiv \sup\{\alpha \in [0,1] : w + \alpha\Delta w \in \mathcal{N}(2\beta)\}$ and $(Rx(w), Rs(w))$ is defined in (13).

A trivial application of Lemma 2.7 is as follows. Let (B, N) be the AS-bipartition at w, i.e.

$$B = B(w) \equiv \{i : |Rs_i^a(w)| \leq |Rx_i^a(w)|\}, \quad N = N(w) \equiv \{i : |Rs_i^a(w)| > |Rx_i^a(w)|\}. \tag{20}$$

and let

$$\varepsilon_2^{\mathrm{a}}(w) := \max\{\|Rx_N^a\|, \|Rs_B^a\|\}. \tag{21}$$

Then, in view of Lemma 2.7 and identity (14), the condition $\varepsilon_2^{\mathrm{a}}(w) \leq \gamma$ implies that

$$\frac{\mu(w + \alpha\Delta w^{\mathrm{a}})}{\mu(w)} \leq \frac{\sqrt{1+\beta} + \gamma}{2\gamma}\varepsilon_2^{\mathrm{a}}(w),$$

showing that, the smaller the quantity $\varepsilon_2^{\mathrm{a}}(w)$ is, the larger the reduction of the duality gap will be, as move from w along the AS direction $\Delta w^{\mathrm{a}}(w)$.

However, a more interesting application of Lemma 2.7 is towards deriving a new scaling-invariant search direction, which we refer to as the trust-region direction (see the definition below). This direction is the one which minimizes the right hand side of (19) subject to the condition (18) when $(B, N) = (B(w), N(w))$. To define this direction, set $(B, N) = (B(w), N(w))$ and consider the two subproblems:

$$\begin{array}{ll} \text{minimize} & \|\delta_N(x_N + \Delta x_N)\| \\ \text{subject to} & \|\delta_B \Delta x_B\|/\sqrt{\mu} \leq \gamma_p \\ & A\Delta x = 0 \end{array} \tag{22}$$

and

$$\begin{array}{ll} \text{minimize} & \|\delta_B^{-1}(s_B + \Delta s_B)\| \\ \text{subject to} & \|\delta_N^{-1}\Delta s_N\|/\sqrt{\mu} \leq \gamma_d \\ & A^T\Delta y + \Delta s = 0 \end{array} . \tag{23}$$

Definition: Given $w \in \mathcal{N}(\beta)$ and positive scalars γ_p and γ_d, let $\Delta x^\tau(w; \gamma_p)$ and $(\Delta y^\tau(w; \gamma_d), \Delta s^\tau(w; \gamma_d))$ denote optimal solutions of subproblems (22) and (23), respectively. The direction $\Delta w^\tau(w; \gamma_p, \gamma_d) \equiv (\Delta x^\tau(w; \gamma_p), \Delta y^\tau(w; \gamma_d), \Delta s^\tau(w; \gamma_d))$ is then referred to as a trust-region direction at w with radius pair (γ_p, γ_d).

We now make a few observations regarding the above definition. First, it can be easily shown that both subproblems (22) and (23) must have optimal solutions although their optimal solutions

are not necessarily unique. We will refer to any pair of optimal solutions of subproblems (22) and (23) as a trust-region step corresponding to the triple $(w;\gamma_p,\gamma_d)$. Second, if $\varepsilon_2^{\rm a}(w) \le \min\{\gamma_p,\gamma_d\}$, then the quantity $\varepsilon_2^{\tau}(w;\gamma_p,\gamma_d)$ defined as

$$\varepsilon_2^{\tau}(w;\gamma_p,\gamma_d) := \max\left\{\|Rx_N^{\tau}(w)\|, \|Rs_B^{\tau}(w)\|\right\}, \tag{24}$$

where $(Rx^{\tau}(w), Rs^{\tau}(w))$ denotes the residual pair for the the TR direction $\Delta w^{\tau}(w;\gamma_p,\gamma_d)$, satisfies $\varepsilon_2^{\tau}(w;\gamma_p,\gamma_p) \le \varepsilon_2^{\rm a}(w)$. In other words, whenever the AS direction is a reasonably good direction in the sense that $\varepsilon_2^{\rm a}(w)$ is sufficiently small, then the TR step is likely to be an even better direction in that it makes the right hand side of (19) smaller. Third, even though our definition of a TR step does not uniquely characterize it, one can easily modify the definition to make it uniquely defined in the following way. Without loss of generality, we consider only the primal direction, which previously was defined as an optimal solution of (22). This clearly implies that Δx_N^{τ} is uniquely defined. Now, minimizing the quantity $\|\delta_B \Delta x_B\|$ under the condition that $A\Delta x = 0$ and $\Delta x_N = \Delta x_N^{\tau}$ uniquely determines the component Δx_B, and hence the whole primal TR step. We note however that our analysis does not require that the TR step be uniquely determined and in fact works for any pair of optimal solutions of (22) and (23).

2.5 Main Algorithm and the convergence results

In this subsection, we describe our algorithm, namely, the predictor-corrector trust-region (PC-TR) algorithm, to solve the linear programming problem (1) and (2), and then state the main result of this paper which guarantees the convergence of the method in a strong sense. More specifically, this result states that the outer iteration-complexity bound for our method depends only on n and the scaling-invariant condition number $\bar{\chi}_A^*$.

We start by stating our predictor-corrector trust-region algorithm.

PC-TR Algorithm:

Let $0 < \beta \le 1/4$ and $\gamma > 0$ satisfying (17), $w^0 \in \mathcal{N}(\beta)$ and a scalar $\varepsilon_0 \in (0, \gamma/3]$ be given. Set $\mu_0 \equiv \mu(w^0)$ and $k = 0$.

1) Set $w = w^k$, compute the AS step $\Delta w^{\rm a}$ at w and the residual $\varepsilon_2^{\rm a}(w)$ as defined in (21);
2) If $\varepsilon_2^{\rm a}(w) > \varepsilon_0$ then set $w \leftarrow w + \alpha_a \Delta w^{\rm a}$, where α_a is defined as in (10) and go to 6);
3) Otherwise, compute the TR step $\Delta w^{\tau} = \Delta w^{\tau}(w;\gamma_p,\gamma_d)$, for scalars $\gamma_p, \gamma_d \in [\gamma/2, 2\gamma]$;
4) Let $w^{\tau} = w + \alpha_{\tau}\, \Delta w^{\tau}$, where $\alpha_{\tau} \equiv \sup\{\alpha \in [0,1] : w + \alpha \Delta w^{\tau} \in \mathcal{N}(2\beta)\}$;
5) If $\mu(w^{\tau}) < (1-\alpha_{\rm a})\mu$ then set $w \leftarrow w^{\tau}$, else set $w \leftarrow w + \alpha_{\rm a}\Delta w^{\rm a}$;
6) If $\mu(w) = 0$ then **stop**;
7) Compute the corrector step $\Delta w^{\rm c}$ at w and set $w \leftarrow w + \Delta w^{\rm c}$;
8) Set $w^{k+1} = w$, increment k by 1 and go to 1).

End

We now make a few comments about the above algorithm. In the main body of the algorithm, step 2 followed by step 7 is a standard predictor-corrector iteration of the type described in Subsection 2.3. This iteration is always performed in those iterations for which $\varepsilon_2^{\rm a}(w) > \varepsilon_0(w)$. In order to save computation time, the TR step is computed only in those iterations for which the current iterate

w satisfies $\varepsilon_2^{\rm a}(w) \leq \varepsilon_0$. In these iterations, the algorithm performs either a standard predictor-corrector iteration or a TR-corrector iteration depending on which of the two iterations gives the lower reduction of the duality gap. This test is performed in step 5 since the term $(1-\alpha_{\rm a})\mu$ is the normalized duality gap obtained when the AS step is taken (see Proposition 2.4(a)).

For the sake of future reference, we note that (17) and the assumption that $\beta \in (0, 1/4]$ imply that

$$\gamma \leq \frac{1}{20}, \quad \varepsilon_0 \leq \frac{\gamma}{3} \leq \frac{1}{60}. \tag{25}$$

We refer to an iteration where the TR step is computed as a *TR-iteration*. The following result is immediate from Lemma 2.7 and the definition of a TR-iteration.

Proposition 2.8 *Let w be an iterate of the PC-TR algorithm and assume that next iterate w^+ after w is obtained by means of a TR-iteration. Then,*

$$\frac{\mu(w^+)}{\mu(w)} \leq \frac{\sqrt{1+\beta}+\gamma}{2\gamma}\varepsilon_2^{\tau}(w;\gamma_p,\gamma_d) \leq \frac{\sqrt{1+\beta}+\gamma}{2\gamma}\varepsilon_2^{\rm a}(w) \leq \frac{\sqrt{1+\beta}+\gamma}{2\gamma}\varepsilon_0 \leq \frac{1}{4}, \tag{26}$$

Proof. First note that if the iteration from w is a TR-iteration then we have $\varepsilon_2^{\rm a}(w) \leq \varepsilon_0 \leq \gamma/3 \leq \min\{\gamma_p, \gamma_d\}$. The first three inequalities in (26) follow from Lemma 2.7, the previous observation and the second observation after (23). Moreover, the last inequality in (26) follows from (25) and the fact that $\beta \leq 1/4$. ∎

We have the following convergence result for the above algorithm.

Theorem 2.9 *The PC-TR algorithm described above finds a primal-dual optimal solution $w = (x^*, s^*, y^*)$ of (1) and (2) in at most $\mathcal{O}(n^{3.5}\log(\bar{\chi}_A^* + n + \varepsilon_0^{-1}))$ iterations, of which $\mathcal{O}(n^3 \log(\bar{\chi}_A^* + n + \varepsilon_0^{-1})/\log \varepsilon_0^{-1})$ are TR-iterations. In particular, if $\varepsilon_0^{-1} = \mathcal{O}((n+\bar{\chi}_A^*)^\kappa)$ for some constant $\kappa > 0$, then the total number of iterations is bounded by $\mathcal{O}(n^{3.5}\log(\bar{\chi}_A^* + n))$. Also, if $\varepsilon_0^{-1} = \Omega((n+\bar{\chi}_A^*)^\kappa)$ for some constant $\kappa > 0$, then the number of TR-iterations is bounded by $\mathcal{O}(n^3)$.*

Note that the PC-TR algorithm is scaling-invariant; i.e., if the change of variables $(x, y, s) = (D\tilde{x}, \tilde{y}, D^{-1}\tilde{s})$ for some $D \in \mathcal{D}$ is performed on the pair of problems (1) and (2) and the PC-TR algorithm is applied to the new dual pair of scaled problems, then the sequence of iterates $\tilde{w}^k$ generated satisfies $(x^k, y^k, s^k) = (D\tilde{x}^k, \tilde{y}^k, D^{-1}\tilde{s}^k)$ for all $k \geq 1$, as long as the initial iterate $\tilde{w}^0 \in \mathcal{N}(\beta)$ in the $\tilde{w}$-space satisfies $(x^0, y^0, s^0) = (D\tilde{x}^0, \tilde{y}^0, D^{-1}\tilde{s}^0)$. For this reason, the PC-TR algorithm should have an iteration-complexity bound which does not depend on the scaled space where the sequence of iterates is generated. Indeed, the iteration-complexity bound stated in Theorem 2.9 is scaling-invariant since the condition number $\bar{\chi}_A^*$ is too. It is worth noting that the PC-TR algorithm is also scaling-invariant with respect to a more strict notion of scaling invariance described in Tunçel [25], which corresponds to choosing the set $\mathcal{D}$ in the above defintion as the full automorphism group of R^n_+. Note that the latter set is larger than the set of positive diagonal maps since it contains the permutation maps, and hence it leads to a stronger notion of scaling invariance.

We note also that, to prove Theorem 2.9, it suffices to show that the the number of iterations of the PC-TR algorithm applied to (1) and (2) is bounded by $O(n^{3.5}\log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$. Indeed, since in the D-scaled space, the iterates can also be viewed as being generated by the PC-TR algorithm (started from a different point), then its complexity is also bounded by

$$O(n^{3.5}\log(\bar{\chi}_{AD} + n + \varepsilon_0^{-1})), \tag{27}$$

and hence by the infimum of (27) over all $D \in \mathcal{D}$, that is, by $O(n^{3.5}\log(\bar{\chi}_A^* + n + \varepsilon_0^{-1}))$.

Finally, Theorem 2.9 does not deal with the overall arithmetic complexity of the PC-TR algorithm. This issue will be dealt with in the next subsection and Section 5, where we discuss the arithmetic complexity involved in the computation of a TR step for a suitable variant of the PC-TR algorithm. Roughly speaking, we will derive a bound on the number of arithmetic operations required to compute a TR step which depends on the ratio between the current duality gap and the initial duality gap. This implies that the overall arithmetic complexity obtained in this paper for the above variant of the PC-TR algorithm depends (weakly) on b and c, though its number of iterations just depends on A as shown in Theorem 2.13.

2.6 Computing the TR step

In this subsection, we present an algorithm to compute the TR step and derive the arithmetic complexity for the PC-TR algorithm.

For the sake of simplicity, we focus our discussion on the computation of the primal TR direction. We start by introducing a search direction that is closely related to the optimal solution of (22). Given a scalar $\lambda > 0$ and $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$, consider the following direction defined as

$$\Delta x(\lambda) := \operatorname{argmin}\left\{\|\delta_N(x_N + \Delta x_N)\|^2 + \lambda\|\delta_B \Delta x_B\|^2 : A\Delta x = 0\right\}, \tag{28}$$

where $\delta \equiv \delta(w)$ and $(B, N) \equiv (B(w), N(w))$. Note that this direction is well-defined in the sense that the above optimization problem has a unique optimal solution. Now, let $\psi_p : \Re_{++} \to \Re_+$ denote the mapping given by

$$\psi_p(\lambda) \equiv \frac{\|\delta_B \Delta x_B(\lambda)\|}{\sqrt{\mu}}, \tag{29}$$

where $\mu \equiv \mu(w)$.

The following technical result can be proved regarding the functions $\Delta x(\lambda)$ and $\psi_p(\lambda)$. Note that in the discussion below, we denote the derivatives of $\Delta x(\lambda)$ and $\psi_p(\lambda)$ as $\Delta x'(\lambda)$ and $\psi_p'(\lambda)$ respectively.

Lemma 2.10 *The following statements hold:*

a) The limits $\Delta x(0) \equiv \lim_{\lambda\to 0^+} \Delta x(\lambda)$ *and* $\psi_p(0) \equiv \lim_{\lambda\to 0^+} \psi_p(\lambda)$ *exist and are given by*

$$\Delta x_N(0) = \operatorname{argmin}\left\{\|\delta_N(x_N + \Delta x_N)\|^2 : A_N \Delta x_N \in \operatorname{Im}(A_B)\right\}, \tag{30}$$

$$\Delta x_B(0) = \operatorname{argmin}\left\{\|\delta_B \Delta x_B\|^2 : A_B \Delta x_B = -A_N \Delta x_N(0)\right\}, \tag{31}$$

$$\psi_p(0) = \|\delta_B \Delta x_B(0)\|/\sqrt{\mu}. \tag{32}$$

b) The limit $\Delta x_B'(0) \equiv \lim_{\lambda\to 0^+} \Delta x_B'(\lambda)$ *exists. Moreover, if* $\psi_p(0) \neq 0$ *then* $\psi_p'(0) \equiv \lim_{\lambda\to 0^+} \psi_p'(\lambda)$ *also exists;*

c) If $\psi_p(0) \neq 0$*, then the function* $\psi_p(\cdot)$ *is strictly convex, strictly decreasing and* $\lim_{\lambda\to\infty} \psi_p(\lambda) = 0$*; otherwise, if* $\psi_p(0) = 0$*, then the function* $\psi_p(\cdot)$ *is identically zero.*

d) If $0 < \lambda_1 \leq \lambda_2$ *then* $\psi_p(\lambda_1)/\psi_p(\lambda_2) \leq \lambda_2/\lambda_1$ *(with the convention that* $0/0 = 0$*).*

We note that, in view of Lemma 2.10, the functions $\Delta x(\lambda)$ and $\psi_p(\lambda)$ can be extended to $\lambda = 0$ and their extensions are continuously differentiable at $\lambda = 0$. The following result relates the direction $\Delta x(\lambda)$ above to the primal TR direction, i.e., the optimal solutions of (22).

Lemma 2.11 *The following statements hold:*

a) For any $\lambda > 0$, $\Delta x(\lambda)$ is an optimal solution of (22) with $\gamma_p = \psi_p(\lambda)$;

b) $\Delta x(0)$ is an optimal solution of (22) for any $\gamma_p \geq \psi_p(0)$.

Proof. a) Using the fact that $\Delta x(\lambda)$ satisfies the optimality conditions for (28), we easily see that it also satisfies the optimality conditions, and hence is an optimal solution, of (22) with $\gamma_p = \psi_p(\lambda)$.

b) In view of Lemma 2.10, we can pass the optimality conditions of (28) to the limit as $\lambda \downarrow 0$ to conclude that $A\Delta x(0) = 0$ and $\delta_N^2(x_N + \Delta x_N(0)) \in \mathrm{Im}(A^T)$. This together with (29) and the assumption that $\gamma_p \geq \psi_p(0)$ imply that $\Delta x(0)$ satisfies the optimality conditions, and hence is an optimal solution, of (22). ∎

Using the above results, the primal TR direction required by the algorithm can be computed as follows. Recall that the goal is to find an optimal solution of (22) for some $\gamma_p \in [\gamma/2, 2\gamma]$. We start by computing $\Delta x(0)$ and then $\psi_p(0)$. If $\psi_p(0) \leq 2\gamma$ then by Lemma 2.11(b), we conclude that $\Delta x(0)$ is an optimal solution of (22) with $\gamma_p = 2\gamma$, and hence can be chosen as the required TR direction. Otherwise, if $\psi_p(0) > 2\gamma$, we search for some $\lambda_p > 0$ such that

$$\gamma/2 \leq \psi_p(\lambda_p) \leq 2\gamma, \tag{33}$$

which always exists in view of Lemma 2.10(c) and the fact that $\psi_p(0) > 2\gamma$.

Now, to find some $\lambda_p > 0$ satisfying (33), it suffices to determine $0 < \lambda_l \leq \lambda_u$ such that

$$\psi_p(\lambda_l) \geq \gamma \geq \psi_p(\lambda_u), \tag{34}$$

$$\lambda_u/\lambda_l \leq 2. \tag{35}$$

In such a case, any scalar $\lambda_p \in [\lambda_l, \lambda_u]$ satisfies (33). Indeed, by Lemma 2.10(d), we have

$$\frac{\gamma}{2} \leq \frac{\lambda_l}{\lambda_u}\gamma \leq \frac{\lambda_l}{\lambda_p}\psi_p(\lambda_l) \leq \psi_p(\lambda_p) \leq \frac{\lambda_u}{\lambda_p}\psi_p(\lambda_u) \leq \frac{\lambda_u}{\lambda_l}\gamma \leq 2\gamma.$$

Assuming that initial λ_l and λ_u satisfying (34) are given, a standard bisection procedure on the $(\log\lambda)$-space can then be used to determine scalars λ_l and λ_u satisfying both (34) and (35). An iteration of this bisection scheme updates λ_l or λ_u as follows. First, compute $\tilde{\lambda}$ such that $\log\tilde{\lambda} = (\log\lambda_l + \log\lambda_u)/2$, that is $\tilde{\lambda} = (\lambda_l\lambda_u)^{1/2}$. Second, if $\psi_p(\tilde{\lambda}) > \gamma$, λ_l is updated to $\tilde{\lambda}$; otherwise λ_u is updated to $\tilde{\lambda}$. It is clear that each iteration of this bisection scheme always preserves condition (34) and halves the length of the interval $[\log\lambda_l, \log\lambda_u]$. Hence, it eventually finds a pair (λ_l, λ_u) satisfying (34) and (35) in $\mathcal{O}(\log(\log(\lambda_u/\lambda_l)))$ bisection iterations, where λ_l and λ_u are the initial values of these scalars at the start of the procedure.

It remains to describe how to choose initial scalars $0 < \lambda_l \leq \lambda_u$ such that (34) holds. We first focus our attention on the description of λ_l. Since $\psi_p(\lambda)$ is convex, we have $\psi_p(\lambda) \geq \psi_p(0) + \psi_p'(0)\lambda$ for every $\lambda > 0$. Hence, choosing λ_l to be the root of the linear equation $\psi_p(0) + \psi_p'(0)\lambda = \gamma$, i.e.,

$$\lambda_l = \frac{\psi_p(0) - \gamma}{|\psi_p'(0)|},$$

we conclude that $\psi_p(\lambda_l) \geq \gamma$.

The following lemma provides the needed information to obtain a lower bound on λ_l. We observe that, in spite of the notation, the quantities $\psi_p(0)$ and $\psi_p'(0)$ depend on the point $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$.

Lemma 2.12 *Let w_0 denote the initial iterate of the PC-TR algorithm and set $\mu_0 = \mu(w_0)$. Then, for any $w \in \mathcal{N}(\beta)$ such that $\mu := \mu(w) \leq \mu_0$, we have:*

$$\frac{\psi_p(0)}{|\psi_p'(0)|} \geq \frac{(1-\beta)^8 \mu^2}{n^4 (1+\beta)^4 \mu_0^2 \, \bar{\chi}^2_{A\delta_0^{-1}}}. \tag{36}$$

Using the above result, a lower bound on λ_l can be obtained by observing that, under the assumption that $\psi_p(0) \geq 2\gamma$, we have

$$\lambda_l = \frac{\psi_p(0) - \gamma}{|\psi_p'(0)|} \geq \frac{\psi_p(0)}{2\,|\psi_p'(0)|} \geq \frac{(1-\beta)^8 \mu^2}{2\, n^4\, (1+\beta)^4 \mu_0^2 \, \bar{\chi}^2_{A\delta_0^{-1}}}. \tag{37}$$

We now discuss how to choose the initial scalar λ_u satisfying (34). In contrast to the choice of λ_l, there is no clear way of choosing λ_u for an arbitrary curve $\psi_p(\lambda)$. Fortunately, the PC-TR algorithm stated in the previous subsection can be slightly modified so as to compute the TR step only when the condition $\psi_p(1) \leq \gamma$ (in addition to the previously required condition that $\varepsilon_2^{\mathrm{a}}(w) > \varepsilon_0$) is satisfied. Hence, we may always choose the initial λ_u to be 1. In view of our discussion above, we conclude that the computation of a TR step in this variant of the PC-TR algorithm requires $\mathcal{O}\left(\log\log \lambda_l^{-1}\right)$ bisection steps, which is bounded by

$$\mathcal{O}\left[\log\left(\log \bar{\chi}_{A\delta_0^{-1}} + \log \frac{\mu_0}{\mu}\right)\right], \tag{38}$$

bisection steps, in view of (37).

We will now precisely discuss the variant of the PC-TR algorithm mentioned in the previous paragraph. First, we mention that the whole discussion of this subsection up to this point also applies to the computation of the dual TR direction, with the dual auxiliary direction

$$(\Delta y(\lambda), \Delta s(\lambda)) := \operatorname{argmin}\left\{\|\delta_B^{-1}(s_B + \Delta s_B)\|^2 + \lambda \|\delta_N^{-1} \Delta s_N\|^2 : A^T \Delta y + \Delta s = 0\right\} \tag{39}$$

replacing $\Delta x(\lambda)$ and the dual curve

$$\psi_d(\lambda) \equiv \frac{\|\delta_N^{-1} \Delta s_N(\lambda)\|}{\sqrt{\mu}} \tag{40}$$

replacing $\psi_p(\lambda)$. We then have the following convergence result about a certain variant of the PC-TR algorithm.

Theorem 2.13 *Consider the variant of the PC-TR algorithm where step 2) is replaced by the following step:*

2') If $\varepsilon_2^{\mathrm{a}}(w) > \varepsilon_0$ and $\max\{\psi_p(1), \psi_d(1)\} > \gamma/18$, then set $w \leftarrow w + \alpha_a \Delta w^{\mathrm{a}}$, where α_a is defined as in (10) and go to 6);

Then, the conclusions of Theorem 2.9 also hold for the resulting variant of the PC-TR algorithm.

We will now briefly discuss the arithmetic complexity of the above variant. We will see later in Section 4 that the bisection procedure to compute a TR step takes

$$T(\mu; w_0) \equiv \mathcal{O}\left[n^3 + n \log\left(\log \bar{\chi}_{A\delta_0^{-1}} + \log(\mu_0/\mu)\right)\right] \tag{41}$$

arithmetic operations, since the procedure requires $\mathcal{O}[\log(\log \bar{\chi}_{A\delta_0^{-1}} + \log(\mu_0/\mu))]$ evaluations of the curves $\psi_p(\lambda)$ and $\psi_d(\lambda)$ with the first evaluation of either curve taking $\mathcal{O}(n^3)$ arithmetic operations and subsequent ones taking only $\mathcal{O}(n)$ arithmetic operations.

The above observation together with Theorem 2.13 yields the following arithmetic complexity result for the above PC-TR variant.

Theorem 2.14 *The number of arithmetic operations performed by the variant of the PC-TR algorithm stated in Theorem 2.13 to find an iterate w such that $\mu(w) \leq \mu_f$ is bounded by*

$$\mathcal{O}\left[\frac{n^3 \log(\bar{\chi}_A^* + n + \varepsilon_0^{-1})}{\log \varepsilon_0^{-1}} T(\mu_f, w_0) + n^{6.5} \log(\bar{\chi}_A^* + n + \varepsilon_0^{-1})\right],$$

where $T(\cdot,\cdot)$ is defined in (41). In particular, if $\varepsilon_0^{-1} = \Theta((n + \bar{\chi}_A^*)^\kappa)$ for some $\kappa > 0$, the above arithmetic complexity bound reduces to

$$\mathcal{O}\left[n^{6.5} \log(\bar{\chi}_A^* + n) + n^4 \log\left(\log \bar{\chi}_{A\delta_0^{-1}} + \log(\mu_0/\mu_f)\right)\right].$$

3 Basic Tools

In this section we introduce the basic tools that will be used in the proof of Theorem 2.9. The analysis heavily relies on the notion of layered least squares(LLS) directions and crossover events due to Vavasis and Ye [29]. Subsection 3.1 below gives the definition of a crossover event which is slightly different than the one used in [29] and discusses some of its properties. Subsection 3.2 defines the layered least squares directions that will be used in the complexity analysis and also states an approximation result that provides an estimation of the closeness between the LLS direction with respect to a partition J of $\{1, \ldots, n\}$ and the AS direction. Subsection 3.3 reviews from a different perspective an important result from [29], namely Lemma 17 of [29], that essentially guarantees the occurrence of crossover events. Since this result is stated in terms of the residual of an LLS step, the use of the approximation result of Subsection 3.2 between the AS and LLS steps allows us to obtain a similar result stated in terms of the residual of the AS direction.

3.1 Crossover events

In this subsection we discuss the notion of crossover event which plays a fundamental role in our convergence analysis.

Definition: For two indices $i, j \in \{1, \ldots, n\}$ and a constant $\mathcal{C} \geq 1$, a *$\mathcal{C}$-crossover event* for the pair (i, j) is said to occur on the interval $(\nu', \nu]$ if

$$\begin{aligned} &\text{there exists } \nu_0 \in (\nu', \nu] \text{ such that } \frac{s_j(\nu_0)}{s_i(\nu_0)} \leq \mathcal{C}, \\ &\text{and, } \frac{s_j(\tilde{\nu})}{s_i(\tilde{\nu})} > \mathcal{C} \text{ for all } \tilde{\nu} \leq \nu'. \end{aligned} \tag{42}$$

Moreover, the interval $(\nu', \nu]$ is said to contain a $\mathcal{C}$-crossover event if (42) holds for some pair (i, j).

Hence, the notion of a crossover event is independent of any algorithm and is a property of the central path only. Note that in view of (3), condition (42) can be reformulated into an equivalent condition involving only the primal variable. For our purposes, we will use only (42).

We have the following simple but crucial result about crossover events.

Proposition 3.1 *Let $\mathcal{C} > 0$ be a given constant. There can be at most $n(n-1)/2$ disjoint intervals of the form $(\nu', \nu]$ containing $\mathcal{C}$-crossover events.*

The notion of $\mathcal{C}$-crossover events can be used to define the notion of $\mathcal{C}$-crossover events between two iterates of the PC-TR algorithm as follows. We say that a $\mathcal{C}$-crossover event occurs between two iterates w^k and w^l, $k < l$, generated by the PC-TR algorithm if the interval $(\mu(w^l), \mu(w^k)]$ contains a $\mathcal{C}$-crossover event. Note that in view of Proposition 3.1, there can be at most $n(n-1)/2$ intervals of this type. We will show in the remaining part of this paper that there exists a constant $\mathcal{C} > 0$ with the following property: for any index k, there exists an index $l > k$ such that $l - k = \mathcal{O}(n^{1.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ and a $\mathcal{C}$-crossover event occurs between the iterates w^k and w^l of the PC-TR algorithm. Proposition 3.1 and a simple argument then show that the PC-TR algorithm must terminate within $\mathcal{O}(n^{3.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations.

3.2 The Layered Least Squares Step

In this subsection we describe another type of direction, namely the layered least squares (LLS) step, which is very important in the analysis of our algorithm. This step was first introduced by Vavasis and Ye in [29]. We also describe two ordered partitions of the index set $\{1, \ldots, n\}$ that are crucial in the definition of the LLS directions.

Let $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ and a partition $(J_1, ..., J_p)$ of the index set $\{1, \ldots, n\}$ be given and define $\delta \equiv \delta(w)$. The primal LLS direction $\Delta x^{\rm ll} = (\Delta x^{\rm ll}_{J_1}, \ldots, \Delta x^{\rm ll}_{J_p})$ at w with the respect to J is defined recursively according to the order $\Delta x^{\rm ll}_{J_p}, \ldots, \Delta x^{\rm ll}_{J_1}$ as follows. Assume that the components $\Delta x^{\rm ll}_{J_p}, \ldots, \Delta x^{\rm ll}_{J_{k+1}}$ have been determined. Let $\Pi_{J_k} : \Re^n \to \Re^{J_k}$ denote the projection map defined as $\Pi_{J_k}(u) = u_{J_k}$ for all $u \in \Re^n$. Then $\Delta x^{\rm ll}_{J_k} \equiv \Pi_{J_k}(L^x_k)$ where L^x_k is given by

$$\begin{aligned} L^x_k &\equiv \mathrm{Argmin}_{u \in \Re^n} \left\{ \|\delta_{J_k}(x_{J_k} + u_{J_k})\|^2 : u \in L^x_{k-1} \right\} \qquad (43) \\ &= \mathrm{Argmin}_{u \in \Re^n} \left\{ \|\delta_{J_k}(x_{J_k} + u_{J_k})\|^2 : u \in \mathrm{Ker}(A),\ u_{J_i} = \Delta x^{\rm ll}_{J_i} \ \text{ for all } i = k+1, \ldots, p \ \right\}, \end{aligned}$$

with the convention that $L^x_0 = \mathrm{Ker}(A)$. The slack component $\Delta s^{\rm ll} = (\Delta s^{\rm ll}_{J_1}, \ldots, \Delta s^{\rm ll}_{J_p})$ of the dual LLS direction $(\Delta y^{\rm ll}, \Delta s^{\rm ll})$ at w with the respect to J is defined recursively as follows. Assume that the components $\Delta s^{\rm ll}_{J_1}, \ldots, \Delta s^{\rm ll}_{J_{k-1}}$ have been determined. Then $\Delta s^{\rm ll}_{J_k} \equiv \Pi_{J_k}(L^s_k)$ where L^s_k is given by

$$\begin{aligned} L^s_k &\equiv \mathrm{Argmin}_{v \in \Re^n} \left\{ \|\delta^{-1}_{J_k}(s_{J_k} + v_{J_k})\|^2 : v \in L^s_{k-1} \right\} \qquad (44) \\ &= \mathrm{Argmin}_{v \in \Re^n} \left\{ \|\delta^{-1}_{J_k}(s_{J_k} + v_{J_k})\|^2 : v \in \mathrm{Im}(A^T),\ v_{J_i} = \Delta s^{\rm ll}_{J_i} \ \text{ for all } i = 1, \ldots, k-1 \ \right\}, \end{aligned}$$

with the convention that $L^s_0 = \mathrm{Im}(A^T)$. Finally, once $\Delta s^{\rm ll}$ has been determined, the component $\Delta y^{\rm ll}$ is determined from the relation $A^T \Delta y^{\rm ll} + \Delta s^{\rm ll} = 0$.

Note that (11) and (12) imply that the AS direction is a special LLS direction, namely the one with respect to the only partition in which $p = 1$. Clearly, the LLS direction at a given $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ depends on the partition $J = (J_1, \ldots, J_p)$ used.

A partition $J = (J_1, \ldots, J_p)$ of $\{1, \ldots, n\}$ is said to be *ordered* with respect to a fixed vector $z \in \Re^n_{++}$ if $\max(z_{J_i}) \leq \min(z_{J_{i+1}})$ for all $i = 1, \ldots, p-1$. In such a case, we define the gap of J with respect to z as

$$\text{gap}(z, J) \;:=\; \min_{1 \leq i \leq p-1} \left\{ \frac{\min(z_{J_{i+1}})}{\max(z_{J_i})} \right\} \geq 1,$$

with the convention that $\text{gap}(z, J) = \infty$ if $p = 1$. We say that a partition J is ordered at $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ if it is ordered with respect to $z = \delta(w)$, in which case we denote the quantity $\text{gap}(\delta(w), J)$ simply by $\text{gap}(w, J)$. For partition $J = (J_1, \ldots, J_p)$ and a point $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$, the spread of the layer J_k with respect to w is defined as

$$\text{spr}(w, J_k) \equiv \frac{\max(\delta_{J_k}(w))}{\min(\delta_{J_k}(w))}, \quad \forall k = 1, \ldots, p.$$

We now state how the AS direction can be well approximated by suitable LLS steps. Lemma 3.2, whose proof can be found in [15], essentially states that the larger the gap of J is, the closer the AS direction and the LLS direction with respect to J will be to one another.

Lemma 3.2 *Let $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ and an ordered partition $J = (J_1, \ldots, J_p)$ at w be given. Define $\delta \equiv \delta(w)$ and let $\Delta w^{\text{a}} = (\Delta x^{\text{a}}, \Delta y^{\text{a}}, \Delta s^{\text{a}})$ and $\Delta w^{\text{ll}} = (\Delta x^{\text{ll}}, \Delta y^{\text{ll}}, \Delta s^{\text{ll}})$ denote the AS direction at w and the LLS direction at w with respect to J, respectively. If $\text{gap}(w, J) \geq 4\, p\, \bar{\chi}_A$, then*

$$\max \left\{ \left\| Rx^{\text{a}}(w) - Rx^{\text{ll}}(w) \right\|_\infty , \left\| Rs^{\text{a}}(w) - Rs^{\text{ll}}(w) \right\|_\infty \right\} \;\leq\; \frac{12\sqrt{n}\;\bar{\chi}_A}{\text{gap}(w, J)},$$

where $(Rx^{\text{a}}(w), Rs^{\text{a}}(w))$ and $(Rx^{\text{ll}}(w), Rs^{\text{ll}}(w))$ denote the residual pairs for the AS direction Δw^{a} and the LLS direction Δw^{ll}, respectively.

In the remainder of this subsection, we describe the two important LLS directions in the analysis of our algorithm that differs in the definition of ordered partitions. The first ordered partition is due to Vavasis and Ye [29]. Given a point $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ and a parameter $\bar{g} \geq 1$, this partition, which we refer to as the *VY partition*, is defined as follows. Let $(i_1, \ldots, i_n)$ be an ordering of $\{1, \ldots, n\}$ such that $\delta_{i_1} \leq \ldots \leq \delta_{i_n}$, where $\delta = \delta(w)$. For $k = 2, \ldots, n$, let $r_k \equiv \delta_{i_k}/\delta_{i_{k-1}}$ and define $r_1 \equiv \infty$. Let $k_1 < \ldots < k_p$ be all the indices k such that $r_k > \bar{g}$. The VY $\bar{g}$-partition J is then defined as $J = (J_1, \ldots, J_p)$, where $J_q \equiv \{i_{k_q}, i_{k_q+1}, \ldots, i_{k_{q+1}-1}\}$ for all $q = 1, \ldots, p$. More generally, given a subset $I \subset \{1, \ldots, n\}$, we can similarly define the *VY $\bar{g}$-partition* of I at w by taking an ordering $(i_1, \ldots, i_m)$ of I satisfying $\delta_{i_1} \leq \ldots \leq \delta_{i_m}$ where $m = |I|$, defining the ratios $r_1, \ldots, r_m$ as above, and proceeding exactly as in the construction above to obtain the partition $J = (J_1, \ldots, J_p)$ of I.

It is easy to see that the following result holds for the partition J described in the previous paragraph.

Proposition 3.3 *Given a subset $I \subseteq \{1, ..., n\}$, a point $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ and a constant $\bar{g} \geq 1$, the VY $\bar{g}$-partition $J = (J_1, \ldots, J_p)$ of I at w satisfies $\text{gap}(w, J) > \bar{g}$ and $\text{spr}(w, J_q) \leq \bar{g}^{|J_q|} \leq \bar{g}^n$ for all $q = 1, \ldots, p$.*

The second ordered partition, which is used heavily in our analysis, was introduced by Monteiro and Tsuchiya [15]. Given a point $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$. First, we compute the bipartition (B, N) of $\{1, \ldots, n\}$ according to (20). Next, an order $(i_1, \ldots, i_n)$ of the index variables is chosen such that $\delta_{i_1} \leq \ldots \leq \delta_{i_n}$. Then, the first block of consecutive indices in the n-tuple $(i_1, \ldots, i_n)$ lying in the same set B or N are placed in the first layer $\mathcal{J}_1$, the next block of consecutive indices lying in the other set is placed in $\mathcal{J}_2$, and so on. As an example assume that $(i_1, i_2, i_3, i_4, i_5, i_6, i_7) \in B \times B \times N \times B \times B \times N \times N$. In this case, we have $\mathcal{J}_1 = \{i_1, i_2\}$, $\mathcal{J}_2 = \{i_3\}$, $\mathcal{J}_3 = \{i_4, i_5\}$ and $\mathcal{J}_4 = \{i_6, i_7\}$. A partition obtained according to the above construction is clearly ordered at w. We refer to it as an *ordered AS-partition*, and denote it by $\mathcal{J} = \mathcal{J}(w)$.

Note that an ordered AS–partition is not uniquely determined since there can be more than one n-tuple $(i_1, \ldots, i_n)$ satisfying $\delta_{i_1} \leq \ldots \leq \delta_{i_n}$. This situation happens exactly when there are two or more indices i with the same value for δ_i. If these tying indices do not all belong to the same set B or N then there will be more than one way to generate an ordered AS–partition $\mathcal{J}$.

We say that the bipartition (B, N) is *regular* if there do not exist $i \in B$ and $j \in N$ such that $\delta_i = \delta_j$. Observe that there exists a unique ordered AS–partition if and only if (B, N) is regular. When (B, N) is not regular, our algorithm avoids the computation of an ordered AS–partition and hence of any LLS direction with respect to such a partition.

3.3 Relation between crossover events, the AS step and the LLS step

In this subsection, we state some variants of Lemma 17 of Vavasis and Ye [29]. Specifically, we present two estimates on the number of iterations needed to guarantee the occurrence of a crossover event. While the first estimate essentially depends on the size of the residual of the LLS step and the step-size at the initial iterate, the second one depends only on the size of the residual of the AS direction at the initial iterate. Lemma 3.4 is a restatement of Lemma 17 of of Vavasis and Ye [29]. Its proof can be found in Lemma 3.4 of Monteiro and Tsuchiya [15].

Lemma 3.4 *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1)$ and an ordered partition $J = (J_1, \ldots, J_p)$ at w be given. Let $\delta \equiv \delta(w)$, $\mu = \mu(w)$ and $(Rx^{\rm ll}(w), Rs^{\rm ll}(w))$ denote the residual of the LLS direction $(\Delta x^{\rm ll}, \Delta y^{\rm ll}, \Delta s^{\rm ll})$ at w with respect to J. Then, for any $q = 1, \ldots, p$ and any constant*

$$\mathcal{C}_q \geq (1 + \beta)\, \mathrm{spr}(w, J_q)/(1 - \beta)^2$$

and for any $\mu' \in (0, \mu)$ such that

$$\frac{\mu'}{\mu} \leq \frac{\|Rx^{\rm ll}_{J_q}(w)\|_\infty \|Rs^{\rm ll}_{J_q}(w)\|_\infty}{n^3 \mathcal{C}_q^2 \bar{\chi}_A^2},$$

the interval $(\mu', \mu]$ contains a $\mathcal{C}_q$-crossover event.

The following lemma is the immediate consequence of Lemma 3.4 and an adaption from Lemma 3.5 of Monteiro and Tsuchiya [15].

Lemma 3.5 *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1/4]$ and an ordered partition $J = (J_1, \ldots, J_p)$ at w be given. Define $\delta \equiv \delta(w)$ and $\mu = \mu(w)$, and let $(Rx^{\rm ll}(w), Rs^{\rm ll}(w))$ denote the residual of the LLS direction $(\Delta x^{\rm ll}, \Delta y^{\rm ll}, \Delta s^{\rm ll})$ at w with respect to J. Then, for every $q \in \{1, \ldots, p\}$ and every*

$$\mathcal{C}_q \geq (1 + \beta)\mathrm{spr}(w, J_q)/(1 - \beta)^2, \tag{45}$$

the following statements hold:

a) the PC-TR algorithm (or its variant) started from the point w will generate an iterate $\hat{w}$ with a $\mathcal{C}_q$-crossover event occurring between w and $\hat{w}$ in $\mathcal{O}(\sqrt{n}\,\Phi)$ iterations, where

$$\Phi \equiv \log(\bar{\chi}_A + n) + \log \mathcal{C}_q + \log\left(\frac{\mu_+/\mu}{\|Rx^{\mathrm{ll}}_{J_q}(w)\|_\infty \|Rs^{\mathrm{ll}}_{J_q}(w)\|_\infty}\right) \tag{46}$$

and μ_+ is the normalized duality gap attained immediately after the first iteration. Moreover, steps 3 through 5 of the PC-TR algorithm (or its variant), and hence computation of the TR step, is performed in only

$$\mathcal{O}(\Phi/\log(\varepsilon_0^{-1})) \tag{47}$$

of these iterations.

b) if, in addition,

$$\mathrm{gap}(w, J) \geq \max\left\{4\, n\bar{\chi}_A\,,\ \frac{24\sqrt{n}\bar{\chi}_A}{\varepsilon^{\mathrm{a}}_{J_q}}\right\} \tag{48}$$

where $\varepsilon^{\mathrm{a}}_{J_q} \equiv \min\left\{\|Rx^{\mathrm{a}}_{J_q}(w)\|_\infty\,,\ \|Rs^{\mathrm{a}}_{J_q}(w)\|_\infty\right\}$, then

$$\Phi = \mathcal{O}\left(\log(\bar{\chi}_A + n) + \log \mathcal{C}_q + \log(\varepsilon^{\mathrm{a}}_{J_q})^{-1}\right). \tag{49}$$

Proof. The proofs of the first part of statement a) and the whole statement b) are given in Lemma 3.5 of [15]. It remains to prove the latter part of statement a). We refer to an iteration of the PC-TR algorithm as a TR-iteration whenever the TR direction is computed. Let N_0 be the number of TR-iterations performed before reaching the first iterate $\hat{w}$ such that a $\mathcal{C}_q$-crossover event occurs between w and $\hat{w}$. We will show that N_0 is bounded by (47). Indeed, let $\tilde{w}$ denote the iterate obtained immediately after the $(N_0 - 1)$-th TR-iteration. Then, in view of Lemma 3.4, we have

$$\frac{\mu(\tilde{w})}{\mu(w)} > \frac{\|Rx^{\mathrm{ll}}_{J_q}(w)\|_\infty \|Rs^{\mathrm{ll}}_{J_q}(w)\|_\infty}{n^3 \mathcal{C}_q^2 \bar{\chi}_A^2}. \tag{50}$$

Since the duality gap is reduced by a factor of μ_+/μ in the first iteration, and by a factor of at least $\left((\sqrt{1+\beta}+\gamma)/(2\gamma)\right)\varepsilon_0$ in subsequent TR-iterations, due to relation (26), we conclude that

$$\log\left(\frac{\mu_+}{\mu}\right) + (N_0 - 2)\log\left(\frac{\sqrt{1+\beta}+\gamma}{2\gamma}\ \varepsilon_0\right) \geq \log\frac{\mu(\tilde{w})}{\mu(w)} > \log\left[\frac{\|Rx^{\mathrm{ll}}_{J_q}(w)\|_\infty \|Rs^{\mathrm{ll}}_{J_q}(w)\|_\infty}{n^3 \mathcal{C}_q^2 \bar{\chi}_A^2}\right],$$

which clearly implies that N_0 is bounded by (47). ∎

4 Convergence Analysis of the PC-TR Algorithm

In this section, we will provide the proofs of Theorems 2.9 and 2.13.

Lemma 3.5 gives a good idea of the effort that will be undertaken in this section, namely, to show that there exists a universal constant $\mathcal{C} = \mathcal{C}(\varepsilon_0) > 0$ with the property that, for each iterate

w of the PC-TR algorithm, or its variant, there exists an ordered partition $J = (J_1, \ldots, J_p)$ and an index $q = 1, \ldots, p$ such that $\mathcal{C} \geq (1+\beta)\text{spr}(w, J_q)/(1-\beta)^2$ and the quantity Φ defined in (46) with $\mathcal{C}_q = \mathcal{C}$ is bounded by $\mathcal{O}(n \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$. In view of Lemma 3.5(a), we would then conclude that a $\mathcal{C}$-crossover event occurs every time $\mathcal{O}(n^{1.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations of the PC-TR algorithm is performed. Proposition 3.1 together with the previous fact would then imply that the PC-TR algorithm, or its variant, terminates in $\mathcal{O}(n^{3.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations.

We start by introducing the aforementioned constant $\mathcal{C} = \mathcal{C}(\varepsilon_0)$ and another global constant used in this section. Let

$$\bar{g}(\varepsilon_0) \equiv \frac{24\, n \bar{\chi}_A}{\varepsilon_0}, \qquad \mathcal{C}(\varepsilon_0) \equiv \frac{(1+\beta)}{(1-\beta)^2} \left[\bar{g}(\varepsilon_0)\right]^n. \tag{51}$$

The proof of the above claim will be broken into three cases, namely: i) $\varepsilon_2^{\text{a}}(w) \geq \varepsilon_0$; ii) $\text{gap}(w, \mathcal{J}) \leq \bar{g}(\varepsilon_0)$, and; iii) $\text{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$ and $\varepsilon_2^{\text{a}}(w) \leq \varepsilon_0$, where $\varepsilon_2^{\text{a}}(w)$ is given by (21), $\mathcal{J}$ is the AS-partition at w and $\text{gap}(w, \mathcal{J})$ is defined in Subsection 3.2. The first result below considers the case i).

Lemma 4.1 *Suppose that $w \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1/4]$ and that $\varepsilon_2^{\text{a}}(w) \geq \varepsilon_0$ for some constant $\varepsilon_0 > 0$. Then PC-TR algorithm, or its variant, started from the point w will generate an iterate $\hat{w}$ with a $\mathcal{C}(\varepsilon_0)$-crossover event occurring between w and $\hat{w}$ in $\mathcal{O}(n^{1.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations, of which $\mathcal{O}(n \log(\bar{\chi}_A + n + \varepsilon_0^{-1}) / \log \varepsilon_0^{-1})$ are TR-iterations.*

Proof. The assumption that $\varepsilon_2^{\text{a}}(w) \geq \varepsilon_0$ implies $\varepsilon_\infty^{\text{a}}(w) \geq \varepsilon_0/\sqrt{n}$, and hence, in view of definition (16), there exists an index $i = 1, \ldots, n$ such that $\min\{|Rx_i^{\text{a}}(w)|, |Rs_i^{\text{a}}(w)|\} \geq \varepsilon_0/\sqrt{n}$. Now let $J = (J_1, \ldots, J_p)$ be a VY $\bar{g}(\varepsilon_0)$–partition at w and let J_q be the layer containing the index i above. Clearly, we have

$$\varepsilon_{J_q}^{\text{a}} \equiv \min\left\{\|Rx_{J_q}^{\text{a}}(w)\|_\infty, \|Rs_{J_q}^{\text{a}}(w)\|_\infty\right\} \geq \varepsilon_0/\sqrt{n}. \tag{52}$$

Using the above inequality, the fact that $\text{gap}(w, J) \geq \bar{g}(\varepsilon_0)$ and (51), we easily see that (48) holds. Since by Proposition 3.3 the spread of every layer of a VY $\bar{g}(\varepsilon_0)$–partition at w is bounded above by $\bar{g}(\varepsilon_0)^n$, we conclude that $\text{spr}(w, J_q) \leq \bar{g}^n$, and hence that the constant $\mathcal{C}(\varepsilon_0)$ defined in (51) satisfies (45) with $\mathcal{C}_q = \mathcal{C}(\varepsilon_0)$. We then conclude from Lemma 3.5(b) that $\Phi = \mathcal{O}(n^{1.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ in view of (52), (51) and the fact that $\log(\mathcal{C}_q) = \mathcal{O}(n \log \bar{g}(\varepsilon_0)) = \mathcal{O}(n \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$. The conclusion of the lemma now follows from the previous observation and Lemma 3.5(a). ∎

The next result takes care of case ii), namely the case in which $\text{gap}(w, \mathcal{J}) \leq \bar{g}(\varepsilon_0)$.

Lemma 4.2 *Suppose that $w \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1/4]$. Let $\bar{g}(\varepsilon_0)$ and $\mathcal{C}(\varepsilon_0)$ be the constants defined in (51). Let $\mathcal{J} = (\mathcal{J}_1, \ldots, \mathcal{J}_r)$ be an ordered AS-partition at w and assume that $\text{gap}(w, \mathcal{J}) \leq \bar{g}(\varepsilon_0)$. Then, the PC-TR algorithm, or its variant, started from the point w will generate an iterate $\hat{w}$ with a $\mathcal{C}(\varepsilon_0)$-crossover event occurring between w and $\hat{w}$ in $\mathcal{O}(n^{1.5} \log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations, of which $\mathcal{O}(n \log(\bar{\chi}_A + n + \varepsilon_0^{-1}) / \log \varepsilon_0^{-1})$ are TR-iterations.*

Proof. Assume that $\text{gap}(w, \mathcal{J}) \leq \bar{g}(\varepsilon_0)$ and let $J = (J_1, \ldots, J_p)$ be a VY $\bar{g}(\varepsilon_0)$–partition at w. Using the assumption that $\text{gap}(w, \mathcal{J}) \leq \bar{g}(\varepsilon_0)$, it is easy to see that there exist two indices i, j of different types, say $i \in B(w)$ and $j \in N(w)$, both lying in some layer J_q of J. By Lemma 2.6 and the definition of $(B(w), N(w))$ given in (20), it follows that $|Rx_i^{\text{a}}(w)| \geq 1/4$ and $|Rs_j^{\text{a}}(w)| \geq 1/4$, and hence that

$$\varepsilon_{J_q}^{\text{a}} \equiv \min\left\{\|Rx_{J_q}^{\text{a}}(w)\|_\infty\,,\, \|Rs_{J_q}^{\text{a}}(w)\|_\infty\right\} \geq \frac{1}{4}. \tag{53}$$

Using this inequality and the fact that $\mathrm{gap}(w,J) \geq \bar{g}(\varepsilon_0) \geq 96\bar{\chi}_A n$, where the last inequality is due to (51) and (25), we easily see that (48) holds. Since by Proposition 3.3 the spread of every layer of a VY $\bar{g}(\varepsilon_0)$–partition at w is bounded above by $\bar{g}(\varepsilon_0)^n$, we conclude that $\mathrm{spr}(w,J_q) \leq \bar{g}^n$, and hence that (45) holds with $\mathcal{C}_q = \mathcal{C}(\varepsilon_0)$ in view of (51). The result now follows from Lemma 3.5 by noting that the quantity Φ in (49) with $\mathcal{C}_q = \mathcal{C}(\varepsilon_0)$ is bounded by $\mathcal{O}(n^{1.5}\log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ in view of (51) and (53). ∎

From now on, we consider case iii), namely the case in which $\mathrm{gap}(w,\mathcal{J}) \geq \bar{g}(\varepsilon_0)$ and $\varepsilon_2^{\mathrm{a}}(w) \leq \varepsilon_0$.

We start by stating a technical result whose proof is given in Lemma 4.3 of [15] and holds for any $\bar{g}(\varepsilon_0) \geq 96n\bar{\chi}_A$, hence for our specific choice of $\bar{g}(\varepsilon_0)$ given in (51), in view of (25).

Lemma 4.3 *Suppose that $w \in \mathcal{N}(\beta)$ for some $\beta \in (0, 1/4]$. Let $\bar{g}(\varepsilon_0)$ and $\mathcal{C}(\varepsilon_0)$ be the constants defined in (51). Let $\mathcal{J} = (\mathcal{J}_1, \ldots, \mathcal{J}_r)$ denote the AS-partition at w and assume that $\mathrm{gap}(w,\mathcal{J}) \geq \bar{g}(\varepsilon_0)$. Let $(Rx^{\mathrm{l}}(w), Rs^{\mathrm{l}}(w))$ denote the residual of the LLS direction at w with respect to $\mathcal{J}$. Let*

$$\hat{\Phi} \equiv n\log\left(\bar{\chi}_A + n + \varepsilon_0^{-1}\right) + \log\left(\frac{\mu_+/\mu}{\varepsilon_\infty^{\mathrm{l}}(w)}\right), \tag{54}$$

μ_+ is the normalized duality gap attained immediately after the first iteration,

$$\varepsilon_\infty^{\mathrm{l}}(w) \equiv \max\left\{\left\|Rx_N^{\mathrm{l}}(w)\right\|_\infty, \left\|Rs_B^{\mathrm{l}}(w)\right\|_\infty\right\} \tag{55}$$

and $(B,N) \equiv (B(w), N(w))$. Then, the PC-TR algorithm started from the point w will generate an iterate $\hat{w}$ with a $\mathcal{C}(\varepsilon_0)$-crossover event occurring between w and $\hat{w}$ in $\mathcal{O}(\sqrt{n}\,\hat{\Phi})$ iterations, of which $\mathcal{O}(\hat{\Phi}/\log\varepsilon_0^{-1})$ are TR-iterations.

Our goal now will be to estimate, under the conditions of case iii), the second logarithm that appears in the iteration-complexity bound (54). In this estimation procedure, it is important to show that the first iteration from w is a TR-iteration. This will always be the case for the PC-TR algorithm since a TR-iteration occurs in this algorithm if and only if $\varepsilon_2^{\mathrm{a}}(w) \leq \varepsilon_0$ and case iii) assumes this condition. On the other hand, for the variant, TR-iteration occurs if and only if, in addition to $\varepsilon_2^{\mathrm{a}}(w) \leq \varepsilon_0$, we also have $\max\{\psi_p(1), \psi_d(1)\} \leq \gamma$, where the curves $\psi_p(\cdot)$ and $\psi_d(\cdot)$ are defined in (29) and (40). The next two results show that the latter condition also holds under case iii).

Given $F \in \Re^{m\times n}$, $h \in \Re^m$ and a scaling vector $z \in \Re^n_{++}$, consider the projection $p^0 \in \Re^n$ given by

$$p^0 \equiv \mathrm{argmin}_{p\in\Re^n}\left\{\|h-p\|^2 : FZp = 0\right\}, \tag{56}$$

where $Z \equiv \mathrm{Diag}(z)$. For a given ordered partition $J = (J_1, \ldots, J_l)$, Lemma 4.4 shows that if $\mathrm{gap}(z,J)$ is large then the projection matrix onto $\mathrm{Ker}(F\,\mathrm{Diag}(z))$ can be well approximated by a block diagonal matrix where each block is a projection matrix associated with some layer J_k of J. This fact was first established in [24] and an alternative proof can be found in [14]. The proof of a slightly stronger version of the variant stated below can be found in [17].

Lemma 4.4 *Let $F \in \Re^{m\times n}$, $h \in \Re^m$, $z \in \Re^n_{++}$ and an ordered partition $J = (J_1, \ldots, J_l)$ of $\{1, \ldots, n\}$ with respect to z be given. Define $p^0 \in \Re^n$ as in (56) and $\tilde{p}^0 \in \Re^n$ as*

$$\tilde{p}^0_{J_k} \equiv \mathrm{argmin}_{\tilde{p}_{J_k}\in\Re^{J_k}}\left\{\|\tilde{p}_{J_k} - h_{J_k}\|^2 : F_{J_k}Z_{J_k}\tilde{p}_{J_k} \in \mathrm{Im}(F_{\bar{J}_k})\right\}, \tag{57}$$

for every $k = 1, \ldots, l$, *where* $\bar{J}_k \equiv J_{k+1} \cup \ldots \cup J_l$. *Then,*

$$\|p^0 - \tilde{p}^0\|_\infty \leq K(3 + 2K)\|h\|, \tag{58}$$

where $K \equiv \bar{\chi}_F / \mathrm{gap}(z, J)$.

Using this approximation result, we are now able to prove the result mentioned just after Lemma 4.3.

Lemma 4.5 *Assume that* $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ *and that* $\mathrm{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$ *for some* $\varepsilon_0 \in (0, 12n]$, *where* $\mathcal{J} = (\mathcal{J}_1, \ldots, \mathcal{J}_r)$ *denotes the ordered AS-partition at* w. *Then, the curves* $\psi_p(\cdot)$ *and* $\psi_d(\cdot)$ *defined in (29) and (40), respectively, satisfy* $\max\{\psi_p(1), \psi_d(1)\} \leq \varepsilon_0/6$.

Proof. We will only show the inequality $\psi_p(1) \leq \varepsilon_0/6$. The proof of the inequality $\psi_d(1) \leq \varepsilon_0/6$ is similar. Consider the projections p^0 and $\tilde{p}^0$ defined in Lemma 4.4 with $F = A$, $h = (h_B, h_N) \equiv (0, \delta_N x_N)$, $z = \delta^{-1}$ and $J = (\mathcal{J}_r, \ldots, \mathcal{J}_1)$, where $\delta \equiv \delta(w)$. It is easy to see that the constant K of Lemma 4.4 is exactly equal to $\bar{\chi}_A / \mathrm{gap}(w, \mathcal{J})$. It then follows from relation (51) and the assumptions $\mathrm{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$ and $\varepsilon_0 \leq 12n$ that $K \leq \varepsilon_0/(24n) \leq 1/2$. Using these two inequalities, the conclusion of Lemma 4.4 and the fact that $\|h\| \leq \|\delta x\| = \sqrt{x^T s} = \sqrt{n\mu}$, we then obtain

$$\frac{1}{\sqrt{\mu}}\|p^0_B - \tilde{p}^0_B\| \leq \frac{\sqrt{n}}{\sqrt{\mu}}\|p^0_B - \tilde{p}^0_B\|_\infty \leq nK(3 + 2K) \leq 4nK \leq \varepsilon_0/6. \tag{59}$$

Moreover, definition (28) clearly implies that $p^0 = \delta \Delta x(1)$, where we recall that $\Delta x(1)$ is the optimal solution of (28) with $\lambda_p = 1$. Using the fact that $h_{J_k} = 0$ for every $J_k \subset B$ and the definition (57), we easily see that $\tilde{p}^0_{J_k} = 0$ for every $J_k \subset B$ and hence that $\tilde{p}^0_B = 0$. The last two observations together with (59) and (29) then imply that

$$\psi_p(1) \;=\; \frac{1}{\sqrt{\mu}}\|\delta_B \Delta x_B(1)\| = \frac{1}{\sqrt{\mu}}\|p^0_B\| = \frac{1}{\sqrt{\mu}}\|p^0_B - \tilde{p}^0_B\| \leq \varepsilon_0/6.$$

■

The following result follows an immediate consequence of Lemma 4.5.

Lemma 4.6 *Assume that* w *is an iterate of the PC-TR variant such that* $\varepsilon_2^{\mathrm{a}}(w) \leq \varepsilon_0$ *and* $\mathrm{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$. *Then, the iteration of the PC-TR variant from* w *is a TR-iteration.*

Proof. In view of Lemma 4.5 and the assumptions that $\mathrm{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$ and $\varepsilon_0 \leq \gamma/3$, we conclude that $\max\{\psi_p(1), \psi_d(1)\} \leq \gamma/18$. This inequality together with the assumption $\varepsilon_2^{\mathrm{a}}(w) \leq \varepsilon_0$, implies that the iteration of the PC-TR variant from w is a TR-iteration (see the statement of Theorem 2.13). ■

When a TR-iteration is performed, it follows from relation (26) that the duality gap is reduced by a factor bounded by $\mathcal{O}(\varepsilon_2^{\tau}(w; \gamma_p, \gamma_d))$. The following result shows that this factor is indeed $\mathcal{O}(\sqrt{n}\varepsilon^{\mathrm{l}}_\infty(w))$, where $\varepsilon^{\mathrm{l}}_\infty(w)$ is defined in (55), thereby giving the necessary means to bound the second logarithm which appears in (54).

Lemma 4.7 *Suppose that $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ is such that $\varepsilon_2^{\rm a}(w) \leq \varepsilon_0$. Let $\Delta w^l = (\Delta x^l, \Delta y^l, \Delta s^l)$ denote the LLS direction at w with respect to the AS-partition $\mathcal{J} = (\mathcal{J}_1, \ldots, \mathcal{J}_r)$ and assume that $\mathrm{gap}(\mathcal{J}) \geq \bar{g}(\varepsilon_0)$. Then, we have:*

$$\max\left\{ \frac{\|\delta_B \Delta x_B^l\|}{\sqrt{\mu}}, \frac{\|\delta_N^{-1} \Delta s_N^l\|}{\sqrt{\mu}} \right\} \leq \frac{3\varepsilon_0}{2}. \tag{60}$$

Moreover, if in addition $\varepsilon_0 \leq \gamma/3$, then

$$\varepsilon_2^{\tau}(w; \gamma_p, \gamma_d) \leq \sqrt{n}\varepsilon_\infty^{\rm l}(w) \tag{61}$$

for any $\gamma_p, \gamma_d \geq \gamma/2$, where $\varepsilon_\infty^{\rm l}(w)$ and $\varepsilon_2^{\tau}(w; \gamma_p, \gamma_d)$ are defined in (55) and (24), respectively.

Proof. Clearly, by definition (9) and (43) we have $A\Delta x^l = 0$. Moreover, from the triangle inequality for norms, Theorem 3.2, relations (14), (21) and (51) and the assumptions that $\mathrm{gap}(w, \mathcal{J}) \geq \bar{g}(\varepsilon_0)$ and $\varepsilon_2^{\rm a}(w) \leq \varepsilon_0$, we conclude that:

$$\begin{aligned} \frac{\|\delta_B \Delta x_B^l\|}{\sqrt{\mu}} &\leq \frac{\|\delta_B \Delta x_B^{\rm a}\|}{\sqrt{\mu}} + \frac{\|\delta_B(\Delta x_B^l - \Delta x_B^{\rm a})\|}{\sqrt{\mu}} \leq \|Rs_B^a\| + \sqrt{n}\,\|Rx^l - Rx^a\|_\infty \\ &\leq \varepsilon_2^{\rm a}(w) + \frac{12n\bar{\chi}_A}{\mathrm{gap}(w, \mathcal{J})} \leq \varepsilon_0 + \frac{12n\bar{\chi}_A}{\bar{g}(\varepsilon_0)} \leq \varepsilon_0 + \frac{\varepsilon_0}{2} \leq \frac{3\varepsilon_0}{2}. \end{aligned}$$

In a similar manner, we can also show that $\|\delta_N^{-1}\Delta s_N^l\|/\sqrt{\mu} \leq 3\varepsilon_0/2$, showing that (60) holds.

Assume now that $\varepsilon_0 \leq \gamma/3$ also holds. In view of (60), it follows that Δx_B^l and Δs_N^l are feasible for subproblems (22) and (23), respectively, whenever $\gamma_p, \gamma_d \geq \gamma/2$. Hence, we conclude that $\|Rx_N^\tau\| \leq \|Rx_N^l\|$ and $\|Rs_B^\tau\| \leq \|Rs_B^l\|$, from which it follows that

$$\varepsilon_2^{\tau}(w; \gamma_p, \gamma_d) := \max\{\|Rx_N^\tau\|, \|Rs_B^\tau\|\} \leq \sqrt{n}\max\{\|Rx_N^l\|_\infty, \|Rs_B^l\|_\infty\} = \sqrt{n}\varepsilon_\infty^{\rm l}(w).$$

■

We are now ready to prove Theorems 2.9 and 2.13.

Proof of Theorems 2.9 and 2.13. Let $\mathcal{C}$ and $\bar{g}(\varepsilon_0)$ be the constant defined in (51). We claim that the PC-TR algorithm started from any $w \in \mathcal{N}(\beta)$ generates an iterate $\hat{w}$ with a $\mathcal{C}(\varepsilon_0)$-crossover event occurring between w and $\hat{w}$ in $\mathcal{O}(n^{1.5}\log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iteration, of which $\mathcal{O}(n\log(\bar{\chi}_A + n + \varepsilon_0^{-1})/\log\varepsilon_0^{-1})$ are TR-iterations. Since by Proposition 3.1 there can be at most $n(n+1)/2$ $\mathcal{C}(\varepsilon_0)$-crossover events, we conclude that the PC-TR algorithm must ultimately terminate in $\mathcal{O}(n^{3.5}\log(\bar{\chi}_A + n + \varepsilon_0^{-1}))$ iterations, of which $\mathcal{O}(n^3\log(\bar{\chi}_A + n + \varepsilon_0^{-1})/\log\varepsilon_0^{-1})$ are TR-iterations. Let $\mathcal{J} = (\mathcal{J}_1, \ldots, \mathcal{J}_r)$ denote an AS-partition at w. We split the proof into one of the following three possible cases: 1) $\varepsilon_2^{\rm a}(w) \geq \varepsilon_0$; 2) $\mathrm{gap}(\mathcal{J}) \leq \bar{g}(\varepsilon_0)$, and; 3) $\varepsilon_2^{\rm a}(w) \leq \varepsilon_0$ and $\mathrm{gap}(\mathcal{J}) \geq \bar{g}(\varepsilon_0)$. The claim clearly holds for the first two cases due to Lemmas 4.1 and 4.2. Moreover, Lemma 4.3 implies that the claim also holds in the third case as long as we can show that the quantity $(\mu_+/\mu)/\varepsilon_\infty^{\rm l}(w)$ appearing in (54) is $\mathcal{O}(\sqrt{n})$. Indeed, assume that $\varepsilon_\infty^{\rm a}(w) \leq \varepsilon_0$ and $\mathrm{gap}(\mathcal{J}) \geq \bar{g}(\varepsilon_0)$. Then, the iteration from w for both the PC-TR algorithm and its variant is a TR-iteration in view of Lemma 4.6. Then, it follows from Proposition 2.8 and Lemma 4.7 that $(\mu_+/\mu)/\varepsilon_\infty^{\rm l}(w) = \mathcal{O}(\sqrt{n})$. ■

5 Arithmetic complexity for the PC-TR variant

In this section, we will provide the details of the several claims made on Subsection 2.6 and prove the main result stated in that subsection, namely Theorem 2.14.

We start by noting that the results stated in Subsection 2.6 remain invariant if elementary row operations are applied to the rows of A. Indeed, the condition that $A\Delta x(\lambda) = 0$ can be replaced by the condition that $\Delta x(\lambda)$ is in the null space of A, which remains invariant when the elementary row operations are performed on A. In this section, we will therefore freely perform elementary row operations to bring A to a more convenient form.

Let $w \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ be given. By placing the columns with indices in $B \equiv B(w)$ before the ones with indices in $N \equiv N(w)$, it is easy to see that there exists a sequence of elementary row operations which brings A to a matrix of the form

$$\begin{pmatrix} B & E \\ 0 & N \end{pmatrix}, \tag{62}$$

where $B \in \Re^{r_b \times |B|}$ and $N \in \Re^{r_n \times |N|}$ are full row rank matrices with $r_b \equiv \text{rank}(A_B)$ and $r_n \equiv m - r_b$. By performing further elementary row operations, we may assume that A contains an $m \times m$ identity matrix, or equivalently, after permuting columns of A if necessary, the matrices B, N and E have the form

$$B = [\ \tilde{B} \quad I\ ], \quad N = [\ I \quad \tilde{N}\ ], \quad E = [\ 0 \quad \tilde{E}\ ], \tag{63}$$

and hence, A has the form

$$A = \begin{pmatrix} \tilde{B} & I & 0 & \tilde{E} \\ 0 & 0 & I & \tilde{N} \end{pmatrix}, \tag{64}$$

where $\tilde{B} \in \Re^{r_b \times (|B| - r_b)}$, $\tilde{N} \in \Re^{r_n \times (|N| - r_n)}$, and $\tilde{E} \in \Re^{r_b \times (|N| - r_n)}$. Note that, by abuse of notation, we still denote the above matrix by A.

The following result, which is only used in the proof of Lemma 2.12, strongly uses the fact that A has the form (64). We observe however that the weaker form (62) of A is sufficient to establish the other results of Subsection 2.6.

Lemma 5.1 *For any positive diagonal $n \times n$ matrix D, there exists a matrix $W \in \Re^{|B| \times |N|}$ such that $E = BW$ and $\|D_B^{-1} W D_N\| \leq \bar{\chi}_{AD}$, where B and E are given by (63).*

Proof. We first prove the result for $D = I$. In this case, we choose W as

$$W = \begin{pmatrix} 0 & 0 \\ 0 & \tilde{E} \end{pmatrix}.$$

It is easy to see that $E = BW$ and that $\|W\| = \|\tilde{E}\| \leq \bar{\chi}_A$, where the last inequality follows as a consequence of Proposition 2.3(c).

Assume now that D is an arbitrary $n \times n$ positive diagonal matrix and let D_I denote the diagonal submatrix of D corresponding to the identity matrix of A. Also, let D_b and D_n denote the diagonal submatrices of D_I corresponding to the first $|B|$ columns and the last $|N|$ columns of the identity matrix of A, respectively. Then, the matrix given by

$$\hat{A} \equiv D_I^{-1} A D = \begin{pmatrix} D_b^{-1} B D_B & D_b^{-1} E D_N \\ 0 & D_n^{-1} N D_N \end{pmatrix}$$

also contains an $m \times m$ identity matrix. Applying the result shown in the first paragraph of this proof to the matrix $\hat{A}$, we conclude that there exists a matrix $\hat{W}$ such that $\|\hat{W}\| \leq \bar{\chi}_{\hat{A}} = \bar{\chi}_{AD}$ and $(D_b^{-1} B D_B)\hat{W} = D_b^{-1} E D_N$, or equivalently $B(D_B \hat{W} D_N^{-1}) = E$. The result now follows by letting $W = D_B \hat{W} D_N^{-1}$. ∎

The following lemma establishes some technical results about the direction $\Delta x(\lambda)$ defined in (28). Its proof is based on techniques developed in [24].

Lemma 5.2 *Let $\Delta x(\cdot)$ be the curve defined as in (28) and let $\Delta x'(\cdot)$ denote its derivative. Then:*

$$\Delta x_B(0) \equiv \lim_{\lambda \to 0^+} \Delta x_B(\lambda) = \Delta_B^{-2} B^T (B\Delta_B^{-2} B^T)^{-1} H \Delta_N x_N; \tag{65}$$

$$\Delta x_N(0) \equiv \lim_{\lambda \to 0^+} \Delta x_N(\lambda) = -\Delta_N^{-1} P_{N\Delta_N^{-1}} x_N; \tag{66}$$

$$\Delta x'_B(0) \equiv \lim_{\lambda \to 0^+} \Delta x'_B(\lambda) = \Delta_B^{-2} B^T (B\Delta_B^{-2} B^T)^{-1} H H^T (B\Delta_B^{-2} B^T)^{-1} H \Delta_N x_N; \tag{67}$$

where $\Delta_B \equiv \mathrm{Diag}\{\delta_B\}$, $\Delta_N \equiv \mathrm{Diag}\{\delta_N\}$, $H \equiv E\Delta_N^{-1} P_{N\Delta_N^{-1}}$ and $P_{N\Delta_N^{-1}}$ denotes the projection matrix onto the null space of $N\Delta_N^{-1}$.

Proof. Defining $D_\lambda \equiv \mathrm{Diag}\{\sqrt{\lambda}\,\delta_B, \delta_N\}$, we can easily see from (28) that $D_\lambda \Delta x(\lambda)$ is the projection of the vector $(0, \delta_N x_N)$ onto the null space of AD_λ^{-1}. Hence, for any $\lambda > 0$ we have:

$$\Delta x_B(\lambda) = \lambda^{-1} \Delta_B^{-2} A_B^T (AD_\lambda^{-2} A^T)^{-1} A_N x_N, \tag{68}$$

$$\Delta x_N(\lambda) = \Delta_N^{-2} A_N^T (AD_\lambda^{-2} A^T)^{-1} A_N x_N - x_N. \tag{69}$$

Using (62) and the definition of D_λ, we have

$$\begin{aligned} AD_\lambda^{-2} A^T &= \begin{pmatrix} B & E \\ 0 & N \end{pmatrix} \begin{pmatrix} \lambda^{-1}\Delta_B^{-2} & 0 \\ 0 & \Delta_N^{-2} \end{pmatrix} \begin{pmatrix} B^T & 0 \\ E^T & N^T \end{pmatrix} \\ &= \begin{pmatrix} \lambda^{-1} B\Delta_B^{-2} B^T + E\Delta_N^{-2} E^T & E\Delta_N^{-2} N^T \\ N\Delta_N^{-2} E^T & N\Delta_N^{-2} N^T \end{pmatrix} = \begin{pmatrix} \lambda^{-1} R_{BB} + R_{EE} & R_{EN} \\ R_{EN}^T & R_{NN} \end{pmatrix}, \end{aligned}$$

where

$$R_{BB} \equiv B\Delta_B^{-2} B^T, \quad R_{NN} \equiv N\Delta_N^{-2} N^T, \quad R_{EE} \equiv E\Delta_N^{-2} E^T, \quad R_{EN} \equiv E\Delta_N^{-2} N^T. \tag{70}$$

Using the standard way to compute the inverse of a 2×2 block matrix (see, for example, page 71-72 of [1]), it is easy to verify that

$$\left(AD_\lambda^{-2} A^T\right)^{-1} = \begin{pmatrix} U_\lambda & V_\lambda \\ V_\lambda^T & Z_\lambda \end{pmatrix},$$

where

$$U_\lambda = (\lambda^{-1} R_{BB} + R_{EE} - R_{EN} R_{NN}^{-1} R_{EN}^T)^{-1}, \tag{71}$$

$$V_\lambda = -U_\lambda R_{EN} R_{NN}^{-1}, \tag{72}$$

$$Z_\lambda = (R_{NN} - R_{EN}^T (\lambda^{-1} R_{BB} + R_{EE})^{-1} R_{EN})^{-1}. \tag{73}$$

Note that, by (70), we have

$$\begin{aligned} R_{EE} - R_{EN}R_{NN}^{-1}R_{EN}^T &= E\Delta_N^{-1}(I - \Delta_N^{-1}N^T(N\Delta_N^{-2}N^T)^{-1}N\Delta_N^{-1})\Delta_N^{-1}E^T \\ &= E\Delta_N^{-1}P_{N\Delta_N^{-1}}\Delta_N^{-1}E^T = HH^T. \end{aligned}$$

where we recall $H = E\Delta_N^{-1}P_{N\Delta_N^{-1}}$ and $P_{N\Delta_N^{-1}}$ denotes the projection matrix onto the null space of $N\Delta_N^{-1}$. Hence, by (71), we have

$$U_\lambda = (\lambda^{-1}R_{BB} + HH^T)^{-1} = \lambda(R_{BB} + \lambda HH^T)^{-1}. \tag{74}$$

Also, by (62) and (72), we have

$$\left(AD_\lambda^{-2}A^T\right)^{-1}A_N = \begin{pmatrix} U_\lambda E + V_\lambda N \\ V_\lambda^T E + Z_\lambda N \end{pmatrix} = \begin{pmatrix} U_\lambda(E - R_{EN}R_{NN}^{-1}N) \\ -R_{NN}^{-1}R_{EN}^T U_\lambda^T E + Z_\lambda N \end{pmatrix}. \tag{75}$$

Hence, using relations (68), (70), (74) and (75) and the definition of H, we obtain

$$\begin{aligned} \Delta x_B(\lambda) &= \lambda^{-1}\Delta_B^{-2}B^T U_\lambda(E - R_{EN}R_{NN}^{-1}N)x_N \\ &= \lambda^{-1}\Delta_B^{-2}B^T U_\lambda E\Delta_N^{-1}\left(I - \Delta_N^{-1}N^T(N\Delta_N^{-2}N^T)^{-1}N\Delta_N^{-1}\right)\Delta_N x_N \\ &= \lambda^{-1}\Delta_B^{-2}B^T U_\lambda E\Delta_N^{-1}P_{ND_N^{-1}}\Delta_N x_N \;=\; \Delta_B^{-2}B^T(R_{BB} + \lambda HH^T)^{-1}H\Delta_N x_N, \end{aligned} \tag{76}$$

from which we can easily see that (65) and (67) hold. Now, using relations (74) and (73), we easily see that

$$\lim_{\lambda\to 0^+} U_\lambda = 0, \quad \lim_{\lambda\to 0^+} Z_\lambda = R_{NN}^{-1}.$$

Relation (66) now follows from the last conclusion and relations (69) and (75). ∎

We need one more technical result before giving the proofs of the results of Subsection 2.6.

Lemma 5.3 *Let $G \in \Re^{p\times q}$ and $g \in \Re^q$ be given. Then, $(GG^T)Gg = 0$ if and only if $Gg = 0$.*

Proof. The assumption $(GG^T)Gg = 0$ clearly implies that $\|G^TGg\|^2 = 0$, and hence that $G^TGg = 0$. Also, the latter condition implies that $\|Gg\|^2 = 0$, or equivalently, $Gg = 0$. ∎

We are now ready to prove Lemma 2.10, Lemma 2.12, and Theorem 2.14 stated in Subsection 2.6.

Proof of Lemma 2.10. We first prove statements a) and b). The existence and characterizations of the two limits $\Delta x(0) \equiv \lim_{\lambda\to 0^+}\Delta x(\lambda)$ and $\Delta x'_B(0) \equiv \lim_{\lambda\to 0^+}\Delta x'_B(\lambda)$ were established in Lemma 5.2. The alternative characterization given by (30) and (31) of the limit $\Delta x(0) \equiv \lim_{\lambda\to 0^+}\Delta x(\lambda)$ can be easily proved by showing that $\Delta x_B(0)$ and $\Delta x_N(0)$ satisfy the optimality conditions, and hence are optimal solutions, of (30) and (31), respectively. Now, relation (32) follows immediately from (29). Moreover, differentiating (29) with respect to λ, we conclude that

$$\psi'_p(\lambda) = \frac{[\delta_B\Delta x_B(\lambda)]^T[\delta_B\Delta x'_B(\lambda)]}{\sqrt{\mu}\|\delta_B\Delta x_B(\lambda)\|} = \frac{[\delta_B\Delta x_B(\lambda)]^T[\delta_B\Delta x'_B(\lambda)]}{\mu\psi_p(\lambda)}. \tag{77}$$

Hence, under the the condition that $\psi_p(0) \neq 0$, the limit $\psi_p'(0) \equiv \lim_{\lambda \to 0^+} \psi_p'(\lambda)$ exists and is equal to the right hand side of (77) with $\lambda = 0$.

We now outline the proof of statement c). Using definition (29) and relation (76) we conclude that

$$\begin{aligned} \mu\psi_p(\lambda)^2 &= x_N^T \Delta_N H^T (R_{BB} + \lambda HH^T)^{-1} R_{BB} (R_{BB} + \lambda HH^T)^{-1} H \Delta_N x_N \\ &= x_N^T \Delta_N H^T R_{BB}^{-1/2} (I + \lambda R_{BB}^{-1/2} HH^T R_{BB}^{-1/2})^{-2} R_{BB}^{-1/2} H \Delta_N x_N, \\ &= \|(I + \lambda \tilde{H})^{-1} \tilde{g}\|^2, \end{aligned} \tag{78}$$

where $\mu \equiv \mu(w)$, $\tilde{H} \equiv R_{BB}^{-1/2} HH^T R_{BB}^{-1/2}$, and $\tilde{g} \equiv R_{BB}^{-1/2} H \Delta_N x_N$. The above formula for $\psi_p(\cdot)$ allows us to express it in terms of the eigenvalues and eigenvector of the positive semidefinite matrix $\tilde{H}$ and the resulting expression easily reveals that: i) $\tilde{H}\tilde{g} = 0$ if and only if $\psi_p(\cdot)$ is identically constant, and; ii) $\tilde{H}\tilde{g} \neq 0$ if and only if $\psi_p(\cdot)$ is strictly decreasing and strictly convex over the interval $(0, \infty)$. Moreover, if case i) occurs, it follows from Lemma 5.3 with $G = R_{BB}^{-1/2} H$ and $g = \Delta_N x_N$ that $0 = Gg = R_{BB}^{-1/2} H \Delta_N x_N$, and hence that $H \Delta_N x_N = 0$. In view of (65), this implies that $\Delta x_B(0) = 0$, and hence that $\psi_p(0) = 0$. We have thus shown that $\psi_p(\cdot)$ is indeed identically zero when case i) occurs.

We now show statement d). Let $0 < \lambda_1 \leq \lambda_2$ be given. By (78), we have

$$\frac{\psi_p^2(\lambda_1)}{\psi_p^2(\lambda_2)} = \frac{\tilde{g}^T (\lambda_1 \tilde{H} + I)^{-2} \tilde{g}}{\tilde{g}^T (\lambda_2 \tilde{H} + I)^{-2} \tilde{g}} = \frac{u^T \tilde{M} u}{u^T u}$$

where $u \equiv (\lambda_2 \tilde{H} + I)^{-1} \tilde{g}$ and $\tilde{M} = (\lambda_2 \tilde{H} + I)(\lambda_1 \tilde{H} + I)^{-2} (\lambda_2 \tilde{H} + I)$. Using the fact that $0 < \lambda_1 \leq \lambda_2$, we easily see that the largest eigenvalue of $\tilde{M}$, and hence $\psi_p^2(\lambda_1)/\psi_p^2(\lambda_2)$, is bounded by $(\lambda_2/\lambda_1)^2$. ∎

Proof of Lemma 2.12. First observe that, by (29), (77) and the Cauchy-Schwarz inequality, we have

$$\frac{|\psi_p'(0)|}{\psi_p(0)} = \frac{\left|[\delta_B \Delta x_B(0)]^T [\delta_B \Delta x_B'(0)]\right|}{\|\delta_B \Delta x_B(0)\|^2} \leq \frac{\|\delta_B \Delta x_B'(0)\|}{\|\delta_B \Delta x_B(0)\|}. \tag{79}$$

We will now use the formulas developed in Lemma 5.2 to bound the above ratio from above. Letting $\Delta_B \equiv \mathrm{Diag}(\delta_B)$ and $\Delta_N \equiv \mathrm{Diag}(\delta_N)$, we have that the matrix H defined in Lemma 5.2 can be written as

$$H \equiv E \Delta_N^{-1} P_{N\Delta_N^{-1}} = B \Delta_B^{-1} (\Delta_B W \Delta_N^{-1} P_{N\Delta_N^{-1}}) = B \Delta_B^{-1} M,$$

where $M \equiv \Delta_B W \Delta_N^{-1} P_{N\Delta_N^{-1}}$ and W is a matrix as in Lemma 5.1. Using this expression for H and relations (65) and (67), we then obtain

$$\begin{aligned} \delta_B \Delta x_B'(0) &= \Delta_B^{-1} B^T (B \Delta_B^{-2} B^T)^{-1} HH^T (B \Delta_B^{-2} B^T)^{-1} H \Delta_N x_N \\ &= \Delta_B^{-1} B^T (B \Delta_B^{-2} B^T)^{-1} \left(B \Delta_B^{-1} M\right) \left(M^T \Delta_B^{-1} B^T\right) (B \Delta_B^{-2} B^T)^{-1} H \Delta_N x_N \\ &= [\Delta_B^{-1} B^T (B \Delta_B^{-2} B^T)^{-1} B \Delta_B^{-1}] (MM^T) [\delta_B \Delta x_B(0)]. \end{aligned}$$

Using the fact that the matrix inside the first bracket in the right hand side of the above inequality is a projection matrix and Lemma 5.1, we then conclude that

$$\frac{\|\delta_B \Delta x_B'(0)\|}{\|\delta_B \Delta x_B(0)\|} \leq \|M\|^2 = \|\Delta_B W \Delta_N^{-1} P_{ND_N^{-1}}\|^2 \leq \|\Delta_B W \Delta_N^{-1}\|^2$$

$$\begin{aligned} &\leq \|\delta_B(\delta_0)_B^{-1}\|_\infty^2 \|(\Delta_0)_B W (\Delta_0)_N^{-1}\|^2 \|\delta_N^{-1}(\delta_0)_N\|_\infty^2 \\ &\leq [\bar{\chi}_{A\Delta_0^{-1}}]^2 \|\delta_B(\delta_0)_B^{-1}\|_\infty^2 \|\delta_N^{-1}(\delta_0)_N\|_\infty^2, \end{aligned} \tag{80}$$

where $\delta_0 \equiv \delta(w_0)$ and $\Delta_0 \equiv \mathrm{Diag}(\delta_0)$. Moreover, using the fact that $\mu \leq \mu_0$ together with Propositions 2.1 and 2.2, we conclude that

$$\|\delta_B(\delta_0)_B^{-1}\|_\infty \leq \frac{1+\beta}{(1-\beta)^2}\sqrt{\frac{\mu_0}{\mu}}\left\|\frac{s(\mu)}{s(\mu_0)}\right\|_\infty \leq \frac{(1+\beta)n}{(1-\beta)^2}\sqrt{\frac{\mu_0}{\mu}} \tag{81}$$

$$\|\delta_N^{-1}(\delta_0)_N\|_\infty \leq \frac{1+\beta}{(1-\beta)^2}\sqrt{\frac{\mu}{\mu_0}}\left\|\frac{s_N(\mu_0)}{s_N(\mu)}\right\|_\infty \leq \frac{1+\beta}{(1-\beta)^2}\sqrt{\frac{\mu_0}{\mu}}\left\|\frac{x_N(\mu)}{x_N(\mu_0)}\right\|_\infty \leq \frac{(1+\beta)n}{(1-\beta)^2}\sqrt{\frac{\mu_0}{\mu}} \tag{82}$$

Inequality (36) now follows by combining the estimates (79), (80), (81) and (82). ∎

Proof of Theorem 2.14. Recall that our goal is to prove that the arithmetic complexity of computing a TR direction during a TR-iteration is bounded by (41). It suffices to examine just the computation of the primal TR direction since the argument for the dual TR direction is analogous. We have seen in the proof of Lemma 2.10 that $\psi_p(\lambda)$ can be expressed as $\psi_p(\lambda) = \|(I+\lambda\tilde{H})^{-1}\tilde{g}\|/\sqrt{\mu}$, where $\tilde{H} \in \Re^{|B|\times|B|}$ and $\tilde{g} \in \Re^{|B|}$ can be computed in $\mathcal{O}(n^3)$ arithmetic operations. It is well-known that we can compute an orthogonal matrix Q such that the matrix $T \equiv Q^T\tilde{H}Q$ is tridiagonal in $\mathcal{O}(n^3)$ arithmetic operations. Moreover, using the fact that orthogonal matrices preserve vector lengths, we easily see that $\psi_p(\lambda) = \|(I+\lambda T)^{-1}Q^T\tilde{g}\|/\sqrt{\mu}$. Hence, for any fixed $\lambda > 0$, the fact that T is tridiagonal implies that $\psi_p(\lambda)$ can be computed in $\mathcal{O}(n)$ arithmetic operations. We have thus shown that the arithmetic complexity to compute a TR direction during a TR iteration is bounded by (41). ∎

6 Conclusion

In this paper, we have developed a predictor-corrector trust-region algorithm for linear programming whose iteration-complexity just depends on $\bar{\chi}_A^*$. The overall arithmetic complexity of the algorithm is not independent of b and/or c, due to work involved in the computation of the trust region steps. An interesting and challenging open question is whether the arithmetic complexity of the PC-TR algorithm, or a variant of it, has an arithmetic complexity that does not depend on b and c.

Appendix

The objective of this section is to provide a proof of Lemma 2.7.

First, we state a technical result whose proof is given in Lemma 4.4 of Monteiro and Tsuchiya [15].

Lemma 6.1 *Let $w = (x,y,s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ be given and assume that $\|xs - \nu e\| \leq \tau\nu$ for some constants $\tau \in (0,1)$ and $\nu > 0$. Then, $(1-\tau/\sqrt{n})\nu \leq \mu(w) \leq (1+\tau/\sqrt{n})\nu$ and $w \in \mathcal{N}(\tau/(1-\tau))$.*

We are now ready to prove Lemma 2.7.

Proof of Lemma 2.7: Define $v(\alpha) \equiv (x + \alpha\Delta x)(s + \alpha\Delta s)$ for all $\alpha \in \Re$. We claim that

$$\|v(\alpha) - (1-\alpha)\mu e\| \leq \frac{2\beta}{1+2\beta}(1-\alpha)\mu \quad \forall\ 0 \leq \alpha \leq 1 - \bar{\alpha} \tag{83}$$

where $\mu \equiv \mu(w)$,

$$\varepsilon_2(w) \equiv \max\{\|Rx_N\|, \|Rs_B\|\}, \qquad \bar{\alpha} \equiv \frac{\sqrt{1+\beta}+\gamma}{4\gamma}\varepsilon_2(w). \tag{84}$$

Using this claim, the result can be proved as follows. By Lemma 6.1 with $w = w + \alpha\Delta w$, $\nu = (1-\alpha)\mu$ and $\tau = 2\beta/(1+2\beta)$, we conclude from the claim that for any $0 \leq \alpha \leq 1 - \bar{\alpha}$, we have $w + \alpha\Delta w \in \mathcal{N}(2\beta)$ and

$$\mu(w + \alpha\Delta w) \leq \left(1 + \frac{2\beta}{\sqrt{n}(1+2\beta)}\right)(1-\alpha)\mu \leq 2(1-\alpha)\mu. \tag{85}$$

By the definition of α_τ, we then conclude that $\alpha_\tau \geq 1 - \bar{\alpha}$. Setting $\alpha = 1 - \bar{\alpha}$ in (85) and using the fact that $\alpha_\tau \geq 1 - \bar{\alpha}$ and $\mu(w + \alpha\Delta w)$ is a decreasing (affine) function of α, we obtain

$$\frac{\mu(w + \alpha_\tau\Delta w)}{\mu(w)} \leq \frac{\sqrt{1+\beta}+\gamma}{2\gamma}\varepsilon_2(w)$$

that is, the result holds.

In the remaining part of the proof, we show that (83) holds. It is easy to see that

$$\begin{aligned} v(\alpha) - (1-\alpha)\mu e &= (x + \alpha\Delta x)(s + \alpha\Delta s) - (1-\alpha)\mu e \\ &= (1-\alpha)(xs - \mu e) + \alpha h^1 + \alpha(1-\alpha)h^2 + \alpha^2 h^3, \end{aligned} \tag{86}$$

where h^1, h^2 and h^3 are vectors in $\Re^n$ defined as

$$\begin{pmatrix} h^1_B \\ h^1_N \end{pmatrix} \equiv \begin{pmatrix} x_B(s_B + \Delta s_B) \\ s_N(x_N + \Delta x_N) \end{pmatrix} = \mu \begin{pmatrix} w_B p_B \\ w_N p_N \end{pmatrix}, \tag{87}$$

$$\begin{pmatrix} h^2_B \\ h^2_N \end{pmatrix} \equiv \begin{pmatrix} s_B\Delta x_B \\ x_N\Delta s_N \end{pmatrix} = \mu \begin{pmatrix} w_B q_B \\ w_N q_N \end{pmatrix}, \tag{88}$$

$$\begin{pmatrix} h^3_B \\ h^3_N \end{pmatrix} \equiv \begin{pmatrix} \Delta x_B(s_B + \Delta s_B) \\ \Delta s_N(x_N + \Delta x_N) \end{pmatrix} = \mu \begin{pmatrix} p_B q_B \\ p_N q_N \end{pmatrix}. \tag{89}$$

Here, the vectors p, q and w appearing in the second alternative expressions for h^1, h^2 and h^3 are defined as

$$\begin{pmatrix} p_B \\ p_N \end{pmatrix} \equiv \begin{pmatrix} Rs_B(w) \\ Rx_N(w) \end{pmatrix}, \qquad \begin{pmatrix} q_B \\ q_N \end{pmatrix} \equiv \begin{pmatrix} \Delta_B\Delta x_B/\sqrt{\mu} \\ \Delta_N^{-1}\Delta s_N/\sqrt{\mu} \end{pmatrix}, \qquad w \equiv \frac{x^{1/2}s^{1/2}}{\sqrt{\mu}}.$$

Clearly, in view of (84), (18) and the fact that $w \in \mathcal{N}(\beta)$, we have

$$\|p\|_\infty \leq \varepsilon_2(w), \quad \|p\| \leq \sqrt{2}\varepsilon_2(w), \qquad \|q\| \leq \sqrt{2}\gamma, \quad \|w\|_\infty \leq \sqrt{1+\beta}, \quad \|w\| = \sqrt{n}. \tag{90}$$

Using (87), (88), (89), and (90), we obtain

$$\begin{aligned}
\|h^1\| &\leq \mu\|w\|_\infty \|p\| \leq \mu\sqrt{2(1+\beta)}\varepsilon_2(w), \\
\|h^2\| &\leq \mu\|w\|_\infty \|q\| \leq \mu\sqrt{2(1+\beta)}\gamma \\
\|h^3\| &\leq \mu\|p\|_\infty \|q\| \leq \mu\sqrt{2}\gamma\varepsilon_2(w).
\end{aligned}$$

Using (86), the triangle inequality for norms, the three estimates above and relations (84) and (17), we then obtain

$$\begin{aligned}
\|v(\alpha) - (1-\alpha)\mu e\| &\leq (1-\alpha)\|xs - \mu e\| + \alpha\|h^1\| + \alpha(1-\alpha)\|h^2\| + \alpha^2\|h^3\| \\
&\leq (1-\alpha)\left(\|xs - \mu e\| + \|h^2\|\right) + \|h^1\| + \|h^3\| \\
&\leq \left[(1-\alpha)\left(\beta + \sqrt{2(1+\beta)}\,\gamma\right) + \left(\sqrt{2(1+\beta)} + \sqrt{2}\,\gamma\right)\varepsilon_2(w)\right]\mu \\
&\leq \left[(1-\alpha)\left(\beta + \sqrt{2(1+\beta)}\,\gamma\right) + 4\sqrt{2}\gamma\bar{\alpha}\right]\mu \\
&\leq \left[\left(\beta + \sqrt{2(1+\beta)}\,\gamma\right) + 4\sqrt{2}\gamma\right](1-\alpha)\mu \\
&\leq \frac{2\beta}{1+2\beta}(1-\alpha)\mu,
\end{aligned}$$

for all $0 \leq \alpha \leq 1 - \bar{\alpha}$. ∎

References

[1] S. Barnett, *Matrices: Methods and Applications*, Oxford Univerisy Press, Oxford, 1990.

[2] I. I. Dikin and V. I. Zorkalcev, *Iterative Solution of Mathematical Programming Problems: Algorithms for the Method of Interior Points* (in Russian), Nauka, Novosibirsk, USSR, 1980.

[3] C. C. Gonzaga: Path-following methods for linear programming. *SIAM Review*, Vol. 34, pp. 167–224.

[4] C. C. Gonzaga and H. J. Lara: A note on properties of condition numbers. *Linear Algebra and its Applications*. Vol. 261 (1997), pp. 269–273.

[5] N. Karmarkar: A new polynomial-time algorithm for linear programming. *Combinatorica* Vol. 4 (1984), No. 4, pp.373–395.

[6] 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-Verlag, New York, 1989, pp. 29–47.

[7] M. Kojima, S. Mizuno and A. Yoshise: A polynomial-time algorithm for a class of linear complementarity problems. *Mathematical Programming*, Vol. 44 (1989), pp. 1–26.

[8] N. Megiddo: Pathways to the optimal set in linear programming. *Progress in mathematical programming* (Pacific Grove, CA, 1987), pp. 131–158, Springer, New York-Berlin, 1989.

[9] N. Megiddo, S. Mizuno and T. Tsuchiya: A modified layered-step interior-point algorithm for linear programming. *Mathematical Programming*, Vol. 82 (1998), pp. 339-355.

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

[11] S. Mizuno, N. Megiddo, and T. Tsuchiya: A linear programming instance with many crossover events. *Journal of Complexity* Vol. 12 (1996) pp. 474–479.

[12] R. D. C. Monteiro and I. Adler: Interior path-following primal-dual algorithms. Part I: Linear programming. *Mathematical Programming*, Vol. 44 (1989), pp. 27–41.

[13] R. D. C. Monteiro and I. Adler. Interior path-following primal-dual algorithms. Part II: Convex quadratic programming. *Mathematical Programming*, Vol. 44 (1989), pp. 43–66.

[14] R. D. C. Monteiro and T. Tsuchiya: Global convergence of the affine scaling algorithm for convex quadratic programming. *SIAM Journal on Optimization*, Vol. 8 (1998), No. 1, pp. 26–58.

[15] R. D. C. Monteiro and T. Tsuchiya: A variant of the Vavasis-Ye layered-step interior-point algorithm for linear programming. *SIAM Journal on Optimization*, Vol. 13 (2003), pp. 1054–1079.

[16] R. D. C. Monteiro and T. Tsuchiya: A new iteration-complexity bound for the MTY predictor-corrector algorithm. *SIAM Journal on Optimization*, Vol. 15 (2004), pp. 319–347.

[17] R. D. C. Monteiro and T. Tsuchiya: A strong bound on the integral of the central path curvature and its relationship with the iteration complexity of primal-dual path-following LP algorithms, working paper, School of ISyE, Georgia Tech, USA, September 2005 (accepted in Mathematical Programming).

[18] J. Renegar: A polynomial-time algorithm, based on Newton's method, for linear programming. *Mathematical Programming*, Vol. 40 (1988), No. 1, pp. 59–93.

[19] G. W. Stewart: On Scaled projections and pseudo-inverses, *Linear Algebra and its Applications*, Vol. 112 (1989) pp. 189–193.

[20] K. Tanabe: Centered Newton method for mathematical programming. *System Modeling and Optimization* (M. Iri and K. Yajima (eds.)), pp. 197-206, Springer-Verlag, Berlin, 1988.

[21] É. Tardos: A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research* Vol. 34, (1986) pp. 250–256.

[22] M. J. Todd: A Dantzig-Wolfe-like variant of Karmarkar's interior-point linear programming algorithm. *Operations Research*, Vol. 38 (1990) pp. 1006–1018.

[23] M. J. Todd, L. Tunçel and Y. Ye: Characterizations, bounds, and probabilistic analysis of two complexity measures for linear programming problems. *Mathematical Programming*, Vol. 90 (2001), No. 1, pp. 59–70.

[24] T. Tsuchiya: Global Convergence Property of the Affine Scaling Methods for Primal Degenerate Linear Programming Problems. *Mathematics of Operations Research*, Vol. 17, No. 3 (1992), pp. 527-557.

[25] L. Tunçel: Primal-dual symmetry and scale invariance of interior-point algorithms for convex optimization. *Mathematics of Operations Research*, Vol. 23 (1998), No. 3, pp. 708–718.

[26] L. Tunçel: Approximating the complexity measure of Vavasis-Ye algorithm is NP-hard. *Mathematical Programming*, Vol. 86 (1999), No. 1, pp. 219–223.

[27] L. Tunçel: On the condition numbers for polyhedra in Karmarkar's form. *Operations Research Letters*, Vol. 24 (1999), No. 4, pp. 149–155.

[28] R. J. Vanderbei and J. C. Lagarias: I. I. Dikin's convergence result for the affine-scaling algorithm, Vol. 17, *Contemporary Mathematics* Vol. 114 (1990) pp. 109–119.

[29] S. Vavasis and Y. Ye: A primal-dual accelerated interior-point method whose running time depends only on A. *Mathematical Programming*, Vol. 74 (1996), pp. 79–120.

[30] S. Vavasis and Y. Ye: A simplification to "a primal-dual interior-point method whose running time depends only on the constraint matrix". High performance optimization, 233–243, Appl. Optim., 33, Kluwer Acad. Publ., Dordrecht, 2000.

[31] S. Vavasis and Y. Ye: On the relationship between layered least squares and affine scaling steps. In *The Mathematics of Numerical Analysis, Lectures in Applied Mathematics, vol. 32* American Mathematical Society, Providence, RI, 1996, pp. 857–865.