McMaster University

Advanced Optimization Laboratory

Title:

An enhanced primal-simplex based Tardos' algorithm

Authors:

Shinji Mizuno, Noriyoshi Sukegawa, and Antoine Deza

AdvOL-Report No. 2015/3

August 2015, Hamilton, Ontario, Canada

An enhanced primal-simplex based Tardos' algorithm

Shinji Mizuno[a], Noriyoshi Sukegawa[b], and Antoine Deza[c]

[a]Graduate School of Decision Science and Technology, Tokyo Institute of Technology, Tokyo, Japan.

[b]Department of Information and System Engineering, Chuo University, Tokyo, Japan.

[c]Advanced Optimization Laboratory, Department of Computing and Software, McMaster University, Hamilton, Ontario, Canada.

Abstract

The authors recently proposed a simplex-based Tardos' algorithm which is strongly polynomial if the coefficient matrix is totally unimodular and the auxiliary problems are non-degenerate. Motivated by the algorithmic practically of such methods, we introduce a modification which circumvents the determination of the largest sub-determinant while keeping the same theoretical performance. Assuming the coefficient matrix is integer and non-degeneracy, the proposed algorithm is polynomial in the dimension of the input data and the maximum absolute value of a sub-determinant of the coefficient matrix.

Keywords: Tardos' algorithm; simplex method; strong polynomiality

1 Introduction, main result, and related works

1.1 Introduction

Linear optimization deals with the minimization problem : $\min\{\, \boldsymbol{c}^\top \boldsymbol{x} \mid A\boldsymbol{x} = \boldsymbol{b},\, \boldsymbol{x} \geq \boldsymbol{0}\,\}$ where the coefficient matrix $A \in \mathbb{R}^{m\times n}$, the right hand side

$\boldsymbol{b} \in \mathbb{R}^m$, and the objective vector $\boldsymbol{c} \in \mathbb{R}^n$ are given data. The question of whether a linear optimization problem can be solved in strongly polynomial time – that is, the existence of an algorithm independent from the input data length and polynomial in m and n – is listed by Smale [21] as one of the top mathematical problems for the XXI century. The celebrated Tardos' algorithm [23, 24] for linear optimization is polynomial in m, n, and the size L_A of A. We recall that the size of $A = [a_{ij}]$ is defined, for A integer, as $L_A = \sum_{i,j} \log(|a_{ij}| + 1)$. Thus, Tardos' algorithm is strongly polynomial if the size of A is polynomial in m and n which is the case for combinatorial problems such as minimum cost flow, bipartite matching, multicommodity flow, and vertex packing in chordal graphs.

A key element of Tardos' algorithm is to identify the coordinates equal to zero at optimality by solving several auxiliary dual problems via an ellipsoid or interior-point method. Considering only the primal problem, Orlin [17] proposed a modification of Tardos' algorithm which specifically identifies the coordinates strictly positive at optimality. Mizuno [15] modified Tardos' algorithm by using a dual simplex method to solve the auxiliary problems. Assuming that A is integer and non-degeneracy, Mizuno's algorithm is polynomial in m, n, and Δ_A. We recall that Δ_A denotes the largest absolute value of a sub-determinant of A and that the non-degeneracy assumption holds if the basic variables are strictly positive for every basic feasible solution of the auxiliary problems. In particular, Mizuno's algorithm is strongly polynomial if A is totally unimodular and non-degeneracy holds. Note that the complexity analysis uses Kitahara and Mizuno's bounds [10, 11] which depend on the values of the entries rather than on the data length. Thus, the complexity of Mizuno's algorithm depends on Δ_A rather than on L_A. Combining Orlin's and Mizuno's approaches, the authors introduced a primal-simplex based Tardos' algorithm with the same theoretical complexity as Mizuno's algorithm, see [16]. Tardos' algorithm and the mentioned modifications by Orlin, Mizuno, and Mizuno et al. are of rather theoretical interest. In particular, the determination of Δ_A might be challenging as the naive upper bound of $m! A_{\max}^m$ is typically impractically large; we recall that $A_{\max} = \max |a_{i,j}|$. In addition, the coefficients of the auxiliary problems might be impractically large too. For instance, the size of the coefficients in Orlin's algorithm can be m times larger than those in Tardos' or Mizuno's algorithm. The complexity analysis of Mizuno et al. algorithm requires total unimodularity for A. We conclude this section by recalling the min-cost flow problems introduced by Zadeh [29] for which the network simplex method requires an exponential number of steps, and thus illustrate the gap between the excellent practical performances of the simplex method and its theoretical properties.

1.2 Main result

We proposed an enhanced primal-simplex based Tardos' algorithm circumventing the determination of Δ_A while slightly strengthening the complexity. The enhanced algorithm is obtained by modifying the auxiliary problem used in Mizuno et al. algorithm. Assuming A is integer and non-degeneracy, the enhanced Mizuno et al. algorithm is polynomial in m, n, and Δ_A. Thus the strong polynomiallity holds for a slightly larger class than totally unimodular matrices, e.g. a coefficient matrix resulting from the addition to a totally unimodular matrix of a fixed number of rows (constraints) with entries polynomially bounded in m and n. The determination of Δ_A is circumvented via a simple search procedure and the practically of the algorithm improves as the coefficients of the auxiliary problems are typically substantially smaller in the enhanced Mizuno et al. algorithm. After recalling some related works in Section 1, the pre-processing and reformulations into auxiliary problems are presented in Section 2. Sections 3 and 4 outline the proposed algorithm: the main procedure – which requires the determination of Δ_A – and, then, the enhanced algorithm which circumvents the determination of Δ_A. The correctness and the complexity analysis of the algorithm are proven in Sections 5 and 6.

1.3 Related works

We recall a few results dealing with the geometry, combinatorics and computational aspects of linear optimization. Finding a good bound on the maximal diameter $\Delta(n,m)$ of the vertex-edge graph of a polytope in terms of its dimension n and the number of its facets m is one of the basic open questions in polytope theory. Although some bounds are known, the behaviour of the function $\Delta(n,m)$ is largely unknown. The Hirsch conjecture, formulated in 1957 and reported in [5], states that $\Delta(n,m)$ is linear in m and n: $\Delta(n,m) \leq m-n$. The conjecture was recently disproved by Santos [19]. However, the asymptotic behaviour of $\Delta(n,m)$ is not well understood: the best upper bounds — due to Kalai and Kleitman [8], Todd [26], and Sukegawa and Kitahara [22] — are quasi-polynomial. The behaviour of $\Delta(n,m)$ is historically closely connected with the theory of the simplex method as $\Delta(n,m)$

is a lower bound for the worst complexity of simplex methods. Bonifas et al. [2] showed that the diameter is an $O(n^4\Delta_A^2\log(n\Delta_A))$ extending the previous bound of $O(m^{16}n^3(\log mn)^3)$ by Dyer and Frieze [6] for totally unimodular instances. Dadush and Hähnle [4] used another parameter associated to the coefficient matrix A, called the *curvature* δ_A, to analyze the behaviour of the shadow simplex method. They showed that the expected number of pivots of the shadow simplex method is an $O(\frac{n^3}{\delta_A}\log\frac{n}{\delta_A})$. Both δ_A and Δ_A can be regarded as a measure of how well-conditioned A is. Note that $1/\delta_A \leq n\Delta_A^2$. Vavasis and Ye [27] proposed a primal-dual path-following interior-point algorithm with an $O(n^{3.5}\log(n\bar{\chi}_A))$ iteration complexity bound where $\bar{\chi}_A$ can be regarded as condition number associated with A. Megiddo et al. [13] proposed a modification that circumvents the determination of $\bar{\chi}_A$ to enhanced the implementability. Another variant of Vavasis and Ye algorithm was proposed by Monteiro and Tsuchiya [14]

In a similar fashion, we circumvent the determination of Δ_A while Megiddo et al. circumvent the determination of $\bar{\chi}_A$. While we assume non-degeneracy, Dadush and Hähnle algorithm analysis is non-deterministic. In practice, degenerate pivots are typically rare and our algorithm may exhibit reasonable performance under moderate degeneracy. The proposed algorithm may visit infeasible points as, for example, Anstreicher and Terlaky monotonic build-up simplex and Paparrizos exterior point simplex, or Fukuda and Terlaky criss-cross method, see [1, 7, 18, 25] and references therein.

2 Pre-processing and reformulation via auxiliary problems

We consider the following linear optimization formulation:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}^\top \boldsymbol{x} \\ \text{s.t.} & A\boldsymbol{x} = \boldsymbol{b}, \\ & \boldsymbol{x} \geq \boldsymbol{0} \end{array} \tag{1}$$

where $A \in \mathbb{R}^{m\times n}$, $\boldsymbol{b} \in \mathbb{R}^m$, and $\boldsymbol{c} \in \mathbb{R}^n$ are given. In addition, A is assumed to have full row rank m and be integer. We recall that $A_{\max}$, respectively Δ_A, denotes the largest absolute value of an entry, respectively a sub-determinant, of A. Note that $A_{\max} \leq \Delta_A$ and $\Delta_A \leq m!A_{\max}^m$.

2.1 Pre-processing and problem reformulations: reduction and scaling

The optimal solution of (1), if any, is assumed without loss of generality to be unique. Otherwise $\boldsymbol{c}$ could be perturbed by $(\epsilon, \epsilon^2, \ldots, \epsilon^n)$ for a sufficiently small $\epsilon > 0$. Such perturbations have no impact on the analysis of the proposed algorithm as it is based on the results of Kitahara and Mizuno [10, 11] which are independent of $\boldsymbol{c}$. From an algorithmic viewpoint, perturbations are not required as one can instead consider a lexicographical order if a tie occurs when choosing the entering variable via the simplex method with Dantzig's rule.

Let $K^* \subseteq N = \{1, 2, \ldots, n\}$ be the set of indices i such that $x_i^* > 0$ for the optimal solution $\boldsymbol{x}^*$ of (1) – which is assumed to exist. The proposed algorithm inductively builds a subset $\bar{K} \subseteq K^*$ through solving an auxiliary problem. If $\bar{K} = K^*$ we obtained the optimal solution. Otherwise, we obtain a smaller, yet equivalent, problem by deleting the variables corresponding to $\bar{K}$. We first observe that (1) is equivalent to:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}_{\bar{K}}^\top \boldsymbol{x}_{\bar{K}} + \boldsymbol{c}_K^\top \boldsymbol{x}_K \\ \text{s.t.} & A_{\bar{K}} \boldsymbol{x}_{\bar{K}} + A_K \boldsymbol{x}_K = \boldsymbol{b}, \\ & \boldsymbol{x}_{\bar{K}} \text{ free}, \ \boldsymbol{x}_K \geq \boldsymbol{0} \end{array} \tag{2}$$

where $K = N \setminus \bar{K}$ and $\bar{K}$ is an arbitrary subset of K^*.

The *reduced problem* (3) is obtained by eliminating free variables in $\boldsymbol{x}_{\bar{K}}$ as follows. Let G be a $m \times m$ sub-matrix of A such that the first $|\bar{K}|$ columns form $A_{\bar{K}}$, and $H = G^{-1}$. The Gaussian elimination for $A\boldsymbol{x} = \boldsymbol{b}$ of the variables x_i for $i \in \bar{K}$ is performed via $HA\boldsymbol{x} = H\boldsymbol{b}$. Let H_1 consist of the first $|\bar{K}|$ rows of H, and H_2 denote the remainder. The equality $HA\boldsymbol{x} = H\boldsymbol{b}$ yields:

$$\begin{array}{l} H_1 A_{\bar{K}} \boldsymbol{x}_{\bar{K}} + H_1 A_K \boldsymbol{x}_K = H_1 \boldsymbol{b}, \\ H_2 A_{\bar{K}} \boldsymbol{x}_{\bar{K}} + H_2 A_K \boldsymbol{x}_K = H_2 \boldsymbol{b} \end{array}$$

where $H_1 A_{\bar{K}} = I$ and $H_2 A_{\bar{K}} = 0$ by the definition of H. Hence, the reduced problem is:

$$\begin{array}{ll} \text{minimize} & (\boldsymbol{c}_K^\top - \boldsymbol{c}_{\bar{K}}^\top H_1 A_K) \boldsymbol{x}_K + \boldsymbol{c}_{\bar{K}}^\top H_1 \boldsymbol{b} \\ \text{s.t.} & H_2 A_K \boldsymbol{x}_K = H_2 \boldsymbol{b}, \\ & \boldsymbol{x}_K \geq \boldsymbol{0}. \end{array} \tag{3}$$

Let $\boldsymbol{x}_K$ be an optimal solution of (3). Then $\boldsymbol{x} = (\boldsymbol{x}_{\bar{K}}, \boldsymbol{x}_K)$ with $\boldsymbol{x}_{\bar{K}} =$

$H_1\boldsymbol{b} - H_1 A_K \boldsymbol{x}_K$ is an optimal solution of (2). Setting $A' = H_2 A_K$, $\boldsymbol{b}' = H_2\boldsymbol{b}$, $\boldsymbol{c}' = \boldsymbol{c}_K - A_K^\top H_1^\top \boldsymbol{c}_{\bar{K}}$, and $\boldsymbol{x}' = \boldsymbol{x}_K$, the reduced problem (3) is reformulated as the standard linear optimization problem (4) where the constant in the objective function is omitted:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^\top \boldsymbol{x}' \\ \text{s.t.} & A'\boldsymbol{x}' = \boldsymbol{b}', \\ & \boldsymbol{x}' \geq \boldsymbol{0}. \end{array} \tag{4}$$

Observe that HA has full row rank and thus A' too. Problems (1) and (4) are equivalent: If L^* is an optimal basis of (4), then $\bar{K} \cup L^*$ is an optimal basis of (1). Let $m' = m - |\bar{K}|$, respectively $n' = n - |\bar{K}|$, denote the number of equality constraints, respectively variables, of (4). To obtain the desired auxiliary problem, we rewrite (4) and get a simplex *tableau* with respect to some basis $L \subseteq K$ of A' and set $\bar{L} = K \setminus L$ as follows:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^\top \boldsymbol{x}' \\ \text{s.t.} & \boldsymbol{x}'_L + (A'_L)^{-1} A'_{\bar{L}} \boldsymbol{x}'_{\bar{L}} = (A'_L)^{-1}\boldsymbol{b}', \\ & \boldsymbol{x}' \geq \boldsymbol{0}. \end{array} \tag{5}$$

Considering a scaling factor $\kappa = \|A'^\top (A'A'^\top)^{-1}\boldsymbol{b}'\|_2 / (mnn' A_{\max}\Delta + m')$ for some Δ, yields the following scaled problem:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^\top \boldsymbol{x}' \\ \text{s.t.} & \boldsymbol{x}'_L + (A'_L)^{-1} A'_{\bar{L}} \boldsymbol{x}'_{\bar{L}} = (A'_L)^{-1}\boldsymbol{b}'/\kappa, \\ & \boldsymbol{x}' \geq \boldsymbol{0}. \end{array} \tag{6}$$

The scaling factor κ is always strictly positive in our algorithm and, thus, a basis is optimal for (6) if and only if it is optimal for (4) and for the simplex tableau (5). Note that since A' has full row rank, $A'A'^\top$ is positive definite and thus $(A'A'^\top)^{-1}$ is well defined.

2.2 Auxiliary problem

The auxiliary problem is obtained from (6) by rounding up the right hand side vector where $\lceil \boldsymbol{a} \rceil$ denotes a vector whose i-th coordinate is the smallest integer not less than the i-th coordinate of $\boldsymbol{a}$:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c'}^\top \boldsymbol{x'} \\ \text{s.t.} & \boldsymbol{x'}_L + (A'_L)^{-1} A'_{\bar{L}} \boldsymbol{x'}_{\bar{L}} = \lceil (A'_L)^{-1} \boldsymbol{b'}/\kappa \rceil, \\ & \boldsymbol{x'} \geq \boldsymbol{0}, \end{array} \tag{7}$$

Note that a feasible basis of (6) is feasible for (7) as (7) is a relaxation of (6). The key feature of (7) is that the coordinates of the right hand side vectors are small integers, see Lemma 6. Thus, (7) can be solved efficiently by the simplex method, yielding the strong polynomiality analysis.

3 Main procedure

The main procedure of the algorithm is frequently called to solve (1) and involves, as subroutine, a two-phase simplex method to solve (7), see Section 3.1. While $\Delta \geq \Delta_A$ guaranties that (1) is solved by the main procedure, (1) may be solved even if $\Delta < \Delta_A$. The enhanced primal-simplex based Tardos' algorithm is presented in Section 4.

3.1 Two-phase simplex method TwoS((1); F, $\bar{K}^*$)

Input: Problem (1).

Output: F which is either INFEASBILE OR UNBOUNDED or FEASIBLE AND FINITE, and an optimal basis $\bar{K}^*$ of (1) if F = FEASIBLE AND FINITE.

Phase I: Solve the following auxiliary problem via the simplex method with Dantzig's rule: a non-negative slack variable is added for each constraint and the sum of the slacks is minimized. The optimal value σ of this auxiliary problem is zero if and only if (1) is feasible. Output INFEASBILE OR UNBOUNDED for F if $\sigma > 0$. If $\sigma = 0$, the associated optimal basis yields a feasible basis $\bar{K}^0$ for (1) used to initialize Phase II.

Phase II: Starting from $\bar{K}^0$, solve (1) via the simplex method with Dantzig's rule. Output INFEASBILE OR UNBOUNDED for F if (1) is unbounded; otherwise output FEASIBLE AND FINITE and an optimal basis $\bar{K}^*$ for (1) is obtained.

3.2 Main procedure Proc((1), Δ; F, $\bar{K}^*$)

Input: Problem (1) and $\Delta > 0$.

Output: F which is either INFEASBILE OR UNBOUNDED, FEASIBLE AND FINITE, DEGENERATE, or UNIDENTIFIED, and an optimal basis $\bar{K}^*$ for (1) if F=FEASIBLE AND FINITE.

Initialization $\bar{K} := \emptyset$.

Step 1: If $\bar{K} \neq \emptyset$, remove the variables x_i in (1) for all $i \in \bar{K}$ to obtain the reduced problem (4). If $\bar{K} = \emptyset$, set $A' = A$, $\boldsymbol{b}' = \boldsymbol{b}$, $\boldsymbol{c}' = \boldsymbol{c}$, and $\boldsymbol{x}' = \boldsymbol{x}$. Go to Step 2.

Step 2: Consider the simplex tableau (5) associated to a basis L of the reduced problem (3). If $(A'_L)^{-1}\boldsymbol{b}' = \boldsymbol{0}$, output F=DEGENERATE. Otherwise, determine $\kappa = \|A'^{\top}(A'A'^{\top})^{-1}\boldsymbol{b}'\|_2/(m'nn'A_{\max}\Delta + m')$ and obtain the auxiliary problem (7). Go to Step 3.

Step 3: Perform TwoS((7); F, L^*). Output F if F=INFEASBILE OR UNBOUNDED. Otherwise, let $\boldsymbol{x}''$ be the optimal solution of (7) associated to the optimal basis L^*. Output F=FEASIBLE AND FINITE and set $\bar{K}^* = \bar{K} \cup L^*$ if $\bar{K} \cup L^*$ is an optimal basis for (1). Otherwise, go to Step 4.

Step 4: Update $\bar{K} := \bar{K} \cup J$ with $J = \{\, i \in L^* \mid x''_i \geq m'nA_{\max}\Delta \}$. Output F=UNIDENTIFIED if $|\bar{K}| = m$. Otherwise, go to Step 1.

Theorem 1 shows that Proc((1), Δ; F, $\bar{K}^*$) solves (1) if $\Delta \geq \Delta_A$, and thus extends Mizuno et al. [16] primal-simplex based Tardos' algorithm.

3.2.1 Annotations for Proc((1), Δ; F, $\bar{K}^*$)

We outline the stopping criteria before providing additional details about the main procedure.

(*i*) If $(A'_L)^{-1}\boldsymbol{b}' = \boldsymbol{0}$, the original problem (1) is degenerate, and (3) is either unbounded or admits zero as an optimal solution.

(*ii*) If (7) is infeasible then (1) is infeasible, and if (7) is unbounded then (1) is unbounded or infeasible. Indeed, (7) being a relaxation of (6) for any Δ, the infeasibility of (7) implies the infeasibilty of (6) and of (3), and thus of (1). If (7) is unbounded, then (6) is unbounded or infeasible, and thus (1) is unbounded or infeasible.

Lemma 1 shows that the set J defined in Step 4 satisfies $J \neq \emptyset$, and thus the main procedure is finite as at most m auxiliary problems are solved. Corollary 1 shows that, if $\Delta \geq \Delta_A$, $x_i^* > 0$ for $i \in J$ with $\boldsymbol{x}^*$ the optimal solution of (1). Thus, Corollary 1 shows that $J \subset K^*$ and validates Step 4, and thus, the correctness of the main procedure for $\Delta \geq \Delta_A$. As the main procedure is guaranteed to solve (1) only if $\Delta \geq \Delta_A$, F is set UNIDENTIFIED if $\Delta < \Delta_A$. However, the correct solution may be obtained even if $\Delta < \Delta_A$. For example, if $\bar{K} = \emptyset$; i.e. no reduction is performed in Step 4, and an optimal basis for (7) turns out to be feasible for (1) in Step 3, then this basis is optimal for (1) as (7) is a relaxation of (1).

3.2.2 Warm start for Proc((1), Δ; F, $\bar{K}^*$)

Although the main procedure builds the simplex tableau (5) and the reduced problem (3) from scratch at each iteration, it is essentially for clarity of the exposition. In practice, one can observe that $L^* \setminus J$ can serve as the basis L for (3) at the next iteration, thus enabling a warm start – as already noticed in Mizuno et al. algorithm [16].

4 An enhanced primal-simplex based Tardos' algorithm

The proposed algorithm circumvents the determination of Δ_A via a simple search procedure in the following algorithm Alg((1), Δ^0, λ; F, $\bar{K}^*$) where, typically, one can use $\Delta^0 = 1$ and $\lambda = mA_{\max}$. Assuming non-degeneracy and Δ_A being polynomially bounded in m and n, the proposed algorithm is strongly polynomial – as shown in Theorem 2.

Input: Problem (1), $\Delta^0 > 0$, and $\lambda > 1$.

Output: F which is either INFEASBILE OR UNBOUNDED, DEGENERATE, or FEASIBLE AND FINITE and an optimal basis $\bar{K}^*$ for (1) if F=FEASIBLE AND FINITE.

Initialization $\Delta := \Delta^0$.

Step 1: Perform Proc((1), Δ; F, $\bar{K}^*$). Output F if F=INFEASBILE OR UNBOUNDED or F=DEGENERATE. Output F and $\bar{K}^*$ if F=FEASBILE AND FINITE. Otherwise, go to Step 2.

Step 2: Update $\Delta := \lambda\Delta$. Go to Step 1.

Theorem 1. *Since $\Delta \geq \Delta_A$ guaranties that the original problem* (1) *is solved by the main procedure* Proc*((1), Δ; F, $\bar{K}^*$), the enhanced primal-simplex based Tardos' algorithm* Alg*((1), Δ^0, λ; F, $\bar{K}^*$) solves (1).*

Theorem 1 is a consequence of (i) Δ eventually satisfies $\Delta \geq m!A_{\max}^m \geq \Delta_A$, and (ii) $J \subset K^*$ for $\Delta \geq \Delta_A$ as shown in Corollary 1; that is, Step 4 is valid.

Theorem 2. Alg*((1), $\Delta^0 = 1, \lambda = mA_{\max}$; F, $\bar{K}^*$) performs* Proc*((1), Δ; F, $\bar{K}^*$) at most $m+1$ times.* Proc*((1), Δ; F, $\bar{K}^*$) performs* TwoS*((7); F, $\bar{K}^*$) at most m times. If all the auxiliary problems are non-degenerate, the number of arithmetic operations used by* Alg*((1), $\Delta^0 = 1$, $\lambda = mA_{\max}$; F, $\bar{K}^*$) to solve* (1) *is polynomial in m, n, and Δ_A.*

The first statement of Theorem 2 is implied by the stopping criterion $\Delta \geq m!A_{\max}^m$ and the setting $\Delta^0 = 1$ and $\lambda = mA_{\max}$. As mentioned in Section 3.2.1, the second statement of Theorem 2 is implied by Lemma 1. Thus, to complete the proof of Theorem 2 one has to show that TwoS((7); F, $\bar{K}^*$) is polynomial in m, n, and Δ_A – as proved in Section 6.

Instances of coefficient matrices with Δ_A polynomial in m and n include the one associated to capacitated network flow problems with additional linear constraints considered by Chen and Saigal [3]. The coefficient matrix they consider consists in the incidence matrix of a directed network, and thus totally unimodilar, to which a fixed number of arbitrary linear constraints on arc flow are added – assuming the entries are polynomial in m and n.

5 Proof of Theorem 1

Lemma 1 implies that, for any Δ, Proc((1), Δ; F, $\bar{K}^*$) performs TwoS((7); F, $\bar{K}^*$) at most m times.

Lemma 1. *For any $\Delta > 0$, a basic solution $\boldsymbol{x}''$ of the auxiliary problem* (7) *satisfies $\|\boldsymbol{x}''\|_\infty \geq m'nA_{\max}\Delta$. Hence, the set J defined in Step 4 satisfies $J \neq \emptyset$.*

Proof. Let $\boldsymbol{x}''$ be a solution of (7). We have $A'\boldsymbol{x}'' = A'_L\lceil(A'_L)^{-1}\boldsymbol{b}'/k\rceil$, $A'A'^\top$ is positive definite, and, for any $\boldsymbol{g}$, $A'^T(A'A'^T)^{-1}\boldsymbol{g}$ is the minimal l_2-norm point satisfying $A'\boldsymbol{x}' = \boldsymbol{g}$. Thus,

$$\begin{array}{rcl} \|\boldsymbol{x}''\|_2 & \geq & \|A'^{\top}(A'A'^{\top})^{-1}A'_L\lceil (A'_L)^{-1}\boldsymbol{b}'/\kappa\rceil\|_2 \\ & \geq & \|A'^{\top}(A'A'^{\top})^{-1}\boldsymbol{b}'/\kappa\|_2 - \|A'^{\top}(A'A'^{\top})^{-1}A'_L\boldsymbol{d}\|_2 \\ & = & (m'nn'A_{\max}\Delta + m') - \|A'^{\top}(A'A'^{\top})^{-1}A'\left(\begin{array}{c}\boldsymbol{d}\\ \boldsymbol{0}_{\bar{L}}\end{array}\right)\|_2 \\ & \geq & (m'nn'A_{\max}\Delta + m') - \|\left(\begin{array}{c}\boldsymbol{d}\\ \boldsymbol{0}_{\bar{L}}\end{array}\right)\|_2 \\ & = & m'nn'A_{\max}\Delta + m' - \|\boldsymbol{d}\|_2 \end{array}$$

where $\kappa = \|A'^{\top}(A'A'^{\top})^{-1}\boldsymbol{b}'\|_2/(m'nn'A_{\max}\Delta + m')$ and $\boldsymbol{d} = \lceil (A'_L)^{-1}\boldsymbol{b}'/\kappa\rceil - (A'_L)^{-1}\boldsymbol{b}'/\kappa$. Since $\|\boldsymbol{d}\|_\infty < 1$ and $\|\boldsymbol{d}\|_2 \leq m'\|\boldsymbol{d}\|_\infty$, we obtain: $\|\boldsymbol{x}''\|_\infty \geq \|\boldsymbol{x}''\|_2/n' > (m'nn'A_{\max}\Delta + m' - m')/n' = m'nA_{\max}\Delta$. □

Applying a key result of Schrijver, recalled in Lemma 2, to (6) and (7) yields Lemma 3 and Corollary 1 guarantying $J \subset K^*$, i.e. Step 4 is valid, for $\Delta \geq \Delta_A$.

Lemma 2 ([20], Theorem 10.5)**.** *Let A be an $m\times n$-matrix, and let Δ^* be such that for each nonsingular submatrix B of A all entries of B^{-1} are at most Δ^* in absolute value. Let $\boldsymbol{c}$ be a column n-vector, and let $\boldsymbol{b}''$ and $\boldsymbol{b}^*$ be column m-vectors such that $P'' : \max\{\, \boldsymbol{c}^{\top}\boldsymbol{x} \mid A\boldsymbol{x} \leq \boldsymbol{b}'' \,\}$ and $P^* : \max\{\, \boldsymbol{c}^{\top}\boldsymbol{x} \mid A\boldsymbol{x} \leq \boldsymbol{b}^* \,\}$ are finite. Then, for each optimal solution $\boldsymbol{x}''$ of P'', there exists an optimal solution $\boldsymbol{x}^*$ of P^* with $\|\boldsymbol{x}'' - \boldsymbol{x}^*\|_\infty \leq n\Delta^*\|\boldsymbol{b}'' - \boldsymbol{b}^*\|_\infty$.*

Lemma 3. *Assume that the scaled problem* (6) *and the auxiliary problem* (7) *are both feasible and finite. Then, for an optimal solution $\boldsymbol{x}''$ of (7), there exists an optimal solution $\boldsymbol{x}^*$ of* (6) *such that $\|\boldsymbol{x}'' - \boldsymbol{x}^*\|_\infty \leq n\Delta_A\|A_L\boldsymbol{d}\|_\infty$ with $\boldsymbol{d} = \lceil (A'_L)^{-1}\boldsymbol{b}'/\kappa\rceil - (A'_L)^{-1}\boldsymbol{b}'/\kappa$.*

Proof. Let $\boldsymbol{x}''$ be an optimal solution of (7). Then, $\tilde{\boldsymbol{x}}'' = (\tilde{\boldsymbol{x}}''_{\bar{K}}, \tilde{\boldsymbol{x}}''_K)$, with $\tilde{\boldsymbol{x}}''_K = \boldsymbol{x}''$ and $\tilde{\boldsymbol{x}}''_{\bar{K}} = H_1(\boldsymbol{b}/\kappa + A_L\boldsymbol{d}) - H_1A_K\boldsymbol{x}''$, is an optimal solution of:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^{\top}\boldsymbol{x}_K \\ \text{s.t.} & \boldsymbol{x}_{\bar{K}} + H_1A_K\boldsymbol{x}_K = H_1(\boldsymbol{b}/\kappa + A_L\boldsymbol{d}), \\ & H_2A_K\boldsymbol{x}_K = H_2(\boldsymbol{b}/\kappa + A_L\boldsymbol{d}), \\ & \boldsymbol{x}_K \geq \boldsymbol{0}. \end{array}$$

Multiplying both sides of the equalities from the left by $G = H^{-1}$, and recalling the definitions of H_1 and H_2 given in Section 2.1, yields:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^\top \boldsymbol{x}_K \\ \text{s.t.} & A_{\bar{K}}\boldsymbol{x}_{\bar{K}} + A_K\boldsymbol{x}_K = \boldsymbol{b}/\kappa + A_L\boldsymbol{d}, \\ & \boldsymbol{x}_K \geq \boldsymbol{0}. \end{array}$$

By Lemma 2, there exists an optimal solution $\tilde{\boldsymbol{x}}^* = (\tilde{\boldsymbol{x}}^*_{\bar{K}}, \tilde{\boldsymbol{x}}^*_K)$ of:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{c}'^\top \boldsymbol{x}_K \\ \text{s.t.} & A_{\bar{K}}\boldsymbol{x}_{\bar{K}} + A_K\boldsymbol{x}_K = \boldsymbol{b}/\kappa, \\ & \boldsymbol{x}_K \geq \boldsymbol{0} \end{array}$$

such that $\|\tilde{\boldsymbol{x}}'' - \tilde{\boldsymbol{x}}^*\|_\infty \leq n\Delta_A\|(\boldsymbol{b}/\kappa + A_L\boldsymbol{d}) - \boldsymbol{b}/\kappa\|_\infty$, and thus $\|\tilde{\boldsymbol{x}}'' - \tilde{\boldsymbol{x}}^*\|_\infty \leq n\Delta_A\|A_L\boldsymbol{d}\|_\infty$. In addition, $\|\boldsymbol{x}'' - \tilde{\boldsymbol{x}}^*_K\|_\infty \leq \|\tilde{\boldsymbol{x}}'' - \tilde{\boldsymbol{x}}^*\|_\infty$ since $\boldsymbol{x}''$ and $\tilde{\boldsymbol{x}}^*_K$ are sub-vectors of, respectively, $\tilde{\boldsymbol{x}}''$ and $\tilde{\boldsymbol{x}}^*$. Note that $\tilde{\boldsymbol{x}}^*_K$ is an optimal solution of the scaled problem (6), and equal to $\boldsymbol{x}^*$. □

Corollary 1. *Lemma 3 implies* $\|\boldsymbol{x}'' - \boldsymbol{x}^*_K\|_\infty \leq n\Delta_A\|A_L\boldsymbol{d}\|_\infty < m'nA_{\max}\Delta_A$*. Hence, for* $i \in J$ *and* $\Delta \geq \Delta_A$*,* $x^*_i > 0$ *with* $\boldsymbol{x}^*$ *the optimal solution of* (1)*. Note that* x_i *is an optimal basic variable of both* (6) *and* (1) *– recall that* (1) *has a unique optimal solution.*

6 Proof of Theorem 2

As mentioned in Section 4, we need to show that TwoS((7); F, $\bar{K}^*$) is polynomial in m, n, and Δ_A which is achieved via the following result of Kitahara and Mizuno and the technical Lemma 5.

Lemma 4 ([11], Corollary 3)**.** *If the problem is nondegenerate, the simplex method with the most negative pivoting rule, i.e. Dantzig's rule, or the best improvement pivoting rule finds an optimal solution in at most* $n\lceil m\frac{\gamma}{\delta}\log(m\frac{\gamma}{\delta})\rceil$ *iterations where* m *is the number of constraints,* n *is the number of variables, and* δ *and* γ *are, respectively, the minimum and the maximum values of all the positive elements of the primal basic feasible solutions.*

Lemma 5. *Let* L *be a basis of* A*. Then, each coordinate of* $A_L^{-1}(A, I)$ *is a rational number whose denominator is* $\det A_L$ *and the absolute value of the numerator is bounded above by* Δ_A*.*

Proof. For $j = 1, 2, \ldots, m+n$, let $\boldsymbol{y}_j$ be j-th column vector of $A_L^{-1}(A, I)$. Then, $A_L \boldsymbol{y}_j = \boldsymbol{a}_j$ where $\boldsymbol{a}_j$ is j-th column vector of (A, I). By Cramer's rule, the i-th coordinate of $\boldsymbol{y}_j$ is $y_{ji} = \det A_L(i,j)/\det A_L$ with $A_L(i,j)$ being the matrix where the i-th column vector of A_L is replaced by $\boldsymbol{a}_j$. □

In order to apply Lemma 4, the quantities γ and δ associated the auxiliary problem (7) are estimated in Lemma 6 and yields $\gamma/\delta \leq m^2(m'nn'A_{\max}\Delta + m')\Delta_A^3 + m\Delta_A^2$. This bound for γ/δ combined with Lemma 4 completes the proof of Theorem 2.

Lemma 6. *Each positive element of a basic feasible solution* $\boldsymbol{x}''$ *of* (7) *is bounded above by* $m^2(m'nn'A_{\max}\Delta + m')\Delta_A^2 + m\Delta_A$ *and below by* $1/\Delta_A$.

Proof. Let $\boldsymbol{x}''$ be a basic feasible solution of (7). Then, $\tilde{\boldsymbol{x}} = (\tilde{\boldsymbol{x}}_{\bar{K}}, \tilde{\boldsymbol{x}}_K)$, with $\tilde{\boldsymbol{x}}_K = \boldsymbol{x}''$ and $\tilde{\boldsymbol{x}}_{\bar{K}} = H_1 A_L \boldsymbol{f} - H_1 A_K \boldsymbol{x}''$, is a basic solution of:

$$\begin{aligned} &\boldsymbol{x}_{\bar{K}} + H_1 A_K \boldsymbol{x}_K = H_1 A_L \boldsymbol{f}, \\ &H_2 A_K \boldsymbol{x}_K = H_2 A_L \boldsymbol{f} \end{aligned}$$

where $\boldsymbol{f} = \lceil (A'_L)^{-1}\boldsymbol{b}'/\kappa \rceil$. Multiplying both sides of the equalities by $G = H^{-1}$ from the left yields $A\boldsymbol{x} = A_L\boldsymbol{f}$. Since $\tilde{\boldsymbol{x}}$ is a basic feasible solution of $A\boldsymbol{x} = A_L\boldsymbol{f}$, any positive coordinate $\tilde{x}_i$ of $\tilde{\boldsymbol{x}}$ is a rational number whose denominator is equal to the determinant of the basis matrix, see Lemma 5, and numerator is bounded below by 1 by the integrality of $A_L\boldsymbol{f}$. Hence the denominator of the coordinate of $\boldsymbol{x}''$ is bounded by Δ_A.

Similarly, $\hat{\boldsymbol{x}} := (\hat{\boldsymbol{x}}_{\bar{K}}, \hat{\boldsymbol{x}}_K)$, with $\hat{\boldsymbol{x}}_K := \boldsymbol{x}''$ and $\hat{\boldsymbol{x}}_{\bar{K}} := H_1 A_L \boldsymbol{d} - H_1 A_K \boldsymbol{x}''$, is a basic solution of:

$$\begin{aligned} &\boldsymbol{x}_{\bar{K}} + H_1 A_K \boldsymbol{x}_K = H_1 A_L \boldsymbol{d}, \\ &H_2 A_K \boldsymbol{x}_K = \boldsymbol{b}'/\kappa + H_2 A_L \boldsymbol{d}. \end{aligned}$$

Multiplying both sides of the equalities by $G = H^{-1}$ from the left yields:

$$A\boldsymbol{x} = G \begin{pmatrix} \boldsymbol{0}_{\bar{K}} \\ \boldsymbol{b}'/\kappa \end{pmatrix} + A_L \boldsymbol{d}.$$

Since G and A_L are submatrices of (A, I) and $\hat{\boldsymbol{x}}$ is a basic solution of this system, from Lemma 5, and the integrality of A, we have

$$\|\hat{\boldsymbol{x}}\|_\infty \leq m\Delta_A \|\boldsymbol{b}'/\kappa\|_\infty + m\Delta_A \|\boldsymbol{d}\|_\infty.$$

Let $\boldsymbol{v} = A'^{\top}(A'A'^{\top})^{-1}\boldsymbol{b}'$; that is, $A'\boldsymbol{v} = \boldsymbol{b}'$. Since the absolute value of an entry of A' is bounded by Δ_A by Cramer's rule, we have

$$\|\boldsymbol{b}'\|_\infty \leq m\Delta_A\|\boldsymbol{v}\|_\infty = m(m'nn'A_{\max}\Delta + m')\Delta_A\kappa$$

and thus: $\|\boldsymbol{x}''\|_\infty \leq \|\hat{\boldsymbol{x}}\|_\infty \leq m^2(m'nn'A_{\max}\Delta + m')\Delta_A^2 + m\Delta_A$. $\square$

Acknowledgment Research supported in part by Grant-in-Aid for Science Research (A) 26242027 of Japan Society for the Promotion of Science (JSPS), Grant-in-Aid for JSPS Fellows, the Natural Sciences and Engineering Research Council of Canada, and the Digiteo Chair C&O program. Part of this research was done while the authors were at the LRI, Université Paris-Sud, Orsay, France, within the Digiteo invited researchers program.

References

[1] K. Anstreicher and T. Terlaky: A monotonic build-up simplex algorithm for linear programming. *Operations Research* (1994) 556–561.

[2] N. Bonifas, M.D. Summa, F. Eisenbrand, N. Hähnle, and M. Niemeier: On sub-determinants and the diameter of polyhedra. *Discrete and Computational Geometry* (2014) 102–115.

[3] S. Chen and R. Saigal: A primal algorithm for solving a capacitated network flow problem with additional linear constraints. *Networks* (1977) 59–79.

[4] D. Dadush and N. Hähnle: On the Shadow Simplex method for curved polyhedra. Proceedings of the 31st International Symposium on Computational Geometry 2015 (to appear).

[5] G. Dantzig: Linear programming and extensions. Princeton University Press (1963).

[6] M. Dyer and A. Frieze: Random walks, totally unimodular matrices, and a randomized dual simplex algorithm. *Mathematical Programming* (1994) 1–16.

[7] K. Fukuda and T. Terlaky: Criss-cross methods: a fresh view on pivot algorithms. *Mathematical Programming* (1997) 369–395.

[8] G. Kalai and D. Kleitman: A quasi-polynomial bound for the diameter of graphs of polyhedra. *Bulletin of the American Mathematical Society* (1992) 315–316.

[9] T. Kitahara and S. Mizuno: Klee-Minty's LP and upper bounds for Dantzig's simplex method. *Operations Research Letters* (2011) 88–91.

[10] T. Kitahara and S. Mizuno: On the number of solutions generated by the dual simplex method. *Operations Research Letters* (2012) 172–174.

[11] T. Kitahara and S. Mizuno: A bound for the number of different basic solutions generated by the simplex method. *Mathematical Programming* (2013) 579–586.

[12] V. Klee and G. Minty: How good is the simplex method. In: O. Shisha (Ed.), Inequalities III, Academic Press, New York, (1972) 159–175.

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

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

[15] S. Mizuno: A strongly polynomial simplex method for totally unimodular *LP*. Technical Report 2014-3, Department of Industrial Engineering and Management, Tokyo Institute of Technology (2014).

[16] S. Mizuno, N. Sukegawa, and A. Deza: A primal-simplex based Tardos' algorithm. AdvOL-Report 2014/2, McMaster University (2014).

[17] J. Orlin: A dual version of Tardos's algorithm for linear programming. *Operations Research Letters* (1986) 221–226.

[18] K. Paparrizos: An infeasible (exterior) simplex algorithms for assignment problems. *Mathematical Programming* (1991) 45–54.

[19] F. Santos: A counterexample to the Hirsch conjecture. *Annals of Mathematics* (2012) 383–412.

[20] A. Schrijver: *Theory of Linear and Integer Programming*. John Wiley & Sons (1986).

[21] S. Smale : Mathematical problems for the next century. *The Mathematical Intelligencer* (1998) 7–15.

[22] N. Sukegawa and T. Kitahara: A refinement of Todd's bound for the diameter of a polyhedron. *Operations Research Letters* (*to appear*).

[23] E. Tardos: A strongly polynomial minimum cost circulation algorithm. *Combinatorica* (1985) 247–255.

[24] E. Tardos: A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research* (1986) 250–256.

[25] T. Terlaky: A convergent criss-cross method. *Mathematische Operationsforschung und Statistik Series Optimization* (1985) 683–690.

[26] M. Todd: An improved Kalai–Kleitman bound for the diameter of a polyhedron. *SIAM Journal on Discrete Mathematics* (2014) 1944–1947.

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

[28] Y. Ye: The simplex and policy-iteration methods are strongly polynomial for the Markov Decision Problem with a fixed discount rate. *Mathematics of Operations Research* (2011) 593–603.

[29] N. Zadeh: A bad network problem for the simplex method and other minimum cost flow algorithms. *Mathematical Programming* (1973) 255–266.