Revisiting Tardos's Framework for Linear Programming: Faster Exact Solutions using Approximate Solvers *

Daniel Dadush[1], Bento Natura[2], and László A. Végh[2]

[1]Centrum Wiskunde & Informatica, The Netherlands
[2]Department of Mathematics, London School of Economics and Political Science

16$^{\text{th}}$ September, 2020

Abstract

In breakthrough work, Tardos (Oper. Res. '86) gave a *proximity* based framework for solving linear programming (LP) in time depending only on the constraint matrix in the bit complexity model. In Tardos's framework, one reduces solving the LP $\min\langle c, x\rangle$, $Ax = b$, $x \geq 0$, $A \in \mathbb{Z}^{m \times n}$, to solving $O(nm)$ LPs in A having small integer coefficient objectives and right-hand sides using *any exact* LP algorithm. This gives rise to an LP algorithm in time $\text{poly}(n, m \log \Delta_A)$, where Δ_A is the largest subdeterminant of A. A significant extension to the real model of computation was given by Vavasis and Ye (Math. Prog. '96), giving a specialized interior point method that runs in time $\text{poly}(n, m, \log \bar{\chi}_A)$, depending on Stewart's $\bar{\chi}_A$, a well-studied condition number.

In this work, we extend Tardos's original framework to obtain such a running time dependence. In particular, we replace the exact LP solves with *approximate* ones, enabling us to directly leverage the tremendous recent algorithmic progress for approximate linear programming. More precisely, we show that the fundamental "accuracy" needed to *exactly* solve any LP in A is inverse polynomial in n and $\log \bar{\chi}_A$. Plugging in the recent algorithm of van den Brand (SODA '20), our method computes an optimal primal and dual solution using $O(mn^{\omega+1+o(1)} \log(\bar{\chi}_A + n))$ arithmetic operations, outperforming the specialized interior point method of Vavasis and Ye and its recent improvement by Dadush et al (STOC '20). By applying the preprocessing algorithm of the latter paper, the dependence can also be reduced from $\bar{\chi}_A$ to $\bar{\chi}_A^*$, the minimum value of $\bar{\chi}_{AD}$ attainable via column rescalings. Our framework is applicable to achieve the $\text{poly}(n, m, \log \bar{\chi}_A^*)$ bound using essentially any weakly polynomial LP algorithm, such as the ellipsoid method.

At a technical level, our framework combines together approximate LP solutions to compute exact ones, making use of constructive proximity theorems—which bound the distance between solutions of "nearby" LPs—to keep the required accuracy low.

1 Introduction

In this paper, we consider the task of computing exact primal and dual solutions for linear programs (LP) in standard form:

$$
\begin{array}{ll}
\min \langle c, x \rangle & \max \langle y, b \rangle \\
Ax = b & A^\top y + s = c \\
x \geq 0 \,, & s \geq 0 \,.
\end{array}
\tag{LP}
$$

Here, $A \in \mathbb{R}^{m \times n}$, $\mathrm{rank}(A) = m \leq n$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ are given in the input, and $x, s \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ are the variables. We consider the program in x to be the primal problem and the program in y, s to be the dual problem.

After the work of Khachiyan [Kha79], who gave the first polynomial algorithm for LP using the ellipsoid method, Megiddo [Meg83] asked whether there exists a "genuinely polynomial", now known as *strongly polynomial*, algorithm for LP. Informally, the goal is to find an algorithm that uses poly(n) *basic arithmetic operations* (e.g. addition, multiplication, etc.), where each such operation must be performed on numbers of size polynomial in the instance encoding length. While no such algorithm is known, the search for a strongly polynomial LP algorithm has spurred tremendous algorithmic advances for many classical combinatorial problems.

Strongly polynomial algorithms have indeed been found for important combinatorial classes of linear programs. Examples include feasibility for two variable per inequality systems [Meg83], minimum-cost circulations [GT89, Orl93, Tar85], generalized flow maximization, [Vég17, OV17], and discounted Markov Decision Processes [Ye05, Ye11].

To generalize these results to larger problem classes, a natural attempt is to seek abstract frameworks that capture known algorithms. In this vein, a recurring principle in strongly polynomial algorithm design is that "good enough" approximate solutions can be used to glean combinatorial information about exact optimal ones. Such information is used to reduce the underlying instance in a way that preserves all optimal solutions.

This was in fact the key idea in Tardos's seminal paper on minimum-cost circulations [Tar85]: solving a problem instance with a suitable rounded cost function reveals an arc that cannot be tight in any dual optimal solution; consequently, we can fix the flow value to 0. As another example, in submodular function minimization any sufficiently small norm point in the base polytope can be used to infer relations in a ring-family containing all minimizers [IFF01, DVZ18].

At a higher level, it can be useful to view strongly polynomial algorithms as reductions from an exact optimization problem to a suitable approximate version of itself. To achieve fast strongly polynomial algorithms using these principles, important considerations are the complexity of the individual approximate solves, e.g., the degree of accuracy required, and the total required number of them.

Tardos's Framework for Linear Programming   Generalizing the above idea from minimum-cost flows to general linear programming, Tardos [Tar86] provided such a framework for solving any standard form primal-dual LP with integer constraint matrix $A \in \mathbb{Z}^{m \times n}$ using a number of operations depending only on n and the logarithm of Δ_A, the maximum absolute value of the determinant of any square submatrix of A. This algorithm is strongly polynomial for minimum-cost flow, noting that digraph incidence matrices are totally unimodular, and therefore $\Delta_A = 1$. At a high level, Tardos's framework reduces getting exact LP solutions to getting exact solutions for "nearby LPs" with simpler coefficient structure, heavily relying on LP *proximity* theorems (e.g., see [Hof52, CGST86]). More precisely, Tardos reduces computing exact primal-dual solutions to max $\langle c, x \rangle$, $Ax = b, x \geq 0$ to computing exact primal-dual solutions to $O(nm)$ LPs in A with "rounded" objectives c' and right hand sides b' having integer coefficients of size $O(n^2 \Delta_A)$. In particular, after $O(n)$ such LP solves, one can determine a coefficient x_i in the support of some optimal solution, allowing to delete the $x_i \geq 0$ constraint. Due to their small coefficients,

the LPs in the reduction can be solved using any weakly polynomial algorithm. We note that the fundamental property enabling the polynomial solvability of these rounded LPs is that the minimum non-zero slack of their basic solutions, i.e., $\min\{x_i : x_i > 0\}$, is lower bounded by $1/(n^{O(1)}\Delta)$ by Cramer's rule.

Achieving $\bar{\chi}_A$ dependence While Tardos's framework is powerful, it inherently relies on the determinant bound Δ_A. This is only applicable for integer constraint matrices; one can obtain bounds for rational constraint matrices via multiplying by the least common denominator of the entries, but this leads to weak bounds that are highly volatile under small changes in the entries. A significant strengthening of [Tar86] was given by Vavasis and Ye [VY96]. They gave an interior point method (IPM) in the real model of computation based on *layered least squares* (LLS) steps that outputs exact primal-dual solutions in $O(n^{3.5}\log(\bar{\chi}_A+n))$ iterations. Improved iteration bounds were later given for certain special cases, in particular, $O(\sqrt{n}\log(\bar{\chi}_A+n))$ for homogeneous conic feasibility [VY95] and $O(n^{2.5}\log(\bar{\chi}_A+n))$ for LP feasibility [Ye06]. In a conceptual advance, Vavasis and Ye's result showed that the polynomial solvability of LP does not require any minimum non-zero slack assumption.

The condition measure replacing Δ_A is Stewart's $\bar{\chi}_A$ [Ste89], which for integer matrices satisfies $\bar{\chi}_A \leq n\Delta_A$. In contrast with Δ_A that relies on the entry numerics, $\bar{\chi}_A$ is a geometric measure that depends only on the kernel of A; Formally, letting $W := \ker(A)$ and $\pi_I(W) = \{x_I : x \in W\}$, one may define $\bar{\chi}_A := \bar{\chi}_W$ as the minimum number $M \geq 1$ such that for any $\emptyset \neq I \subseteq [n]$ and $z \in \pi_I(W)$, there exists $y \in W$ with $y_I = z$ and $\|y\| \leq M\|z\|$. In words, it represents the cost of lifting partial fixings of coordinates into the subspace W.

Very recently, the authors and Huiberts [DHNV20], building on the work of Monteiro and Tsuchiya [MT03, MT05], gave an improved LLS optimization algorithm and analysis requiring only $O(n^{2.5}\log n \log(\bar{\chi}_A^* + n))$ iterations, where $\bar{\chi}_A^*$ is the minimum $\bar{\chi}_{AD}$ over positive diagonal matrices $D > 0$. The paper [DHNV20] further gave a nearly optimal rescaling algorithm which runs in $O(m^2n^2 + n^3)$ time and computes $D > 0$ satisfying $\bar{\chi}_{AD} \leq n(\bar{\chi}_A^*)^3$. Thus, by suitable preprocessing, any algorithm achieving $\bar{\chi}_A$ dependence can be converted into one with $\bar{\chi}_A^*$ dependence.

A key tool in [DHNV20] is to study the 'circuit imbalance measure' κ_A. This closely approximates $\bar{\chi}_A$, with $\log(\bar{\chi}_A+n) = \Theta(\log(\kappa_A+n))$, and has very favourable combinatorial properties. Our approach also relies on κ_A and κ_A^*, even though we state the results in terms of the better known $\bar{\chi}_A$ and $\bar{\chi}_A^*$.

The condition number $\bar{\chi}_A^*$ can be smaller than $\bar{\chi}_A$ by an arbitrary factor, and in turn, $\bar{\chi}_A$ can be much smaller Δ_A even for integer matrices A. Let $A \in \mathbb{R}^{n \times m}$ be the node-edge incidence matrix of an undirected graph on n nodes and m edges. If the graph has k node-disjoint odd cycles, then $\Delta_A \geq 2^k$. However, it is easy to verify that for any graph, $\kappa_A \leq 2$ (see the definition of κ_A in Section 2.1). Using Proposition 2.7, we get the bound $\bar{\chi}_A \leq 2m$.

Harnessing the progress in approximate solvers The complexity of fast approximate LP algorithms has seen substantial improvements in recent years [LS19, CLS19, vdB20, vdBLSS20, LSZ19, JSWZ20]. Taking the recent algorithm [vdB20], given a feasible LP $\min \langle c, x \rangle$, $Ax = b, x \geq 0$, having an optimal solution of ℓ_2 norm at most R, for $\varepsilon > 0$ it computes a point $\tilde{x} \geq 0$ satisfying

$$\langle c, \tilde{x} \rangle \leq \min_{Ax=b, x \geq 0} \langle c, x \rangle + \varepsilon \cdot \|c\|_2 \cdot R \quad \text{and} \quad \|A\tilde{x} - b\|_2 \leq \varepsilon \cdot (\|A\|_F \cdot R + \|b\|_2), \quad \text{(APX-LP)}$$

in deterministic time $O(n^{\omega+o(1)}\log(n/\varepsilon))$, where $\omega < 2.38$ is the matrix multiplication exponent.

Tardos's framework requires an exact black box solver for systems with the same matrix A but replacing b and c by small integer vectors. It is possible to use the approximate solver (APX-LP) to obtain exact optimal solution for integer matrices for sufficiently small ε. Assume

3

$A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ and $\|b\|_\infty, \|c\|_\infty \leq n^{O(1)} \Delta^t$, and let OPT denote the optimum value of (LP). We may call (LP) in a suitable extended system with $\varepsilon = 1/\left(n^{O(1)} \Delta_A^{O(t)}\right)$, and use a Carathéodory reduction to identify primal and dual optimal basic solutions. Integrality is used in multiple parts of such a reduction: e.g., for establishing a bound $R = n^{O(1)} \Delta_A^{O(t)}$ from Cramer's rule, and for showing that for any primal feasible solution x, $\langle c, x \rangle < \text{OPT}$ implies $\langle c, x \rangle < \text{OPT} - \varepsilon \|c\|_2 R$. For a matrix $A \in \mathbb{R}^{m \times n}$, we cannot obtain an exact solver by applying the approximate solver for high enough accuracy in terms of the condition numbers $\bar{\chi}_A$ or κ_A. This is the main reason why we cannot work with explicitly rounded systems, but require a more flexible approach. Let us also note that recovering an exact solution from the approximate solver comes at a high arithmetic cost that we can save if using the approximate solution directly.

Fast algorithms with $\bar{\chi}_A$ dependence The layered least squares interior point methods discussed above represent substantial advances in the strongly polynomial solvability of LP, yet it is highly non-obvious how to combine these techniques with those of recent fast LP solvers. For example, for the results of [LS19, vdBLSS20], one would have to develop analogues of LLS steps for weighted versions of the logarithmic barrier. Furthermore, the proofs of exact convergence are intricate and deeply tied to the properties of the central path, and may leave one wondering whether the $\bar{\chi}_A$ solvability of LP is due to "IPM magic". It would therefore be desirable to have an elementary proof of the $\bar{\chi}_A$ solvability of LP.

Partial progress on this question was given by Tunçel and Ho [HT02], who generalized Tardos's framework in the real number model. Firstly, they showed that one can still round instances to have minimum non-zero slack $\tau_A > 0$, depending only on A. Second, they showed that applying the Mizuno-Todd-Ye [MTY93] predictor-corrector IPM on the homogeneous self-dual formulation, these rounded instances can be solved $\text{poly}(n, \log \tau_A, \log(\Delta_A/\delta_A))$ time, where δ_A is the absolute value of the minimum non-zero determinant of any square submatrix of A. Here, they prove the relation $\bar{\chi}_A \leq n \Delta_A/\delta_A$ and note that Δ_A/δ_A can be arbitrarily larger than $\bar{\chi}_A$. Lastly, they provide a different algorithm that removes the dependence on τ_A, assuming one has access to the Vavasis-Ye algorithm as a subroutine only on instances with $b \in \{\pm 1, 0\}^m$, $c \in \{0, \pm 1\}^n$.

1.1 Our Contributions

As our main contribution, we provide a substantially improved Tardos style framework for LP which achieves both $\bar{\chi}_A$ dependence and relies only on approximate LP solves: we use the output (APX-LP) of the approximate LP solvers in a black-box manner. Our main result using the deterministic solver in [vdB20] is summarized below. The more precise technical statements generalized to non-deterministic solvers are given as Theorem 6.4 for feasibility and Theorem 7.2 for optimization.

The system (Init-LP) is an extended system used for initialization, defined in Section 8.1.

Theorem 1.1 (Enhanced Tardos Framework for Feasibility). *Assume we are given a feasibility LP $Ax = b$, $x \geq 0$ with data $A \in \mathbb{R}^{m \times n}$, $\text{rank}(A) = m$, and $b \in \mathbb{R}^m$.*

(i) If the primal program is feasible, then one can find a feasible solution x using $O(m)$ approximate LP solves (APX-LP) with accuracy $\varepsilon = 1/(n\bar{\chi}_A)^{O(1)}$, on extended systems of the form (Init-LP), together with additional $O(mn^{\omega+o(1)})$ arithmetic operations. This gives a total complexity $O(mn^{\omega+o(1)} \log(\bar{\chi}_A + n))$ using the solver of van den Brand [vdB20].

(ii) If the primal program is infeasible, then a Farkas certificate of infeasibility $y \in \mathbb{R}^m$, satisfying $A^\top y \geq 0$, $\langle b, y \rangle < 0$ can be found using the amount of computation as in (i), and $O(nm^2 + n^{\omega+o(1)}) \log \log(\bar{\chi}_A + n))$ additional arithmetic operations.

Next, we state our result for optimization:

Theorem 1.2 (Enhanced Tardos Framework for Optimization). *Assume we are given primal-dual* (LP) *with data* $A \in \mathbb{R}^{m \times n}$, $\mathrm{rank}(A) = m$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.

    (i) *If both primal and dual programs are feasible, then one can obtain an optimal primal-dual pair* (x, y, s) *of solutions, using at most* $O(nm)$ *approximate LP solves* (APX-LP) *as in Theorem 1.1(i), together with an additional* $O(mn^{\omega+1+o(1)})$ *arithmetic operations. This gives a total complexity* $O(mn^{\omega+1+o(1)} \log^2(n) \log(\bar{\chi}_A + n))$ *using [vdB20].*

    (ii) *If either of the primal or dual programs are infeasible, then we can obtain a Farkas certificate of primal or dual infeasibility in the same running time as in (i), plus* $O(n^3 m^2 \log \log(\bar{\chi}_A + n))$ *additional arithmetic operations.*

This theorem yields the first LP algorithm achieving $\bar{\chi}_A$ dependence that is not based of the analysis of the central path. At a high level, we achieve this by more deeply exploiting the power of LP proximity theorems, which are already at the core of Tardos's framework. In the rest of this section, we explain some of the key ideas behind the above theorem and how it compares to Tardos's original algorithm as well as that of Vavasis and Ye.

Overview of the approach Both Tardos's and our approach use variants of Hoffman's proximity bounds, see Section 3. The fundamental difference is that while Tardos uses an exact solver where the perturbed objective and right hand side vectors are fixed in advance before calling the solver, we decide these perturbations "on the fly" as a function of the returned approximate solutions we receive.

Let us illustrate Tardos's and our approaches on the dual feasibility LP

$$A^\top y + s = c, s \geq 0. \tag{D}$$

The feasibility algorithm in [Tar85] proceeds as follows. Define $\tilde{b}_i = \sum_{i=1}^n (\Delta_A + 1)^{i-1} a_i$, where a_i is the i-th column vector of A, and consider the primal system

$$\min \langle c, x \rangle \quad \text{s.t. } Ax = \tilde{b}, \ x \geq 0. \tag{$\tilde{P}$}$$

Note that by the choice of $\tilde{b}$, this system is always feasible. If it is unbounded, then we may conclude infeasibility of (D). The reason for the particular choice of $\tilde{b}$ is that whenever the system is bounded, the dual of $(\tilde{P})$ has a unique optimal solution; this can be shown by a determinant argument. Consequently, for any optimal solution x^* to $(\tilde{P})$ and $S^* = \mathrm{supp}(x^*)$, the system $a_i^\top y = c_i$, $i \in S^*$ yields a feasible solution to (D). The exact LP solver will be applied to a series of rounded problem instances of the form

$$\min \langle \tilde{c}, x \rangle \quad \text{s.t. } Ax = \tilde{b}, \ x \geq 0, \ x_T = 0, \tag{$\hat{P}$}$$

where $\tilde{c} \in \mathbb{Z}^n$, $\|\tilde{c}\|_\infty \leq n^2 \Delta_A$, and $T \subseteq [n]$ is a set of indices i where we have already concluded that $x_i^* = 0$ in every optimal solution to $(\tilde{P})$. This is initialized as $T = \emptyset$, and every call to the LP solver enables the addition of at least one new index; thus, we need $O(n)$ oracle calls to solve feasiblity. According to the definition of $\tilde{b}$, this is an integer vector with $\|\tilde{b}\| = \Theta(\sqrt{m}\Delta_A^n)$. As explained above, we can obtain an exact solution to $(\hat{P})$ by calling (APX-LP) for accuracy $\varepsilon = 1/\left(n^{O(1)} \tilde{\Delta}_A^{O(n)}\right)$.

To conclude that $i \in T$ for some $i \in [n]$, Tardos uses a proximity theorem that is a variant of Lemma 3.4. It implies that if $\|\tilde{c} - c\|_\infty$ is "small", then $(\tilde{P})$ has a dual optimal solution that is "close" to the dual optimal solution obtained for $(\hat{P})$.

In contrast, our approach in Section 6 proceeds as follows. If $c \geq 0$, we simply return $s = c$. Otherwise, the norm of the negative coordinates $\|c^-\|_1$ will play a key role. We can strengthen (D) by adding the constraint

$$\|s - c\|_\infty \leq 16 \kappa_A^2 n \|c^-\|_1, \tag{1}$$

5

where κ_A is the circuit imbalance measure; for integer matrices $\kappa_A \leq \Delta_A$. A proximity result (Corollary 3.2) implies that whenever (D) is feasible, there is a feasible solution also satisfying (1).

We can use (APX-LP) directly to obtain a solution $(\tilde{y}, \tilde{s})$ such that $A^\top \tilde{y} + \tilde{s} = c$, $\|\tilde{s} - c\|_\infty \leq 3\kappa_A^2 n \|c^-\|_1$, and $\|\tilde{s}^-\|_\infty \leq \varepsilon \|c^-\|_1$ for $\varepsilon = 1/O(n^4 \kappa_A^4)$. Again, note that in addition to approximate feasiblity, we also require proximity of s to c; we can obtain such a solution with this extra property without an increase in the running time cost.

From here, we can identify a set K of coordinates such that $\tilde{s}_i$ is large enough to conclude that there exists a feasible solution s to (D) with $\tilde{s}_i > 0$ for $i \in K$; this is done similarly as in Tardos's approach.

We project out all variables in K, meaning that we remove the inequalities $a_i^\top y + s_i = c_i$ for $i \in K$ from the system. We recurse on the smaller subsystem. From the recursive call, we obtain a feasible solution y' to (D) in the smaller system that also satisfies (1). The proximity constraints enables us to easily map back y' to a feasible solution y to (D) by a simple 'pullback' operation.

As noted above, the very existence of an exact LP oracle heavily relies on the integrality assumption of A. This integrality is also used to establish the relation between the optimal solutions of ($\tilde{P}$) and the solutions of (D), using a determinant argument. In contrast, the proximity arguments as in Lemma 3.4 and Corollary 3.2 do not rely on integrality; we can use here κ_A instead of Δ_A.

Even for integer matrices and $\kappa_A = \Theta(\Delta_A)$, and using the same solver for (APX-LP), our algorithm is faster by a factor $\Omega(n^2/m)$. A key ingredient in the running time improvement is to strengthen the system with (1). This allows us to use $\varepsilon = 1/(n^{O(1)} \kappa_A^{O(1)})$; otherwise, we would need to require a higher precision $\varepsilon = 1/(n^{O(1)} \kappa_A^{O(n)})$. This yields a factor n improvement over [Tar85].

Another factor n/m improvement is obtained as follows. In the approach sketched above, if the set of "large" coordinates K is nonempty, we get a bound n on the number of recursive calls. Using a slightly more careful recursive setup, we can decrease the rank of the system at each iteration, improving this bound to m.

Let us now turn to optimization. Our algorithm will be more similar to the one in [Tar85], and for integer matrices with $\kappa_A = \Theta(\Delta_A)$ and $m = \Omega(n)$, the asymptotic running time bounds will be the same.

We now outline Tardos's approach. Given an optimization problem (LP), we first check for both primal and dual feasibility. If these are both feasible, then we go through $\leq m$ main loops. In each main loop, we use the same approach as above to solve ($\tilde{P}$) with a perturbed $\tilde{b} \in \mathbb{Z}^m$ with $\|\tilde{b}\|_\infty \leq n^2 \Delta_A$. Using $\leq n$ oracle calls, we obtain optimal primal and dual solutions (x, y, s). Again, proximity guarantees that if $\tilde{b}$ is "close" to b, then we can identify an index i with a "large" $x_i > 0$ where we can conclude $s_i^* = 0$ in every optimal solution. Equivalently, x_i is in the support of some optimal solution, and hence we may delete the constraint $x_i \geq 0$, and proceed to the next main loop after projecting out the variable x_i. We note that the bound n on the inner loops is in reality $n - m$, and this can be improved to m by swapping the primal and dual sides.

In our approach in Section 7, the goal is to end up in the same place as Tardos at the end of the main loop, where the difference will be how we get there. As mentioned above, in Tardos's setting, one already knows beforehand that the final objective and right hand side for which one will have optimal primal-dual solutions will be $\tilde{b}$, a rounded version of b, and the original c. However, the only important property is that at the end of the loop we end up with a primal-dual optimal pair for the original objective c, and *some* right hand side b' close enough to the original b. In particular, b' need not be known at the beginning of the algorithm and can thus

be chosen *adaptively* depending on the outcome of the approximate LP solves.

For the above purpose, we utilize proximity theorems (see Section 3 for precise statements) to allow us to stitch together the "large" coordinates of approximate dual solutions to achieve feasibility. At the same time, we perform a similar complementary stitching of primal approximate solutions, where we judiciously perturb "small" coordinates to 0, inducing a corresponding change of right hand side, to enforce complementarity with the dual solution. Here proximity allows us to control how much the solutions will change in future iterations, which is crucial to not destroying the structure of the solutions built so far.

We also note that Grötschel, Lovász, and Schrijver [GLS12, Theorem 6.6.3] give a different proof for Tardos's result using simultaneous Diophantine approximation (see also [FT87]). This shows that (LP) can be solved by creating a single perturbed instance with integer $\tilde{b}$ and $\tilde{c}$ bounded in terms of the encoding length of A such that the set of optimal bases coincide in the two systems. The perturbed instance can be solved in poly$(n, m, \log \Delta_A)$; we simply take an optimal basis and compute the corresponding primal and dual optimal solutions for the original b and c. However, this reduction inherently relies on integrality arguments.

Comparison to layered least squares IPM methods To setup a comparison, we first recall that standard log-barrier based IPMs follow the *central path* $\{(x(\mu), s(\mu), y(\mu)) : \mu > 0\}$ defined by the equations $x_i(\mu)s_i(\mu) = \mu, \forall i \in [n]$, together with feasibility for (LP). μ represents the normalized duality gap and (x_μ, y_μ, s_μ) converges to optimal solutions as $\mu \to 0$. The number of calls to the approximate LP solver above can be usefully compared to the number of so-called disjoint *crossover events* on the central path used in the analysis of the Vavasis–Ye algorithm [VY96]. A crossover event occurs for a pair of distinct indices (i, j) between the times $\mu^1 < \mu^0$, if $x_i(\mu^0)\bar{\chi}_A^n \geq x_j(\mu^0)$ and for all times $\mu' < \mu^1$, $x_j(\mu^0) > x_i(\mu^0)$. In words, an (i, j) crossover happens between time μ^0 and μ^1 if the variables x_i, x_j are "close" to being in the wrong order at time μ^0 and are in the correct order at all times after μ^1. The Vavasis and Ye LLS step was in fact designed to ensure that a new cross-over event occurs a "short time" after the step, i.e., sometime before $\mu/\bar{\chi}_A^n$ if the step ends at μ. From here, it is obvious that the number of distinct crossover events, i.e., on a new pair of indices, is bounded by $\binom{n}{2}$.

The approximate LP solves in our algorithm have the effect of inducing similar crossover type events, though this number is $O(mn)$ instead of $O(n^2)$. Precisely, after each LP solve, we are able identify two non-empty disjoint subsets of variables $I, J \subseteq [n]$, such that at least one of the variables $x_j, j \in J$, will end up being substantially larger than all the variables $x_i, i \in I$ in the final optimal solution. Lastly, the accuracy requirement of $\varepsilon = 1/(n\bar{\chi}_A)^{O(1)}$ for each LP solve is in a sense analogous to moving down the central path by that amount. We note that [DHNV20] gave an improved analysis of the Vavasis and Ye algorithm, showing that on "average" one sees $\Omega(1/\log n)$ (slightly different) crossover events after $(n\bar{\chi}_A)^{O(1)}$ time units, which is slightly worse than what we achieve here per approximate LP solve.

Failure will be certified Our algorithm requires an estimate on the circuit imbalance parameter κ_A (see definition in Section 2.1). This is a common assumption shared by most previous literature: Tardos's algorithm uses an estimate of Δ_A; Vavasis and Ye require a bound on $\bar{\chi}_A$. These parameters are hard to compute [Kha95, Tun99]. However, knowing these values are not required, and we can use the following simple guessing procedure, attributed to J. Renegar in [VY96]. We start with a low guess on $\bar{\chi}_A$ (or some other parameter), say $M = 100$. If the algorithm fails to return the required solution, then we conclude that the estimate was too low, and replace the guess M by M^2. Thus, we can still obtain a dependence on $\log(\bar{\chi}_A + n)$, without knowing the value.

A new aspect of our algorithm is that in case of a failure, we do not simply conclude that our estimate was too low indirectly from the failure of the algorithm, but we also obtain an explicit *certificate*. Namely, an elementary operation is to compute *lifts* mentioned previously:

for the subset $W = \ker(A)$, an index set $I \subseteq [n]$, and a vector $y \in \pi_I(W)$, we compute the minimum-norm vector $z \in W$ such that $z_I = y$. Our parameter κ_A satisfies $\|z\|_\infty \leq \kappa_A \|y\|_1$ (Proposition 2.6). Whenever our algorithm fails due to underestimating $M < \kappa_A$, this will be certified by an index set $I \subseteq [n]$ and a vector $y \in \pi_I(W)$, and lift z with $\|z\|_\infty > M\|y\|_1$.

1.2 Organization

In Section 2, we recall the subspace formulation of LP and review important properties of the condition numbers $\bar{\chi}_A$ and its combinatorial cousin, the circuit imbalance measure κ_A. In Section 3, we give a self-contained review of known LP proximity results, based on Hoffman type bounds, as well as novel variant (Theorem 3.6), which leverages the structure of the linear independence matroid on A. In Section 4, we present a constructive strongly polynomial time variant of Hoffman's proximity theorem, which will be useful for extracting Farkas infeasibility certificates from approximate solutions. In Section 5, we review the current state of the art approximate LP solvers and state our main theorems for extracting the solutions we need from these solvers in both the feasibility and optimization context, Theorems 5.2 and 5.3 respectively. The proofs of these theorems are deferred to Section 8, where we also describe the LP extended system we use (*Init-LP-sub*). In Section 6, give the describe our framework for LP feasibility, and in Section 7 our framework for LP optimization.

2 Preliminaries

We let $[n] := \{1, \dots, n\}$. Let $\mathbb{R}_{++}$ denote the set of positive reals, and $\mathbb{R}_+$ the set of nonnegative reals. We denote the support of a vector $x \in \mathbb{R}^n$ by $\mathrm{supp}(x) = \{i \in [n] : x_i \neq 0\}$. We let $\mathbb{1}_n$ denote the n-dimensional all-ones vector, or simply $\mathbb{1}$, whenever the dimension is clear from the context. Let e_i denote the i-th unit vector.

For vectors $v, w \in \mathbb{R}^n$ we denote by $\min\{v, w\}$ the vector $z \in \mathbb{R}^n$ with $z_i = \min\{v_i, w_i\}, i \in [n]$; analogously for $\max\{v, w\}$. Further, we use the notation $v^+ = \max\{v, 0_n\}$ and $v^- = \max\{-v, 0_n\}$;note that both v^+ and v^- are nonnegative vectors.

We will use ℓ_1, ℓ_2 and ℓ_∞ vector norms, denoted as $\|.\|_1, \|.\|_2$, and $\|.\|_\infty$, respectively. By $\|v\|$, we always mean the 2-norm $\|v\|_2$. Further, for a matrix $A \in \mathbb{R}^{m \times n}$, $\|A\|$ will refer to the $\ell_2 \to \ell_2$ operator norm, $\|A\|_F = \sqrt{\sum_{i,j} |A_{ij}|^2}$ to the Frobenius-norm, and $\|A\|_{\max} = \max_{i,j} |A_{ij}|$ to the max-norm.

For a vector $v \in \mathbb{R}^n$, we denote by $\mathrm{diag}(v)$ the diagonal matrix whose i-th diagonal entry is v_i. For two vectors $x, y \in \mathbb{R}^n$, we let $\langle x, y \rangle = x^\top y$ denote their scalar product. We denote by the binary operation $\circ$ the element-wise multiplication $x \circ y = \mathrm{diag}(x)y$. We let $\mathbf{D}$ denote the set of all positive definite $n \times n$ diagonal matrices.

For an index subset $I \subseteq [n]$, we use $\pi_I : \mathbb{R}^n \to \mathbb{R}^I$ for the coordinate projection. That is, $\pi_I(x) = x_I$, and for a subset $S \subseteq \mathbb{R}^n$, $\pi_I(S) = \{x_I : x \in S\}$. We let $\mathbb{R}_I^n = \{x \in \mathbb{R}^n : x_{[n] \setminus I} = 0\}$.

For a subspace $W \subseteq \mathbb{R}^n$, we let $W_I = \pi_I(W \cap \mathbb{R}_I^n)$. It is easy to see that $\pi_I(W)^\perp = (W^\perp)_I$. Assume we are given a matrix $A \in \mathbb{R}^{m \times n}$ such that $W = \ker(A)$. Then, $W_I = \ker(A_I)$, and we can obtain a matrix A' from A such that $\pi_I(W) = \ker(A')$ by performing a Gaussian elimination of the variables in $[n] \setminus I$.

For a subspace $W \subseteq \mathbb{R}^n$ and a vector $d \in \mathbb{R}^n$ we define by d/W the orthogonal projection of d onto $W^\perp$, that is $d/W = \pi_{W^\perp}(d)$. In particular, d/W is the minimum-norm vector in $W + d$. Further, for a subspace $W \subseteq \mathbb{R}^n$, we let $W_+ = W \cap \mathbb{R}_+^n$.

Linear programming in subspace formulation Let $A \in \mathbb{R}^{m \times n}$, and $W = \ker(A) \subseteq \mathbb{R}^n$. For $c, d \in \mathbb{R}^n$, we can write (LP) in the following equivalent form, where $d \in \mathbb{R}^n$ such that

$$Ad = b.$$

$$\begin{array}{ll} \min \; \langle c, x \rangle & \max \; \langle d, c - s \rangle \\ \quad x \in W + d & \quad s \in W^\perp + c \qquad (\textit{Primal-Dual}(W, d, c)) \\ \quad x \geq 0\,, & \quad s \geq 0\,. \end{array}$$

Note that (x, s) are optimal primal and dual solutions if and only if they are feasible and $\langle x, s \rangle = 0$. Thus, $\textit{Primal-Dual}(W, d, c)$ is equivalent to the following feasibility problem:

$$\begin{aligned} & x \in W + d \\ & s \in W^\perp + c \\ & \langle x, s \rangle = 0 \\ & x, s \geq 0\,. \end{aligned} \qquad (2)$$

Circuits For a linear subspace $W \subseteq \mathbb{R}^n$ and a matrix A such that $W = \ker(A)$, a circuit is an inclusion-wise minimal dependent set of columns of A. Recall that this corresponds to standard notion of circuits in the linear matroid associated with A.

This notion only depends on the subspace W, and not on the particular representation A; an equivalent definition is that $C \subseteq [n]$ is a circuit if and only if $W \cap \mathbb{R}^n_C$ is one-dimensional and that no strict subset of C has this property. The set of circuits of W is denoted $\mathcal{C}_W$.

For a subset $I \subseteq [n]$, we let $\mathrm{cl}(I)$ denote its *closure* in the matroidal sense. That is, $\mathrm{cl}(I) = J$ is the unique maximal set containing $J \supseteq I$ such that $\mathrm{rk}(A_J) = \mathrm{rk}(A_I)$. Equivalently,

$$\mathrm{cl}(I) = I \cup \{j \in [n] \setminus I : \exists C \in \mathcal{C}_W, j \in C \subseteq I \cup \{j\}\}$$

We will make the assumption that

$$|C| > 1 \quad \forall C \in \mathcal{C}_W\,, \qquad (3)$$

that is, W does not contain any loops (trivial circuits).

If $W = \ker(A)$, then a loop corresponds to zero-columns of A. For solving the feasibility problem $x \in W + d, x \geq 0$, we can eliminate all variables $i \in [n]$ that form a loop, without affecting feasibility. For the optimization problem $\textit{Primal-Dual}(W, d, c)$, if $i \in [n]$ forms a loop, and the primal problem $x \in W + d, x \geq 0$ is feasible, then $c_i < 0$ means that the problem is unbounded, and if $c_i \geq 0$, then we can find an optimal solution with $x_i = 0$.

Sign-consistent circuit decompositions We say that the vector $y \in \mathbb{R}^n$ is *sign-consistent* with $x \in \mathbb{R}^n$ if $x_i y_i \geq 0$ for all $i \in [n]$ and $x_i = 0$ implies $y_i = 0$ for all $i \in [n]$. Given a subspace $W \subseteq \mathbb{R}^n$, a *sign-consistent circuit decomposition* of a vector $z \in W$ is a decomposition

$$z = \sum_{k=1}^{h} g^k,$$

where $h \leq n$, the vectors $g^1, g^2, \ldots, g^h \in W$ are sign-consistent with z and $\mathrm{supp}(g^k) = C_k$ for all $k \in [h]$ for circuits $C_1, C_2, \ldots, C_h \in \mathcal{C}_W$.

Lemma 2.1. *For every subspace $W \subseteq \mathbb{R}^n$, every $z \in W$ admits a sign-consistent circuit decomposition.*

Proof. Let $F \subseteq W$ be the set of vectors sign-consistent with z. F is a polyhedral cone; its faces correspond to inequalities of the form $y_k \geq 0$, $y_k \leq 0$, or $y_k = 0$. The rays (edges) of F are of the form $\{\alpha g : \alpha \geq 0\}$ with $\mathrm{supp}(g) \in \mathcal{C}_W$. Clearly, $z \in F$, and thus, z can be written as a conic combination of at most n rays by the Minkowski-Weyl theorem. Such a decomposition yields a sign-consistent circuit decomposition. $\qquad \square$

2.1 The condition numbers $\bar{\chi}$ and κ

For a matrix $A \in \mathbb{R}^{m \times n}$, the condition number $\bar{\chi}_A$ is defined as

$$
\begin{aligned}
\bar{\chi}_A &= \sup \left\{ \left\| A^\top \left(ADA^\top \right)^{-1} AD \right\| : D \in \mathbf{D} \right\} \\
&= \sup \left\{ \frac{\|A^\top y\|}{\|p\|} : y \text{ minimizes } \left\| D^{1/2}(A^\top y - p) \right\| \text{ for some } 0 \neq p \in \mathbb{R}^n \text{ and } D \in \mathbf{D} \right\}.
\end{aligned}
\tag{4}
$$

This quantity was first studied by Dikin [Dik67], Stewart [Ste89], and Todd [Tod90], and has been extensively studied in the context of interior point methods; we refer the reader to [HT02, MT03, VY96] for further results and references.

It is important to note that $\bar{\chi}_A$ only depends on the subspace $W = \ker(A)$. Hence, we can also write $\bar{\chi}_W$ for a subspace $W \subseteq \mathbb{R}^n$, defined to be equal to $\bar{\chi}_A$ for some matrix $A \in \mathbb{R}^{k \times n}$ with $W = \ker(A)$. We will use the notations $\bar{\chi}_A$ and $\bar{\chi}_W$ interchangeably.

Proposition 2.2 ([TTY01, VY96])**.** *For every matrix $A \in \mathbb{R}^{m \times n}$,*

$$
\bar{\chi}_A = \max \left\{ \|A_B^{-1}A\| : A_B \text{ non-singular } m \times m\text{-submatrix of } A \right\}.
$$

Let us define the *lifting map* $L_I^W : \pi_I(W) \to W$ by

$$
L_I^W(p) = \arg\min \left\{ \|z\| : z_I = p, z \in W \right\}.
$$

Note that L_I^W is the unique linear map from $\pi_I(W)$ to W such that $L_I^W(p)_I = p$ and $L_I^W(p)$ is orthogonal to $W \cap \mathbb{R}^n_{[n] \setminus I}$. If $|I| = \Theta(n)$ any explicit computation of the lifting map requires $\Omega(n^2)$ operations as $L_I^W \in \mathbb{R}^{n \times |I|}$. The following lemma shows that the lift of a single vector can be obtained more efficiently if $m \ll n$.

Lemma 2.3. *Let $A \in \mathbb{R}^{m \times n}$, $W = \ker(A)$ and $I \subseteq [n]$. Then, both of the following can be done in time $O(\min\{m^2n, n^\omega\})$.*

    (i) Computing $L_I^W(p)$, for any $p \in \pi_I(W)$.

    (ii) Computing $L_I^{W^\perp}(q)$, for any $q \in \pi_I(W^\perp)$.

Proof. Let us begin with (i). First, obtain a vector $w \in W$ such that $w_I = p$. This can be done by solving the linear system $A_{[n] \setminus I} x = -A_I p$ and setting $w = (x, p)$ in time $O(\min\{m^2n, n^\omega\})$. The components in $[n] \setminus I$ of $L_I^W(p)$ are now given by the orthogonal projection of x onto $(W_{[n] \setminus I})^\perp$. Note that $W_{[n] \setminus I} = \ker(A_{[n] \setminus I})$. So, for $B := A_{[n] \setminus I}$ we have

$$
[L_I^W(p)]_{[n] \setminus I} = B^\top (BB^\top)^{-1} Bx,
$$

where $BB^\top \in \mathbb{R}^{m \times m}$ can be computed in $O(\min\{m^2n, n^\omega\})$ and its inverse in $O(m^\omega) \leq O(\min\{m^2n, n^\omega\})$. All other operations are matrix-vector products and can be computed in time $O(mn)$. Therefore, the overall running time of computing $L_I^W(p)$ is $O(\min\{m^2n, n^\omega\})$.

To show (ii), analogously to (i) we first obtain a vector $\hat{w} \in W^\perp$ such that $\hat{w}_I = q$. Note that $W^\perp = \mathrm{Im}(A^\top)$ and so we find $y \in \mathbb{R}^m$ such that $(A_I)^\top y = q$. This can again be done in time $O(\min\{m^2n, n^\omega\})$. Then, we can set $\hat{w} = A^\top y \in W^\perp$. The coordinates in $[n] \setminus I$ of $L_I^{W^\perp}(q)$ are now given by by the projection of $\hat{w}_{[n] \setminus I}$ onto $\pi_{[n] \setminus I}(W)$. The space $\pi_{[n] \setminus I}(W)$ can be represented as the kernel of a matrix $\hat{A}$, that arises by performing Gaussian elimination and pivoting the entries in I in time $O(nm^{\omega-1}) \leq O(\min\{m^2n, n^\omega\})$ [BH74, IMH82]. Then, we have

$$
[L_I^{W^\perp}(q)]_{[n] \setminus I} = I - \hat{A}^\top (\hat{A}\hat{A}^\top)^{-1} \hat{A} \hat{w}_{[n] \setminus I}.
$$

As in (i) this can be computed in time $O(\min\{m^2n, n^\omega\})$. $\qquad\square$

A useful characterization of $\bar{\chi}_W$ can be given in terms of the operator norm of the lifting map. This was shown in [DHNV20], by using results from [Ste89] and [O'L90].

Proposition 2.4 ([DHNV20]). *For a linear subspace* $W \subseteq \mathbb{R}^n$,

$$\bar{\chi}_W = \max \left\{ \|L_I^W\| : I \subseteq [n], I \neq \emptyset \right\} .$$

The circuit imbalance measure Consider now the circuits $\mathcal{C}_W$ of the subspace W. For a circuit $C \in \mathcal{C}_W$, let $g^C \in W$ be such that $\mathrm{supp}(g^C) = C$. Note that g^C is unique up to multiplication by scalar. We define the *circuit imbalance measure*

$$\kappa_W = \max \left\{ \frac{\max_{i \in C} \left|g_i^C\right|}{\min_{j \in C} \left|g_j^C\right|} : C \in \mathcal{C}_W \right\} ,$$

as the largest ratio between two entries of any minimum support nonzero vector in W. This was studied in [Vav94, HT02, DHNV20]. Note that $\kappa_W = 1$ corresponds to totally unimodular spaces. As shown in Proposition 2.7 below, the condition measures $\bar{\chi}_W$ and κ_W are closely related: $O(\log(\bar{\chi}_W + n)) = O(\log(\kappa_W + n))$ holds. However, κ_W has several advantageous combinatorial properties. It fits particularly nicely with the proximity results in Section 3 and 4. In fact, the argument in the proof of Tardos's main proximity result using Cramer's rule is implicitly bounding circuit imbalances, see discussion of Lemma 3.4. Therefore, we will use κ_W instead of $\bar{\chi}_W$ throughout the paper.

We can give a characterization analogous to Proposition 2.2, using max-norm instead of ℓ_2-norm.

Proposition 2.5. *For every matrix* $A \in \mathbb{R}^{m \times n}$ *with* $\mathrm{rk}(A) = m$,

$$\kappa_A = \max \left\{ \|A_B^{-1} A\|_{\max} : A_B \text{ non-singular } m \times m\text{-submatrix of } A \right\} .$$

Proof. Consider the matrix $A' = A_B^{-1} A$ for any non-singular $m \times m$ submatrix A_B. Let us renumber the columns such that B corresponds to the first m columns. Then, for every $m + 1 \leq j \leq n$, the jth column of A' represents a circuit where $x_j = 1$, and $x_i = -A'_{ij}$ for $i \in [m]$. Hence, $\|A'\|_{\max}$ gives a lower bound on κ_A.

Conversely, take the circuit solution g^C that gives the maximum in the definition of κ_A; let g_j^C be the minimum absolute value element. For any circuit C, we can select a base B such that $C \setminus \{j\} \subseteq B$. Then, the largest absolute value in the j-th column of $A_B^{-1} A$ will be κ_A. $\qquad\square$

Proposition 2.4 asserts that $\bar{\chi}_W$ is the maximum $\ell_2 \to \ell_2$ operator norm of the mappings L_I^W over $I \subseteq [n]$. We can show that the maximum $\ell_1 \to \ell_\infty$ operator norm of these mappings is κ_W, even though the lifting map is defined with respect to ℓ_2 norms.

Proposition 2.6. *For a linear subspace* $W \subseteq \mathbb{R}^n$,

$$\kappa_W = \max \left\{ \frac{\|L_I^W(p)\|_\infty}{\|p\|_1} : I \subseteq [n], I \neq \emptyset, p \in \pi_I(W) \setminus \{0\} \right\} .$$

Proof. We first show that for any $I \neq \emptyset$, and $p \in \pi_I(W) \setminus \{0\}$, $\|L_I^W(p)\|_\infty \leq \kappa_W \|p\|_1$ holds.

Let $z = L_I^W(p)$, and take a sign-consistent decomposition $z = \sum_{k=1}^h g^k$ as in Lemma 2.1. For each $k \in [h]$, let $C_k = \mathrm{supp}(g^k)$. We claim that all these circuits must intersect I. Indeed, assume for a contradiction that one of them, say C_1 is disjoint from I, and let $z' = \sum_{k=2}^h g^k$. Then, $z' \in W$ and $z'_I = z_I = p$. Thus, z' also lifts p to W, but $\|z'\|_2 < \|z\|_2$, contradicting the definition of $z = L_I^W(p)$ as the minimum-norm lift of p.

11

By the definition of κ_W, $\|g^k\|_\infty \leq \kappa_W \|g_I^k\|_1$ for each $k \in [h]$. The claim follows since $p = z_I = \sum_{k=1}^h g_I^k$, moreover, sign-consistency guarantees that $\|p\|_1 = \sum_{k=1}^h \|g_I^k\|_1$. Therefore,

$$\|z\|_\infty \leq \sum_{k=1}^h \|g^k\|_\infty \leq \kappa_W \sum_{k=1}^h \|g_I^k\|_1 = \kappa_W \|p\|_1.$$

We have thus shown that the maximum value in the statement is at most κ_W. To show that equality holds, let $C \in \mathcal{C}_W$ be the circuit and $g^C \in W$ the corresponding vector and $i, j \in C$ such that $\kappa_W = |g_i^C| / |g_j^C|$.

Let us set $I = ([n] \setminus C) \cup \{j\}$, and define $p_k = 0$ if $k \in [n] \setminus C$ and $p_j = g_j^C$. Then $p \in \pi_I(W)$, and the unique extension to W is g^C; thus, $L_I^W(p) = g^C$. We have $\|L_I^W(p)\|_\infty = |g_i^C|$. Noting that $\|p\|_1 = |g_j^C|$, it follows that $\kappa_W = \|L_I^W(p)\|_\infty / \|p\|_1$. $\qquad\square$

Using Proposition 2.6, we can easily relate the quantities $\bar{\chi}_W$ and κ_W. The upper bound was already shown in [Vav94]; the slightly weaker lower bound $\sqrt{\bar{\chi}_W^2 - 1}/n \leq \kappa_W$ was given in [DHNV20].

Proposition 2.7. *For a linear subspace $W \subseteq \mathbb{R}^n$,*

$$\frac{1}{n}\bar{\chi}_W \leq \kappa_W \leq \sqrt{\bar{\chi}_W^2 - 1}.$$

Proof. Note that the inequality $\frac{1}{n}\bar{\chi}_W \leq \kappa_W \leq \bar{\chi}_W$ is a simple consequence from Proposition 2.4, Proposition 2.6 and bounds between the ℓ_∞, ℓ_1, and ℓ_2-norms. Let us now turn to the second inequality. Note that by assumption (3), Proposition 2.4 implies $\bar{\chi}_W \geq \sqrt{2}$. So the inequality already holds if $\kappa_W = 1$. For the rest, assume that $\kappa_W > 1$. We pick an element $p \in \mathbb{R}^I, p \in \pi_I(W)$ for which κ_W attains its value. Then for $J = [n] \setminus I$

$$\kappa_W = \frac{\|L_I^W(p)\|_\infty}{\|p\|_1} = \frac{\|L_I^W(p)_J\|_\infty}{\|p\|_1} \leq \frac{\|L_I^W(p)_J\|_2}{\|p\|_2} = \sqrt{\frac{\|L_I^W(p)\|_2^2}{\|p\|_2^2} - 1} \leq \sqrt{\bar{\chi}_W^2 - 1},$$

where the second equality used $\|L_I^W(p)_J\|_\infty > \|L_I^W(p)_I\|_\infty = \|p\|_\infty$, as $\kappa_W > 1$. $\qquad\square$

We state two more useful properties of $\bar{\chi}_W$ and κ_W. The first property shows that $\bar{\chi}_W$ and κ_W are self-dual, shown respectively in [GL97] and in [DHNV20].

Proposition 2.8 ([GL97, DHNV20]). *For any linear subspace $W \subseteq \mathbb{R}^n$, we have $\bar{\chi}_W = \bar{\chi}_{W^\perp}$ and $\kappa_W = \kappa_{W^\perp}$.*

The next proposition asserts that these condition numbers are non-increasing under fixing and projecting variables.

Proposition 2.9. *For any linear subspace $W \subseteq \mathbb{R}^n$ and $J \subseteq [n]$, we have*

$$\bar{\chi}_{W_J} \leq \bar{\chi}_W, \quad \bar{\chi}_{\pi_J(W)} \leq \bar{\chi}_W, \quad \kappa_{W_J} \leq \kappa_W, \quad and \quad \kappa_{\pi_J(W)} \leq \kappa_W.$$

Proof. The statements on W_J are immediate from Proposition 2.4 and Proposition 2.6. The two other statements follow from the statements on W_J, Proposition 2.8 and noting that $\pi_J(W)^\perp = (W^\perp)_J$. $\qquad\square$

The estimate M and lifting certificates The value of κ_W and $\bar{\chi}_A$ may not be known. In fact, these are hard to approximate even within a factor $2^{\mathrm{poly}(m)}$ [Tun99]. Throughout our algorithms, we maintain a guess M on the value of $2\kappa_W$, initialized as $M = 2$. At certain points in the algorithm, we may find an index set $I \subseteq [n]$ and a vector $p \in \pi_I(W)$ such that $\|L_I^W(p)\|_\infty > M\|p\|_1$. In this case, we conclude that $M < \kappa_W$ by Proposition 2.6. Such a pair (I, p) is called a *lifting certificate* of $M > \kappa$. We can then restart the algorithm with an updated estimate $M' = \max\{2\|L_I^W(p)\|_\infty / \|p\|_1, M^2\}$.

Remark 2.10. *During the algorithm, we project out variable sets $J \subset [n]$ and work recursively with the space $W' := \pi_{[n]\setminus J}(W)$. A lifting certificate for W' is then a pair (I, p) with $I \subset [n] \setminus J$, $p \in \mathbb{R}^I$, such that $\|L_I^{W'}(p)\|_\infty > M\|p\|_1$. While Proposition 2.9 already certifies that the guess M is wrong for W, it is unsatisfactory that the obtained certificate holds for a different space. But it is easy to see that the certificate still holds up to a factor of $\sqrt{n}$ also for the original space W: Let $\hat{p} \in W$ be an arbitrary vector such that $p_{[n]\setminus I} = L_I^{W'}(p)$. Then*

$$\sqrt{n}\|L_I^W(p)\|_\infty \geq \|L_I^W(p)\| \geq \|[L_I^W(p)]_{[n]\setminus J}\| \geq \|L_I^{W'}(p)\| \geq \|L_I^{W'}(p)\|_\infty > M\|p\|_1.$$

In particular, the inequality above shows $\|L_I^W(p)\| > M\|p\|_1 \geq M\|p\|$, so (I, p) is a certificate for W in the classical ℓ_2-norm. For ease of presentation we disregard this detail in the remainder of the paper.

Optimal rescalings For every $D \in \mathbf{D}$, we can consider the condition numbers $\bar{\chi}_{WD} = \bar{\chi}_{AD^{-1}}$ and $\kappa_{WD} = \kappa_{AD^{-1}}$. We let

$$\bar{\chi}_W^* = \bar{\chi}_A^* = \inf\{\bar{\chi}_{WD} : D \in \mathbf{D}\}$$
$$\kappa_W^* = \kappa_A^* = \inf\{\kappa_{WD} : D \in \mathbf{D}\}$$

denote the best possible values of $\bar{\chi}$ and κ that can be attained by rescaling the coordinates of W. A near-optimal rescaling can be found in strongly polynomial time .

Theorem 2.11 ([DHNV20]). *There is an $O(n^2 m^2 + n^3)$ time algorithm that for any matrix $A \in \mathbb{R}^{m \times n}$ and $W = \ker(A)$, computes a value t such that*

$$t \leq \bar{\chi}_W \leq t(\bar{\chi}_W^*)^2$$

and a $D \in \mathbf{D}$ such that

$$\kappa_{WD} \leq (\kappa_W^*)^3 \quad and \quad \bar{\chi}_{WD} \leq n(\bar{\chi}_W^*)^3 \,.$$

As a consequence, after using this preprocessing step, any algorithm that has running time dependence on $\log(\kappa_W + n)$ is turned into an algorithm with dependence on $\log(\kappa_W^* + n)$. We note however that for small values of $\log(\kappa_W + n)$, this preprocessing may turn out to be a bottleneck operation for our feasibility algorithm.

3 Proximity via Hoffman-bounds

Hoffman's seminal work [Hof52] has analyzed proximity of LP solutions. Given $P = \{x \in \mathbb{R}^n : Ax \leq b\}$, $x_0 \in \mathbb{R}^n$, and norms $\|.\|_\alpha$ and $\|.\|_\beta$, we are interested in the minimum of $\|x - x_0\|_\alpha$ over $x \in P$. Hoffman showed that this can be bounded as $H_{\alpha,\beta}(A)\|(Ax_0 - b)^+\|_\beta$, where the Lipschitz-bound $H_{\alpha,\beta}(A)$ is a constant that only depends on A and the norms. Such bounds have been shown for several different problem forms and norms; we refer the reader to [PVZ20] for results and references.

We will use a Hoffman-bound for a system of the form $x \in W$, $\ell \leq x \leq u$. We show that $H_{\infty,1} = \kappa_W$ for such a system. Related bounds using $\bar{\chi}_A$ have been shown in [HT02]; here, we present a self-contained proof.

For vectors $d, c \in \mathbb{R}^n$, let us define the set

$$\Lambda(d, c) := \mathrm{supp}(d^-) \cup \mathrm{supp}(c^+) \,. \tag{5}$$

Theorem 3.1 (Hoffman Proximity Theorem). *Let $W \subseteq \mathbb{R}^n$ be a subspace and $\ell \in (\mathbb{R} \cup \{-\infty\})^n$, $u \in (\mathbb{R} \cup \{\infty\})^n$ be lower and upper bounds, and assume that $P = \{x \in W : \ell \leq x \leq u\}$ is non-empty. Then, for every $x \in P$ we have*

$$\|\ell^+ + u^-\|_1 \leq \|x_{\Lambda(u,\ell)}\|_1,$$

and there exists $x \in P$ such that

$$\|x\|_\infty \leq \kappa_W \|\ell^+ + u^-\|_1.$$

Proof. Let us start with the first statement. Clearly, $\operatorname{supp}(u^-) \cap \operatorname{supp}(\ell^+) = \emptyset$. If $u_i < 0$, then $|x_i| \geq |u_i|$, and if $\ell_i > 0$, then $|x_i| \geq |\ell_i|$. Thus, $\|\ell^+ + u^-\|_1 \leq \|x_{\Lambda(u,\ell)}\|_1$ follows for every $x \in P$.

For the second statement, select $x \in P$ such that $\|x\|_1$ minimal and let $x = \sum_{k=1}^h g^k$ be a sign-consistent circuit decomposition of x as in Lemma 2.1. For each $k \in [h]$, we show that $C_k = \operatorname{supp}(g^k)$ must either contain an element $i \in C_k$ with $x_i = u_i < 0$, or with $x_i = \ell_i > 0$. For a contradiction, assume that one of them, say C_1, contains no such element. Then, for some $\varepsilon > 0$, $x' = (1 - \varepsilon)g^1 + \sum_{k=2}^h g^k \in P$, giving a contradiction, since $\|x'\|_1 < \|x\|_1$.

The inequality $\|x\|_\infty \leq \kappa_W \|\ell^+ + u^-\|_1$ then follows as in the proof of Proposition 2.6. $\square$

We can derive useful corollaries for feasibility and optimization problems.

Corollary 3.2. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $d \in \mathbb{R}^n$. If the system $x \in W + d$, $x \geq 0$ is feasible, then the system*

$$x \in W + d$$
$$\|x - d\|_\infty \leq \kappa_W \|d^-\|_1$$
$$x \geq 0,$$

is also feasible.

Proof. Using the variable $z = d - x$, the system can be equivalently written as $z \in W$, $z \leq d$. Thus, Theorem 3.1 guarantees a solution with $\|z\|_\infty \leq \kappa_W \|d^-\|_1$, as required. $\square$

Corollary 3.3. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d \in \mathbb{R}^n$, and assume $c \geq 0$. If Primal-Dual(W, d, c) is feasible, then there is an optimal solution (x, s) such that*

$$\|x - d\|_\infty \leq \kappa_W \|d_{\Lambda(d,c)}\|_1.$$

Proof. Let (x^*, s^*) be in Primal-Dual(W, d, c). That is, x^* minimizes $\langle c, x \rangle$ over $x \in W + d$, $x \geq 0$. Consider the feasibility system $x \in W + d$, $x \geq 0$, and $x_i \leq x_i^* \ \forall i \in \operatorname{supp}(c)$. Note that x^* is a feasible solution. In fact, the inequality $x_i \leq x_i^*$ must be tight for all $i \in \operatorname{supp}(c)$, as otherwise $\langle c, x \rangle < \langle c, x^* \rangle$, contradicting the optimality of x^*.

For $z = d - x$, we get the system $z \in W$, $\ell \leq z \leq d$, where $\ell_i = d_i - x_i^*$ if $i \in \operatorname{supp}(c)$, and $\ell_i = -\infty$ if $c_i = 0$. Note that $\ell_i^+ \leq d_i^+$, for $i \in \operatorname{supp}(c) = \operatorname{supp}(c^+)$, and therefore,

$$\|d_{\Lambda(d,c)}\|_1 = \|d_{\operatorname{supp}(c)}^+\|_1 + \|d^-\|_1 \geq \|\ell^+ + d^-\|_1.$$

Thus, the claim follows by Theorem 3.1. $\square$

The next lemma will be used to conclude that a primal variable $s_i^* = 0$ in every solution (x^*, s^*) to Primal-Dual(W, d, c). For integer matrices, a similar statement was given by Cook et al. [CGST86, Theorem 5], see also [Sch98, Theorem 10.5] with a bound in terms of the maximum subdeterminant Δ_A. A variant of this statement is used by Tardos [Tar85, Lemma 1.1] as the main underlying proximity statement of her algorithm. Ho and Tunçel [HT02, Theorem 6.3] generalized this bound to arbitrary matrices, using the condition number $\bar{\chi}_A$. This implies our statement with $n\kappa_W$ instead of $\kappa_W + 1$. We note that the arguments in [CGST86, Tar85] are based on Cramer's rule. In essence, this is used to bound the circuit imbalances in terms of Δ_A. Hence, our formulation with κ_W can be seen as a natural extension.

Lemma 3.4. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d, \tilde{d} \in \mathbb{R}^n$. Let $(\tilde{x}, s)$ be an optimal solution to* Primal-Dual$(W, \tilde{d}, c)$*. Then there exists an optimal solution (x^*, s^*) to* Primal-Dual(W, d, c) *such that*

$$\|x^* - \tilde{x}\|_\infty \leq (\kappa_W + 1)\|d - \tilde{d}\|_1 \,.$$

Proof. Let $x = \tilde{x} - \tilde{d} + d$. Note that $W + x = W + d$, and also $W^\perp + s = W^\perp + c$. Thus, the systems Primal-Dual(W, d, c) and Primal-Dual(W, x, s) define the same problem.

We apply Corollary 3.3 to W, x, s. This guarantees the existence of an optimal (x^*, s^*) to Primal-Dual(W, x, s) such that $\|x^* - x\|_\infty \leq \kappa_W \|x_{\Lambda(x,s)}\|_1$. Recall that $\Lambda(x, s) = \operatorname{supp}(x^-) \cup \operatorname{supp}(s^+)$, and thus, $\|x_{\Lambda(x,s)}\|_1 = \|x^-\|_1 + \|x^+_{\operatorname{supp}(s^+)}\|_1$.

Since $\tilde{x} \geq 0$, we get that $\|x^-\|_1 \leq \|x_{\operatorname{supp}(x^-)} - \tilde{x}_{\operatorname{supp}(x^-)}\|_1$. Second, by the optimality of $(\tilde{x}, s)$, we have $\tilde{x}_{\operatorname{supp}(s^+)} = 0$, and thus $x_{\operatorname{supp}(s^+)} = x_{\operatorname{supp}(s^+)} - \tilde{x}_{\operatorname{supp}(s^+)}$. Noting that $x - \tilde{x} = d - \tilde{d}$, these together imply that

$$\|x^* - \tilde{x}\|_\infty \leq \|x^* - x\|_\infty + \|x - \tilde{x}\|_\infty \leq (\kappa_W + 1)\|d - \tilde{d}\|_1 \,. \qquad \square$$

We can immediately use this theorem to derive a conclusion on the support of the optimal dual solutions to Primal-Dual(W, d, c), using the optimal solution to Primal-Dual$(W, \tilde{d}, c)$.

Corollary 3.5. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d, \tilde{d} \in \mathbb{R}^n$. Let $(\tilde{x}, s)$ be an optimal solution to* Primal-Dual$(W, \tilde{d}, c)$*. Let*

$$R := \{i \in [n] : \tilde{x}_i > (\kappa_W + 1)\|d - \tilde{d}\|_1\} \,.$$

Then for every dual optimal solution s^* to Primal-Dual(W, d, c)*, we have $s^*_R = 0$.*

Proof. By Lemma 3.4 there exists an optimal solution (x', s') to Primal-Dual(W, d, c) such that $\|x' - \tilde{x}\|_\infty \leq (\kappa_W + 1)\|d - \tilde{d}\|_1$. Consequently, $x'_R > 0$, implying $s^*_R = 0$ for every dual optimal s^* by complementary slackness. $\qquad \square$

We now formulate a strengthening of this corollary. We show that besides setting dual variables in R to 0, we are also able to set certain primal variables to 0. This will be the key to decrease the number of recursive calls from n to m.

More precisely, we show the following. Assume x' in the previous proof contains a 'large' coordinate set I_L, significantly larger than the threshold for R in Corollary 3.5. Assume that the closure $\operatorname{cl}(I_L)$ contains some indices from $[n] \setminus R$. Then, we can transform x' in the proof to another optimal solution x'' where all these coordinates are set to 0. This can be achieved by changing the coordinates in I_L only, and their high value in x'' guarantees that they remain positive.

Theorem 3.6. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d, \tilde{d} \in \mathbb{R}^n$. Let $(\tilde{x}, s)$ be an optimal solution to* Primal-Dual$(W, \tilde{d}, c)$*, and let $\tau \geq (\kappa_W + 1)\|d - \tilde{d}\|_1$ and $T \geq (2n\kappa_W + 1)\tau$. Let us define the following partition of $[n]$ into large, medium, and small indices.*

$$I_L = \{i \in [n] : \tilde{x}_i > T\}, \quad I_M = \{i \in [n] : T \geq \tilde{x}_i > \tau\}, \quad I_S = \{i \in [n] : \tau \geq \tilde{x}_i\} \,.$$

We further partition I_S as

$$I_S^0 = I_S \cap \operatorname{cl}(I_L), \quad I_S^+ = I_S \setminus \operatorname{cl}(I_L) \,.$$

Then, there exists a primal optimal solution x'' to Primal-Dual(W, d, c) *such that $x''_{I_L \cup I_M} > 0$, and $x''_{I_S^0} = 0$.*

Proof. Note that $I_L \cup I_M \subseteq R$ in Corollary 3.5, and consider the same optimal x' with $\|x' - \tilde{x}\|_\infty \leq \tau$ as guaranteed by Lemma 3.4. Thus, $x'_i > 2n\kappa_W\tau$ for $i \in I_L$, $x'_i > 0$ for $i \in I_M$, and $2\tau \geq x'_i \geq 0$ for $i \in I_S$.

For every $i \in I_S^0$, there exists a circuit $C \in \mathcal{C}_W$ with $i \in C \subseteq I_L \cup \{i\}$. Consequently, there exists a vector $z \in W$ such that $z_{I_S^0} = x'_{I_S^0}$ and $z_{I_M \cup I_S^+} = 0$ (note that we can find such a z with arbitrary values on I_S^0). We now take the lift

$$w = L_{I_M \cup I_S}^W \left(z_{I_M \cup I_S} \right).$$

Since $|x'_i| \leq 2\tau$ for all $i \in I_{S^0}$, we get $\|z_{I_M \cup I_S}\|_1 \leq 2\tau n$. Therefore, $\|w\|_\infty \leq 2n\kappa_W\tau$ by Proposition 2.6. Consequently, for $x'' := x - w$ we have $x'' \geq 0$, $x''_{I_{S^0}} = 0$ and $x''_{I_L \cup I_M} > 0$. Note that x'' is also an optimal solution, since $\text{supp}(x'') \subseteq \text{supp}(x')$. The claim follows. $\qquad\square$

4   Constructive proximity algorithms

In this section, we give an algorithmic implementation of the Hoffman Proximity Theorem (Theorem 3.1), assuming that a feasible solution is already given: we obtain another feasible solution in strongly polynomial time that also satisfies the proximity bounds. This is given in Corollary 4.4.

This can be derived from a 'Carathéodory-type' algorithm that we present in a more general form, as it may be of independent interest. We present the algorithm in two stages, with a basic subroutine in Lemma 4.1, and the main algorithm described in Lemma 4.3.

Lemma 4.1. *Let $A \in \mathbb{R}^{m \times n}$ with $\text{rk}(A) = m$. Let $W = \ker(A)$, $y \in W \setminus \{0\}$, and $J \subseteq [n]$ such that $y_J \neq 0$. Then, in $O(m(n-m)^2 + nm^{\omega-1})$ time, we can find a vector*
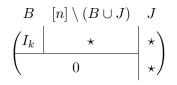
$$z = y - \sum_{j=1}^{t} \alpha_j y^{(j)},$$

such that

- *$z \in W$ such that z is sign-consistent with y and $z_J = 0$ ($z = 0$ is possible);*

- *$t \leq n$, $\alpha_1, \ldots, \alpha_t \geq 0$, $\sum_{j=1}^{t} \alpha_j = 1$, and*

- *for $j \in [t]$, $y^{(j)} \in W$ and $y^{(j)}$ has inclusion-wise minimal support subject to $y_J^{(j)} = y_J$.*

Proof. If $y_J = 0$, then we can simply output $z = y$. Otherwise, we set $\hat{y} = y$, and gradually modify $\hat{y}$.

At the beginning iteration τ of the algorithm, we maintain $\hat{y} \in W$ and $\hat{\alpha} \in [0,1]$, such that $\hat{y}$ is sign-consistent with y, $\hat{y} = y - \sum_{j=1}^{\tau-1} \alpha_j y^{(j)}$, $\alpha_j \geq 0$, $\sum_{j=1}^{\tau-1} \alpha_j = 1 - \hat{\alpha}$, and $y^{(j)} \in W$ satisfy the required property. We have $\hat{\alpha} = 1$ at the beginning and $\hat{\alpha} = 0$ at termination. At termination, we return $z = \hat{y}$, noting that $\hat{y}_J = 0$.

Further, we will maintain an index set $T \subseteq [n] \setminus J$, initialized as $T = \emptyset$, such that $\hat{y}_T = 0$. Every iteration will add at least one new index into T. If the method fails at any point, then we obtain a support minimal $y' \in W$ as in the second alternative.

Throughout, we maintain a maximal set $B \subseteq [n] \setminus (J \cup T)$ of linearly independent columns; $|B| = k \leq m$. This can be obtained initially using Gaussian elimination; we update B each time T is extended. Thus, we transform A with row operations to the following form.

$$\begin{array}{ccc} B & [n] \setminus (B \cup J) & J \\ \left(\begin{array}{c|c|c} I_k & \star & \star \\ \hline 0 & & \star \end{array} \right) \end{array}$$

Note that row operations do not affect $\ker(A)$; thus, we still have $W = \ker(A)$.

Consider now any iteration $\tau \geq 1$, with the matrix given in the above form, and the current vector $\hat{y} \in W$ with $\hat{y}_J = \hat{\alpha} y_J$. Let us define the vector y' as

$$
y'_i = \begin{cases}
y_i, & \text{if } i \in J, \\
-A_{i,J} y_J, & \text{if } i \in B, \\
0, & \text{otherwise}.
\end{cases}
$$

Here, $A_{i,J}$ is the i-th row of the matrix restricted to the columns in J. First, note that the construction guarantees $y' \in W$, with $y'_J = y_J$. Further, y' has minimal support subject to $y'_J = y_J$, because the columns of A corresponding to B are pairwise supported on disjoint rows.

Let us update $\hat{y}$ to $\hat{y}' = \hat{y} - \alpha y'$ for the smallest value $\alpha \geq 0$ such that $\hat{y}'_i = 0$ for some $i \in B$, or $\alpha = \hat{\alpha}$; we set the new value of $\hat{\alpha}$ to $\hat{\alpha}' = \hat{\alpha} - \alpha$. Clearly, $\hat{y}' \in W$ in either case, and $\hat{y}'_J = \hat{\alpha}' y_J$. We add $y^{(\tau)} = \hat{y}'$ to the combination with $\alpha_\tau = \alpha$.

If $\alpha = \hat{\alpha}$, then we terminate at the end of this iteration. If $\alpha < \hat{\alpha}$, then we add all columns where $\hat{y}'_i = 0$ to the set T; thus, T is extended by at least one element. We then update B by pivot operations, removing all indices from $B \cap T$, and replacing them by columns in $[n] \setminus (J \cup T)$ if possible; the size of B may also decrease.

Since T is extended in each step, the number of iterations is bounded by n. The initial Gaussian elimination to find a basis can be performed in $O(nm^{\omega-1})$ time [BH74, IMH82]. There are at most $n - m$ additional pivot operations, each of these taking $O(m(n-m))$ time, giving a total running time bound $O(m(n-m)^2 + nm^{\omega-1})$. $\qquad\square$

We will use the following proximity variant later:

Corollary 4.2. *Let $A \in \mathbb{R}^{m \times n}$ with $\mathrm{rk}(A) = m$. Let $W = \ker(A)$, $y \in W$, $M \geq 1$, and $J \subseteq [n]$. In $O(m(n-m)^2 + nm^{\omega-1})$ time, we can either find a vector $z \in W$, such that z is sign-consistent with y, $\|z - y\|_\infty \leq M\|y_J\|_1$, and $z_J = 0$, or find a lifting certificate of $M < \kappa_W$.*

Proof. If $y_J = 0$, we simply output $z = y$. Otherwise, we consider the output $z \in W$ and $y^{(j)}$, α_j, $j \in [t]$ as in Lemma 4.1. We have

$$
\|z - y\|_\infty = \left\| \sum_{j=1}^{t} \alpha_j y^{(j)} \right\|_\infty \leq \sum_{j=1}^{t} \alpha_j \left\| y^{(j)} \right\|_\infty.
$$

If $\max_j \|y^{(j)}\|_\infty \leq M\|y_J\|_1$, then z satisfies the requirements.

Assume now that for some $y' = y^{(j)}$, we have $\|y'\|_\infty > M\|y_J\|_1$. In this case, we claim that the set $I = J \cup ([n] \setminus \mathrm{supp}(y'))$ and $p = y'_I$ form a lifting certificate of $M < \kappa_W$.

Clearly, $y'_I \in \pi_I(W)$, and by the support minimality of y', there is a unique lift of y'_I to W, namely, $L_I^W(p) = y'$. We have $\|p\|_1 = \|y'_J\|_1 = \|y_J\|_1$. Hence, $\|L_I^W(p)\|_\infty / \|p\|_1 > M$ follows. $\qquad\square$

The next statement formulates the outcomes of the general algorithm, using Lemma 4.1 as a subroutine.

Lemma 4.3. *Let $A \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(A) = m$, $W = \ker(A)$, $\ell, u \in \mathbb{R}^n$, and let $x \in W$, $\ell \leq x \leq u$. Then, in $O(nm(n-m)^2 + nm^{\omega-1})$ time, we can find vectors $y, y^{(1)}, \ldots, y^{(t)} \in W$, $t \leq n$, a set $J \subseteq [n]$ and coefficients $\alpha_1, \ldots, \alpha_t \geq 0$, $\sum_{j=1}^{t} \alpha_j = 1$ such that*

(i) $\ell \leq y \leq u$ and $y_J = (\ell^+ - u^-)_J$, and

(ii) $y = \sum_{j=1}^{t} \alpha_j y^{(j)}$, and each $y^{(j)} \in W$ has minimal support subject to $y_J^{(j)} = (\ell^+ - u^-)_J$.

Proof. We repeatedly use the algorithm in Lemma 4.1 to transform x to such a vector y. Let

$$P = \{i \in [n] : \ell_i > 0 \ \text{ or } \ u_i < 0\}, \quad N = [n] \setminus P.$$

If $P = \emptyset$ we return $y = 0$ and $J = \emptyset$. Otherwise, we initialize $y = x$, and

$$J = \{i \in P : \ y_i = \ell_i > 0 \ \text{ or } \ y_i = u_i < 0\} \cup \{i \in N : \ y_i = 0\}. \tag{6}$$

The vector y and the set J will be iteratively updated, maintaining $y \in W$, $\ell \leq y \leq u$ and (6). Thus, property (i) is maintained throughout, since $y_J = (\ell^+ - u^-)_J$. Note that J must include all indices $i \in [n]$ with $\ell_i = u_i$.

In every iteration, we call the subroutine in Lemma 4.1 for y and J. If $z = 0$, then we terminate with the convex combination returned. Otherwise, we obtain $z \in W \setminus \{0\}$ that is sign-consistent with y, and $z_J = 0$. In the very first step, $J = \emptyset$ is possible, in which case the output will be $z = y$.

We now update y to the new value $y' = y - \alpha z$, where $\alpha \geq 0$ is chosen as the smallest value where either $y'_i = \ell_i > 0$, or $y'_i = u_i < 0$, or $y'_i = 0$ for some $i \notin J$. This α is finite, since there exists a coordinate $z_i \neq 0$, and z is sign-consistent with y.

This finishes the description of the algorithm. It is clear that the set J will be strictly increased in such an iteration, and hence, the number of calls to Lemma 4.1 is bounded by n, giving the total running time bound $O(nm(n - m)^2 + n^2 m^{\omega-1})$. The second term corresponds to computing a basis of A, which only has to be done a single time. So the runtime reduces to $O(nm(n - m)^2 + nm^{\omega-1})$. $\qquad\square$

The above algorithm directly corresponds to a constructive version of Theorem 3.1. The proof is analogous to the proof of Lemma 4.3.

Corollary 4.4. *Let $A \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(A) = m$, $W = \ker(A)$, $\ell, u \in \mathbb{R}^n$, $M \geq 1$, and let $x \in W$, $\ell \leq x \leq u$. Then, in $O(nm(n - m)^2 + nm^{\omega-1})$ time, we can either find a vector $y \in W$, such that $\ell \leq y \leq u$, and $\|y\|_\infty \leq M\|\ell^+ + u^-\|_1$, or find a lifting certificate of $M < \kappa_W$.*

5 Black-box linear programming solvers

Our feasibility and optimization algorithms in Sections 6 and 7 use oracles that return approximate LP solutions. These can be implemented by using any weakly-polynomial algorithm that returns approximately optimal approximately feasible solutions as in (APX-LP). We will use the following result that summarizes recent developments on interior point methods. Whereas the papers only formulate the main statements on primal solutions, they all use primal-dual interior-point methods, and also find dual solutions with similar properties. We present the results in such a primal-dual form.

Theorem 5.1 ([LS19, vdB20, vdBLSS20, JSWZ20]). *Consider (LP) for $A \in \mathbb{R}^{m \times n}$ with $\mathrm{rk}(A) = m$. Assume both the primal and dual programs are feasible, let $\delta \in [0, 1]$ and $d \in \mathbb{R}^n_+$ be such a feasible primal solution i.e. $Ad = b$. Let R_P and R_D be the diameters of the primal and dual solution sets in ℓ_2 norm, i.e., $\|x\|_2 \leq R_P$ for all primal feasible solutions x, and $\|s\|_2 \leq R_D$ for all dual feasible solutions (y, s). Then, we can find a vector $(x, y, s) \in \mathbb{R}^{n+m+n}$ with $x, s \geq 0$ such that*

(i) $\langle c, x \rangle \leq \langle b, y \rangle + \delta \cdot (\|c\|_2 \cdot R_P + \|d\|_2 \cdot R_D)$,

(ii) $\|Ax - b\|_2 \leq \delta \cdot (\|A\|_F \cdot R_P + \|b\|_2)$, and

(iii) $\|A^\top y + c - s\|_2 \leq \delta \cdot (\|A\|_F \cdot R_D + \|c\|_2)$.

in the following running time bounds:

(1) In $O((mn + m^3) \log^{O(1)}(n) \log(n/\delta))$ expected running time [vdBLSS20].

(2) In $O(n^{\omega+o(1)} \log(n/\delta))$ deterministic running time, assuming $\omega \geq 13/6$ [vdB20]. The same expected running time is achievable assuming $\omega \geq 2 + 1/18$ [JSWZ20].

(3) In $O\big((\mathrm{nnz}(A) + m^2)\sqrt{m} \log^{O(1)}(n) \log(n/\delta)\big)$ expected running time [LS19], where $\mathrm{nnz}(A)$ denotes the number of nonzero entries in A.

We use the notation $\Psi(A)$ to denote the 'cost per unit' in these results. Namely, a δ-approximate solution can be obtained in time $O(\Psi(A) \log(n/\delta))$, where

$$\Psi(A) \leq \log^{O(1)}(n) \min\{mn + m^3, n^{\omega+o(1)}, \sqrt{m}(\mathrm{nnz}(A) + m^2)\}. \tag{7}$$

We note that the third bound will not be directly applicable, since we will use the oracle in various subspaces, where the number of nonzero entries may increase.

We now state the main forms of feasibility and optimization oracles we use. In Section 8, we derive these results from Theorem 5.1, by running the algorithms on an extended system. The oracles used in Sections 6 and 7 can be implemented from Theorem 5.2 and Theorem 5.3.

The main difference is in the allowable violation of primal and dual feasibility. Theorem 5.1 returns $x, s \geq 0$, but they might not be in the right subspace: proximity bounds are given on $\|Ax - b\|_2$ and $\|A^\top y + c - s\|_2$. In contrast, our theorems require $Ax = b$ and $A^\top y + c - s$, but allow small negative components. For a suitable represented matrix, as assumed in the statements, we can easily convert the first type of violation into the second.

Theorem 5.2. *Let $A \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(A) = m$, $W = \ker(A)$, and $d \in \mathbb{R}^n$, and assume the matrix $A_B^{-1}A$ is provided for some $m \times m$ non-singular submatrix A_B. Let $0 < \varepsilon \leq 1/2$, and let $M \geq 2$ be an estimate of κ_W. Consider a linear feasibility problem $x \in W + d, x \geq 0$. There exists an algorithm that returns either of the following outcomes:*

(F1) A near-feasible solution $x \in W + d$, $\|x^-\|_1 \leq \varepsilon\|d/W\|_1$, $\|x\|_\infty \leq 2M\|d\|_1$.

(F2) A Farkas certificate of infeasibility: $s \in (W^\perp)_+$, $\langle d, s \rangle < 0$.

(F3) A lifting certificate of $M < \kappa_W$.

The running time is $O(\Psi(A) \log((M+n)/\varepsilon))$ for outcome (F1), and $O(\Psi(A) \log((M+n)/\varepsilon) + (n-m)m^2 + n^\omega)$ in outcomes (F2) and (F3).

Theorem 5.3. *Let $A \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(A) = m$, $W = \ker(A)$, and $d \in \mathbb{R}^n$, and assume the matrix $A_B^{-1}A$ is provided for some $m \times m$ non-singular submatrix A_B. Let $0 < \varepsilon \leq 1/2$, and let $M \geq 2$ be an estimate of κ_W. Consider a linear program of the form Primal-Dual(W, d, c). There exists an algorithm that returns either of the following outcomes:*

(M1) A pair of primal and dual near-feasible and near-optimal solutions $x \in W + d, s \in W^\perp + c$, that is,

$$\begin{aligned}
\|x^-\|_1 &\leq \varepsilon\|d/W\|_1, \\
\|s^-\|_1 &\leq \varepsilon\|c/W^\perp\|_1, \\
\|x \circ s\|_1 &\leq 5\varepsilon M\|d/W\|_1\|c/W^\perp\|_1, \\
\|x\|_\infty &\leq 2M\|d/W\|_1, \text{ and} \\
\|s\|_\infty &\leq 2M\|c/W^\perp\|_1
\end{aligned} \tag{8}$$

Further, if $\kappa_W \leq M$, and both primal and dual sides of Primal-Dual(W, d, c) are feasible, and the optimum value is OPT, then

$$\langle c, x \rangle - 5\varepsilon M\|d/W\|_1\|c/W^\perp\|_1 \leq OPT \leq \langle d, c - s \rangle + 5\varepsilon M\|d/W\|_1\|c/W^\perp\|_1. \tag{9}$$

19

(M2) A Farkas certificate of primal infeasibility: $s \in (W^\perp)_+$, $\langle d, s \rangle < 0$.

(M3) A Farkas certificate of dual infeasibility: $x \in W_+$, $\langle c, x \rangle < 0$.

(M4) A lifting certificate of $M < \kappa_W = \kappa_{W^\perp}$.

The running time is $O(\Psi(A) \log((M + n)/\varepsilon))$ for outcome (M1). For (M2), (M3) and (M4) the runtime is $O(\Psi(A) \log((M + n)/\varepsilon) + n^2 m + n^\omega)$ if $d = 0$ or $c = 0$ and $O(\Psi(A) \log((M + n)/\varepsilon) + n^3 m)$ if $c \neq 0$ and $d \neq 0$.

6 The feasibility algorithm

Given a matrix $A \in \mathbb{R}^{m \times n}$ and $d \in \mathbb{R}^n$, we let $W = \ker(A)$. In this section, we consider the feasibility problem $x \in W + d, x \geq 0$.

A key insight is to work with a stronger system, including a proximity constraint. According to Corollary 3.2, whenever the problem $x \in W + d$, $x \geq 0$ is feasible and $\kappa_W \leq M$, then the following system is also feasible. In fact, this would be true even with the stronger bound M instead of $16M^2 n$; we use this weaker bound to leave sufficient slack for the recursive argument. Note that if $d \geq 0$, then the only feasible solution is $x = d$.

$$
\begin{aligned}
x &\in W + d \\
\|x - d\|_\infty &\leq 16M^2 n \|d^-\|_1 \qquad\qquad \text{(Feas-LP(W, d, M))} \\
x &\geq 0.
\end{aligned}
$$

We use a black-box approach assuming an oracle that returns an approximately feasible solution. We will assume that an oracle *Prox-Feas-Solver*(W, d, M, ε) is given as in Oracle 1. Outcome (i) gives an approximately feasible solution with a bound on the negative components and a somewhat stronger proximity guarantee as in (Feas-LP(W, d, M)). Outcome (ii) gives a Farkas certificate of infeasibility, whereas outcome (iii) gives a lifting certificate of $M < \kappa_W$.

Input : A subspace $W \subseteq \mathbb{R}^n$, given as $W = \ker(A)$ for $A \in \mathbb{R}^{m \times n}$, a vector $d \in \mathbb{R}^n$, $M, \varepsilon > 0$.

Output: One of the following three outcomes

    (i) A solution x to the system

$$
\begin{aligned}
x &\in W + d \\
\|x - d\|_\infty &\leq 3M^2 n \|d^-\|_1 \qquad\qquad \text{(Prox-Feas(W, d, M, ε))} \\
\|x^-\|_\infty &\leq \varepsilon \|d^-\|_1
\end{aligned}
$$

    (ii) A vector $y \in W^\perp$, $y \geq 0$, $\langle d, y \rangle < 0$,

    (iii) A subset $I \subseteq [n]$ and a vector $p \in \pi_I(W)$ such that $\|L_I^W(p)\|_\infty > M \|p\|_1$.

Oracle 1: *Prox-Feas-Solver*(W, d, M, ε)

This oracle can be derived from Theorem 5.2, by finding an approximately feasible solution to a modification of the system $x \in W + d, x \geq 0$. The derivation is given in Section 8.3; the running time is stated as follows. Recall the definition of $\Psi(A)$ from (7). The running time is stated as follows. Recall the definition of $\Psi(A)$ from (7).

Lemma 6.1. *There exists an $O(\Psi(A) \cdot \log(M + n) + nm^{\omega - 1 + o(1)})$ time algorithm, that either returns a solution to Prox-Feas(W, d, M, ε), or concludes that (ii) or (iii) should be the outcome*

of Prox-Feas-Solver(W, d, M, ε). In the latter case, these outcomes can be obtained in additional time $O(nm^2 + n^{\omega+o(1)})$.

The next lemma will be the key technical tool of the algorithm. It allows to solve Feas-LP(W, d, M) by combining an approximate solution to Prox-Feas(W, d', M, ε) for some $d' \in W + d$ with an exact solution to Feas-LP(W, d, M) obtained recursively from a smaller system.

Recall that for a set $K \subseteq [n]$, cl(K) denotes the closure of K i.e., the unique largest set $J \subseteq [n]$ such that $K \subset J$ and rk(A_J) = rk(A_K).

We select a set K of indices i where x_i is very large in the approximate solution x; for such indices, proximity guarantees that there must be a feasible solution $x^* \in W + d$, $x^* \geq 0$ with $x_i^* > 0$. We project out all these indices, along with all other indices $J = \text{cl}(K) \setminus K$ in their closure, and recurse on the remaining index set I. We note that the purpose of the set J is to maintain property (3) for the recursive call, i.e. to avoid creating loops from the recursive instances.

The choice of the proximity bounds allow us to 'stitch together' the solution obtained on $\pi_I(W)$ from the recursive call with the approximate solution x to a feasible solution to the original system. Roughly speaking, the amount of change required to cancel out all negative coordinates in x_I is small enough so that x remains positive on K.

An important feature in the scheme is the choice of the vector d' for the approximate system. This will be either $d' = d$ or $d' = d/W$; hence $W + d' = W + d$. However, this choice makes a difference due to the proximity bounds: the system Feas-LP(W, d, M) features $\|d^-\|_1$ as well as a bound on $\|x - d\|_\infty$.

In particular, if $\|d^-\|_1$ is 'too big', then we may end up with an empty index set K and cannot recurse. In this case, we swap to $d' = d/W$; otherwise, we keep $d' = d$. We note that always swapping to $d' = d/W$ does not work either: Feas-LP(W, d, M) features the bound $\|x - d\|_\infty$, and using $\|x - d/W\|_\infty$ in the approximate system may move us too far from d. Fortunately, the bad cases for these two choices turn out to be complementary. We note that the distinguished role of d/W is due to the bound $\|x\|_2 \geq \|d/W\|_2$ for any $x \in W + d$. We formulate the following simple consequence:

Lemma 6.2. *For any subspace $W \subseteq \mathbb{R}^n$, and vectors $x, d \in \mathbb{R}^n$, $x \in W + d$, we have*

$$\|x\|_\infty \geq \frac{\|x\|_2}{\sqrt{n}} \geq \frac{\|d/W\|_1}{n} \, .$$

Proof. By definition of d/W, we have $\|x\|_2 \geq \|d/W\|_2$ for any $x \in W + d$. The claim follows combining this with the norm inequalities $\|x\|_\infty \geq \|x\|_2/\sqrt{n}$ and $\|d/W\|_2 \geq \|d/W\|_1/\sqrt{n}$. $\qquad\square$

Lemma 6.3. *Let $M, \varepsilon > 0$ such that $\varepsilon \leq 1/(16M^4n^4)$, let $d \in \mathbb{R}^n$ and define*

$$d' = \begin{cases} d & \text{if } \|d^-\|_1 < \max\left\{M\|d/W\|_1, \frac{\|d\|_\infty}{4M^2n}\right\}, \\ d/W & \text{otherwise.} \end{cases}$$

Let x be a feasible solution to Prox-Feas(W, d', M, ε), and let

$$\begin{aligned} K &= \left\{i \in [n] \colon x_i \geq 16n^2M^3\|x^-\|_1\right\}, \\ J &= \text{cl}(K) \setminus K, \\ I &= [n] \setminus \text{cl}(K). \end{aligned} \tag{10}$$

Then $K \neq \emptyset$. Further, if Feas-LP($\pi_I(W), x_I, M$) is feasible, then let $w \in \mathbb{R}^I$ be any feasible solution, and let

$$x' = x + L_{I \cup J}^W\big((w - x_I, x_J^-)\big).$$

Then, either x' is feasible to Feas-LP(W, d, M), or $\|L^W_{I \cup J}((w - x_I, x^-_J))\|_\infty > M\|(w - x_I, x^-_J)\|_1$, that is, $M < \kappa_W$.

If $y \in \pi_I(W)^\perp$, $y \geq 0$, $\langle x_I, y \rangle < 0$ is an infeasibility certificate to $x' \in \pi_I(W) + x_I$, $x_I \geq 0$, then $y' = (y, 0_{J \cup K})$ is an infeasibility certificate to $x \in W + d$, $x \geq 0$.

Proof. We first show that x' is feasible to Feas-LP(W, d, M). Then, we verify that $K \neq \emptyset$.

Feasibility of x': Let us first show that x' is well-defined; this needs that $(w - x_I, x^-_J) \in \pi_{I \cup J}(W)$. By definition, $w \in \pi_I(W) + x_I$ means that $w = \hat{w}_I$ for some $\hat{w} \in W + x$. Clearly, $w - x_I = (\hat{w} - x)_I \in \pi_I(W)$. By the definition of J, $(0_I, z) \in \pi_{I \cup J}(W)$ for any $z \in \mathbb{R}^J$. Thus, $(w - x_I, z') \in \pi_{I \cup J}(W)$ for any $z' \in \mathbb{R}^J$.

The containment $x' \in W + d$ is immediate, since $L^W_{I \cup J}((w - x_I, x^-_J)) \in W$, and $W + x = W + d$. For the rest of the proof, assume that $\|L^W_{I \cup J}((w - x_I, x^-_J))\|_\infty \leq M\|(w - x_I, x^-_J)\|_1$.

Let us verify $x' \geq 0$. By definition, if $i \in I$, then the corresponding coordinate of $L^W_{I \cup J}((w - x_I, x^-_J))$ equals $w_i - x_i$, and thus $x'_i = w_i \geq 0$. Analogously, for $j \in J$, the corresponding coordinate of $L^W_{I \cup J}((w - x_I, x^-_J))$ equals x^-_j and so $x'_j = x_j + x^-_j \geq 0$. For $k \in K$, we have $x'_k \geq x_k - \|L^W_{I \cup J}((w - x_I, x^-_J))\|_\infty$. By definition of K, $x_k \geq 16n^2 M^3 \|x^-_{I \cup J}\|_1$. Then, $x'_k \geq 0$ follows as

$$\|L^W_{I \cup J}((w - x_I, x^-_J))\|_\infty \leq M\|(w - x_I, x^-_J)\|_1 \leq nM\|(w - x_I, x^-_J)\|_\infty \leq 16n^2 M^3 \|x^-_{I \cup J}\|_1 \quad (11)$$

The first inequality is by the assumption we made; the second inequality estimates the 1-norm by the ∞-norm; the third inequality is since w is feasible to Feas-LP$(\pi_I(W), x_I, M)$.

To complete the proof that x' is feasible to Feas-LP(W, d, M), it remains to verify the proximity bound $\|x' - d\|_\infty \leq 16M^2 n\|d^-\|_1$. First, we need an auxiliary claim.

Claim 6.3.1. $\|x - d\|_\infty \leq 8M^2 n\|d^-\|_1$ and $\|x^-\|_1 \leq n\varepsilon\|d^-\|_1$.

Proof of Claim. If $d' = d$, then from the feasibility of x to Prox-Feas(W, d', M), we have $\|x - d\|_\infty \leq 3M^2 n\|d^-\|_1$ and $\|x^-\|_1 \leq n\|x^-\|_\infty \leq n\varepsilon\|d^-\|_1$. If $d' \neq d$, then $d' = d/W$, $\|d^-\|_1 > M\|d/W\|_1$ and $\|d^-\|_1 > \|d\|_\infty/(4M^2 n)$ and so

$$\|x - d\|_\infty \leq \|x - d/W\|_\infty + \|d/W\|_\infty + \|d\|_\infty \leq 3M^2 n\|(d/W)^-\|_1 + \|d/W\|_\infty + 4M^2 n\|d^-\|_1$$
$$\leq (3M^2 n + 1)\|d/W\|_1 + 4M^2 n\|d^-\|_1 \leq (3M^2 n + 1)M^{-1}\|d^-\|_1 + 4M^2 n\|d^-\|_1$$
$$\leq 8M^2 n\|d^-\|_1, \quad \text{and}$$
$$\|x^-\|_1 \leq n\|x^-\|_\infty \leq n\varepsilon\|(d')^-\|_1 = n\varepsilon\|(d/W)^-\|_1 \leq n\varepsilon M^{-1}\|d^-\|_1$$

proving the claim. ∎

Using Claim 6.3.1 as well as the bound in (11), we see that

$$\|x' - d\|_\infty \leq \|x - d\|_\infty + \|L^W_{I \cup J}((w - x_I, x^-_J))\|_\infty \leq 8M^2 n\|d^-\|_1 + 16n^2 M^3 \|x^-_{I \cup J}\|_1 \leq 16M^2 n\|d^-\|_1.$$

Recursion on smaller subset: We show that $K \neq \emptyset$. If $d = d'$ then either $\|d^-\|_1 < M\|d/W\|_1$ or $\|d^-\|_1 < \|d\|_\infty/(4M^2 n)$. If $\|d^-\|_1 < M\|d/W\|_1$ then, using Lemma 6.2,

$$\|x\|_\infty \geq \frac{\|d/W\|_1}{n} > \frac{\|d^-\|_1}{Mn} \geq \frac{\|x^-\|_\infty}{\varepsilon Mn} \geq \frac{\|x^-\|_1}{\varepsilon Mn^2} \geq 16n^2 M^3 \|x^-\|_1,$$

and if $\|d^-\|_1 < \|d\|_\infty/(4M^2 n)$, then $\|x - d\|_\infty \leq 3M^2 n\|d^-\|_1$ by the call to Prox-Feas(W, d', M) and so

$$\|x\|_\infty \geq \|d\|_\infty - \|x - d\|_\infty > 4M^2 n\|d^-\|_1 - 3M^2 n\|d^-\|_1 = M^2 n\|d^-\|_1 \geq \frac{M^2 n}{\varepsilon n}\|x^-\|_1 \geq 16n^2 M^3 \|x^-\|_1.$$

The remaining case is $d' = d/W$. Again by Lemma 6.2,

$$\|x\|_\infty \geq \frac{\|d/W\|_1}{n} \geq \frac{\|x^-\|_\infty}{\varepsilon n} \geq \frac{\|x^-\|_1}{\varepsilon n^2} \geq 16n^2 M^3 \|x^-\|_1.$$

Infeasibility certificate: Consider now the case when we have an infeasibility certificate $y \in \pi_I(W)^\perp$, $y \geq 0$, $\langle x_I, y \rangle < 0$ to the system $x' \in \pi_I(W) + x_I$, $x_I \geq 0$. Recall that $\pi_I(W)^\perp = (W^\perp)_I$, that is, $y' = (y, 0_{J \cup K}) \in W^\perp$. Clearly, $y' \geq 0$, and $\langle d, y' \rangle = \langle x, y' \rangle = \langle x_I, y \rangle < 0$. In the first equality, we used that $x \in W + d$ and $y' \in W^\perp$. □

Algorithm 1: FEASIBILITY-ALGORITHM

    Input   : $A \in \mathbb{R}^{m \times n}$, $W = \ker(A) \subseteq \mathbb{R}^n$, $\mathrm{rk}(A) = m$ satisfying (3), $d \in \mathbb{R}^n$, $M \geq 2$.
    Output: One of the following: *(i)* a solution x to *Feas-LP(W, d, M)*; *(ii)* a Farkas
                certificate of infeasibility, or *(iii)* a lifting certificate that $M < \kappa_W$.

1 **if** $\|d^-\|_1 \geq \max\left\{M\|d/W\|_1, \|d\|_\infty/4M^2n\right\}$ **then**
2 | $d \leftarrow d/W$;

3 **if** $d \geq 0$ **then**
4 | **return** *solution d*

5 $\varepsilon \leftarrow 1/(2Mn)^4$;
6 **switch** *Prox-Feas-Solver(W, d, M, ε)* **do**
7 | **case** *(i)*
8 | | $x \leftarrow$ *Prox-Feas-Solver(W, d, M, ε)* ;
9 | | $K \leftarrow \{i \in [n] : x_i \geq 16n^2M^3\|x^-\|_1\}$;
10 | | $J \leftarrow \mathrm{cl}(K) \setminus K$;
11 | | $I \leftarrow [n] \setminus (J \cup K)$;
12 | | **switch** *output of* FEASIBILITY-ALGORITHM$(\pi_I(W), d_I, M)$ **do**
13 | | | **case** *(i)*
14 | | | | $x' \leftarrow$ FEASIBILITY-ALGORITHM$(\pi_I(W), d_I, M)$;
15 | | | | $w \leftarrow L_{I \cup J}^W(x' - x_I, x_J^-)$;
16 | | | | **if** $\|w\|_\infty \leq M\|(x' - x_I, x_J^-)\|_1$ **then**
17 | | | | | **return** $x + w$
18 | | | | **else**
19 | | | | | **return** *lifting certificate of $M < \kappa_W$.*
20 | | | **case** *(ii)*
21 | | | | $y \leftarrow$ FEASIBILITY-ALGORITHM$(\pi_I(W), d_I, M)$;
22 | | | | $y' \leftarrow (y, 0_{J \cup K})$;
23 | | | | **return** *Farkas certificate y' to Feas-LP(W, d, M)*
24 | | | **case** *(iii)*
25 | | | | **return** *lifting certificate of $M < \kappa_W$*
26 | **case** *(ii)*
27 | | **return** *Farkas certificate*
28 | **case** *(iii)*
29 | | **return** *lift. certificate of $M < \kappa_W$*

The overall feasibility algorithm is given in Algorithm 1, a recursive implementation of Lemma 6.3. The output can be *(i)* a feasible solution to *Feas-LP(W, d, M)*; *(ii)* a Farkas certificate of infeasibility, or *(iii)* a lifting certificate of $M < \kappa_W$. The latter will always be of the form of an index set $I \subseteq [n]$ and a vector $p \in \pi_I(W)$ such that $\|L_I^W(p)\|_\infty > M\|p\|_1$. In this case, we can restart the entire algorithm, after updating M to $\max\{\|L_I^W(p)\|_\infty/\|p\|_1, M^2\}$.

The algorithm calls Oracle 1. For outputs *(ii)* and *(iii)*, we return the Farkas certificate or the lifting certificate for $M < \kappa_W$. For output *(i)*, we construct the sets I, J, and K and recurse on $\pi_I(W)$, as in Lemma 6.3.

We are now ready to state the central theorem of this section, which in particular proves

Theorem 1.1.

Theorem 6.4. *Algorithm 1 is correct. If $M > \kappa_W$ and the system $x \in W + d, x \geq 0$ is feasible, then the algorithm returns a solution in time $O(m \, \Psi(A) \log(M + n) + mn^{\omega + o(1)})$. If the system is infeasible or $M < \kappa_W$, then a Farkas certificate or a lifting certificate can be obtained in additional running time $O(nm^2 + n^{\omega + o(1)})$.*

Proof. The bulk of of the correctness proof follows by Lemma 6.3. We prove correctness by induction on the number of variables. For $n = 1$ the statement is trivial. Assume we have already proved correctness for $< n$ variables; we now prove it for n variables.

The recursive step calls the algorithm for $\pi_I(W)$; this has fewer than n variables as Lemma 6.3 shows that $K \neq \emptyset$. By induction hypothesis, this recursive call returns a solution to *Feas-LP*(W, d, M) which can be lifted to a feasible solution in line 17 of Algorithm 1 by Lemma 6.3, unless we conclude in time $O(nm^2 + n^\omega)$ with either a Farkas certificate or $M < \kappa_W$ by Lemma 6.1. The second part of the same lemma asserts that a Farkas certificate to the smaller system can be lifted to a Farkas certificate of the original system.

Also note that by Remark 2.10 we are able to transfer a lifting certificate for a subspace into one for W.

Let us show that the required property (3) holds in every recursive call. Indeed, assume that some $i \in I$ forms a loop in $\pi_I(W)$. Then, there is a circuit $C \in \mathcal{C}_W$ such that $C \cap I = \{i\}$. Consequently, $\mathrm{rk}(K \cup J \cup \{i\}) = \mathrm{rk}(K \cup J)$, contradicting the choice of $J \cup K = \mathrm{cl}(K)$. (Note that the linear matroid on $\pi_I(W)$ corresponds to the contraction of the set $K \cup J$ in the linear matroid of W.)

We next show that in every recursive call, $\dim(W^\perp) > \dim(\pi_I(W)^\perp)$. Consequently, the total number of recursive calls to Algorithm 1, and thus, to *Prox-Feas-Solver*(W, d, M, ε) can be bounded by $m = n - \dim(W) = \dim(W^\perp)$.

For a contradiction, assume that $\dim(W^\perp) = \dim(\pi_I(W)^\perp)$. Recall that $\pi_I(W)^\perp = (W^\perp)_I$. Thus, $W^\perp = \{0\}^{J \cup K} \times (W^\perp)_I$ and therefore $W = \mathbb{R}^{K \cup J} \times \pi_I(W)$, which by $K \neq \emptyset$ contradicts the assumption (3) on W.

The running time bounds are dominated by the running time of the oracle calls as stated in Lemma 6.1 as well as the additional linear algebra. These calculations are to identify the index sets J, the projections $\pi_I(W)$ and computing the lifting vector as in 17. In particular, we need to maintain throughout $W = \ker(A)$ along with the form $A_B^{-1} A$, as required in Theorem 5.3.

We can maintain such a representation as follows. Let A denote the original matrix. Initially, we perform a Gaussian elimination to obtain a basis $B \subseteq [n]$ and the form $A' = A_B^{-1} A$ in time $O(m^2 n)$. At any point of the algorithm, let $H \subseteq [n]$ denote the current index subset; we let $W^H \subseteq \mathbb{R}^H$ denote the current subspace. We also maintain the basis $B \subseteq H$, and the form $A' = A_B^{-1} A_H$; thus, $W^H = \ker(A'_H)$. At every iteration, we need to partition $H = I \cup J \cup K$, and recurse to $\pi_I(W^H)$. Identifying K is straightforward, but finding J requires pivot operations. First, we use row operations to exchange a maximal number of columns in K into B. Now, $\mathrm{cl}(K)$ is given by $K \cap B$ along with the columns generated by them in the current form; we set $J = \mathrm{cl}(K) \setminus K$ and let $I = H \setminus \mathrm{cl}(K)$.

At this point, the submatrix $A'_{(I \cap B) \times I}$ represents $\pi_I(W^H)$. Here, the rows $I \cap B$ correspond in abuse of notation to the columns of B that are in I of the identity submatrix of A'. Hence, we can charge to every removed column at most two basis exchanges, giving a total of $O(n^2 m)$ for these operations.

It is left to compute the lift from line 15, which recursively has to be performed m times and takes $O(\min\{m^2 n, n^\omega\})$ by Lemma 2.3. $\qquad\square$

7 The optimization algorithm

In this section, we show how Primal-Dual(W, d, c) can be solved using an approximate LP solver. As in the feasibility algorithm, we let M denote our current upper estimate on $2\kappa_W$. We present an algorithm that comprises an Inner and an Outer Loop. The calls to the approximate LP solver will happen inside the Inner Loop.

The outer loop gives an algorithmic implementation of Theorem 3.6. The subroutine INNER-LOOP(W, d, c, M) returns a solution $(\tilde{d}, \tilde{x}, s)$, where $\tilde{d}$ is a 'perturbed' version of d, and $(\tilde{x}, s)$ are optimal solutions to Primal-Dual($W, \tilde{d}, c$). We get $(\tilde{d}, \tilde{x}, \tilde{s})$ as solutions to the following system.

$$\|\tilde{d} - d\|_1 \leq \frac{\|\tilde{x}\|_\infty}{4n^2 M^2}$$
$$\tilde{x} \in W + \tilde{d}$$
$$\tilde{s} \in W^\perp + c \qquad \qquad (\text{F-Primal}(W, d, c, M))$$
$$\langle \tilde{x}, \tilde{s} \rangle = 0$$
$$\tilde{x}, \tilde{s} \geq 0.$$

The subroutine will be described in Section 7.2; we now state the running time.

Theorem 7.1. *Assume we are given a matrix $A \in \mathbb{R}^{m \times n}$, vectors $c \in \mathbb{R}^n$ and $d \in \mathbb{R}^n_+$; let $W = \ker(A)$, and M be an estimate on κ_W. There exists an $O(n \Psi(A) \log(M + n) + n^{\omega+1+o(1)})$ time algorithm (Algorithm 3) that returns a solution $(\tilde{d}, \tilde{x}, \tilde{s})$ to F-Primal(W, d, c, M), or decides that the system is either primally infeasible or $M < \kappa_W$. To obtain a Farkas certificate of primal infeasibility or a certificate that $M < \kappa_W$ is obtained in additional time $O(n^3 m)$.*

The overall algorithm described in Section 7.1 repeatedly calls INNERLOOP to set primal and dual variables to 0 according to Theorem 3.6, and recurses to lower dimensional subspaces. The final optimal solutions are obtained via calling the feasibility algorithm on both primal and dual side. The drawback of this variant is that, in case the algorithm fails, we do not obtain lifting certificates of $M < \kappa_W$—a guarantee we can achieve for feasibility in Section 6. A post-processing, as described in Lemma 7.5 is able to create a certificate of $M < \kappa_W$, at an additional computational expense.

7.1 The Outer Loop

Consider an instance of Primal-Dual(W, d, c) and an estimate M on κ_W. We first use the feasibility algorithm and check if both systems $x \in W + d$, $x \geq 0$ and $s \in W^\perp + c$, $s \geq 0$ are feasible. For the remainder of this section, let us assume both these systems are feasible, and consequently, Primal-Dual(W, d, c) is also feasible. Moreover, we can write an equivalent system with nonnegative $d \geq 0$. (We could also impose $c \geq 0$, but this will not be used).

The overall algorithm is presented in Algorithm 2. We let (W^0, d^0, c^0) denote the original input, where W^0 is a subspace of $\mathbb{R}^n$, and $d^0, c^0 \in \mathbb{R}^n$, $d^0 \geq 0$. We will maintain an index set $I \subseteq [n]$, initialized as $I = [n]$. We gradually move every index into the set B or N. We apply Theorem 3.6 with thresholds $\tau = \|x\|_\infty / (3n^2 M)$ and $T = \|x\|_\infty / n$. The bound $\|\tilde{d} - d\|_1 \leq \|\tilde{x}\|_\infty / 4n^2 M^2$ in F-Primal(W, d, c, M) guarantees that these are suitable choices. Assuming that $M \geq \kappa_W$, we obtain $s_B^* = 0$ for every dual optimal solution s^*, and that there exists a primal optimal solution x^* with $x_N^* = 0$. Note the asymmetry between the two sides; the weaker guarantee on N already suffices for correctness.

At every iteration, we have an index set $I \subseteq [n]$ of 'undecided indices' and a subspace $W \subseteq \mathbb{R}^I$. We consider the partition $I = I_L \cup I_M \cup I_S^+ \cup I_S^0$ according to Theorem 3.6, and add $I_L \cup I_M$ to B and I_S^0 to N. The optimal solution x'' guaranteed in the theorem implies that the optimum value is the same on W and $W' = W \cap \mathbb{R}^I_{I_L \cup I_S \cup I_S^+}$, i.e. the subspace where all entries

in I_S^0 are forced to 0. We then update $I = I_S^+$ as the remaining set of undecided indices and recurse on the subspace $\pi_I(W')$.

The algorithm terminates when d is contained in W. The remaining indices I are split up between B and N based on whether they are in the support of the optimal dual solution of the perturbed system. Finally, we obtain the primal and dual solutions by solving feasibility problems on the subsets B and N. If both are feasible, they form a complementary pair of primal and dual solutions, and hence they are optimal. In case of a failure, we conclude that the underlying assumption $M \geq \kappa_W$ was wrong.

Algorithm 2: Optimization Algorithm

> **Input** : $W^0 \subseteq \mathbb{R}^n$; $c^0 \in \mathbb{R}^n$, $d^0 \in \mathbb{R}^n_+$ such that $Primal\text{-}Dual(W^0, d^0, c^0)$ is feasible, and $M \geq 2$.
>
> **Output**: Solution (x^*, s^*) to $Primal\text{-}Dual(W^0, d^0, c^0)$ or the answer $M < \kappa_{W^0}$.
>
> **1** $W \leftarrow W^0$; $d \leftarrow d^0$; $c \leftarrow c^0/W^\perp$;
>
> **2** $I \leftarrow [n]$; $B \leftarrow \emptyset$; $N \leftarrow \emptyset$;
>
> **3 while** $I \neq \emptyset$ and $d \notin W$ **do**
>
> **4** $\quad$ **if** INNERLOOP(W, d, c, M) returns $M < \kappa_W$ **then**
>
> **5** $\quad\quad$ **return** $M < \kappa_W$.
>
> **6** $\quad$ $(\tilde{d}, \tilde{x}, \tilde{s}) \leftarrow$ INNERLOOP(W, d, c, M) ;
>
> **7** $\quad$ $I_L \leftarrow \{i \in I : \tilde{x}_i > \|\tilde{x}\|_\infty/n\}$;
>
> **8** $\quad$ $I_M \leftarrow \{i \in I : \|\tilde{x}\|_\infty/n \geq \tilde{x}_i > \|\tilde{x}\|_\infty/(3n^2 M)\}$;
>
> **9** $\quad$ $I_S \leftarrow \{i \in I : \|\tilde{x}\|_\infty/(3n^2 M) \geq \tilde{x}_i\}$;
>
> **10** $\quad$ $I_S^0 \leftarrow I_S \cap \mathrm{cl}(I_L)$; $I_S^+ \leftarrow I_S \setminus \mathrm{cl}(I_L)$;
>
> **11** $\quad$ $B \leftarrow B \cup I_L \cup I_M$;
>
> **12** $\quad$ $N \leftarrow N \cup I_S^0$;
>
> **13** $\quad$ $W' \leftarrow W \cap \mathbb{R}^I_{I_L \cup I_M \cup I_S^+}$;
>
> **14** $\quad$ $I \leftarrow I_S^+$;
>
> **15** $\quad$ $W \leftarrow \pi_I(W')$; $d \leftarrow d_I$; $c \leftarrow \tilde{s}_I$;
>
> **16** $N \leftarrow N \cup \big(I \cap \mathrm{supp}(\tilde{s})\big)$;
>
> **17** $B \leftarrow B \cup \big(I \setminus \mathrm{supp}(\tilde{s})\big)$;
>
> **18 if** FEASIBILITY-ALGORITHM(W^0_B, d^0) and FEASIBILITY-ALGORITHM$((W^0)^\perp_N, c^0)$ are feasible **then**
>
> **19** $\quad$ $x^* \leftarrow$ FEASIBILITY-ALGORITHM(W^0_B, d^0) ;
>
> **20** $\quad$ $s^* \leftarrow$ FEASIBILITY-ALGORITHM$((W^0)^\perp_N, c^0)$;
>
> **21** $\quad$ **return** (x^*, s^*)
>
> **22 else**
>
> **23** $\quad$ **return** $M < \kappa_{W^0}$

Theorem 7.2. *Assuming that* Primal-Dual(W^0, d^0, c^0) *is both primal and dual feasible, Algorithm 2 either finds an optimal solution to* Primal-Dual(W^0, d^0, c^0) *or correctly concludes that $M < \kappa_{W^0}$ by at most m calls to the subroutine* INNERLOOP. *The total runtime is $O(mn\,\Psi(A)\log(M + n) + mn^{\omega+1+o(1)})$ to find a primal-dual optimal pair. Obtaining a lifting certificate requires additional time $O(n^3 m^2)$.*

Proof. Let us assume that $M \geq \kappa_{W^0}$. The correctness of constructing the sets B and N follows from Theorem 3.6. Clearly, $I_L \neq \emptyset$.

The bound of at most m recursive calls can be established with the same argumentation as in the proof of Theorem 6.4 for the feasibility algorithm, by showing that $\dim(W^\perp)$ decreases by at least one in every step. Assume $\dim(\pi_{I_S^+}(W')^\perp) = \dim(W^\perp)$. Note that $\pi_{I_S^+}(W')^\perp =$

$\pi_{I_S^+}(W)^\perp = (W^\perp)_{I_S^+}$ and and so $\dim(W^\perp) = \dim((W^\perp)_{I_S^+})$. But then $W = \mathbb{R}^{I \setminus I_S^+} \times \pi_{I_S^+}(W)$, a contradiction as $I \setminus I_S^+ \neq \emptyset$ and assumption (3) on W holds. Also, assumption (3) holds for $\pi_{I_S^+}(W')$ as $\mathrm{cl}(K) \cap I_S^+ = \emptyset$.

The running time bound then follows from Theorem 7.1, noting that the computational cost is dominated by the calls to INNERLOOP and recomputing the representation of the subspace W. This can be done similarly as in the proof of Theorem 6.4. The claim on the lifting certificates will be proved in Lemma 7.5. $\qquad\square$

By proving Theorem 7.2, we also proved the special case Theorem 1.2.

7.2 The Inner Loop

For the Inner Loop and $d \geq 0$ we formulate the stronger version $\textit{F-Primal-Prox}(W, d, c, M, \varepsilon)$ of $\textit{F-Primal}(W, d, c, M)$, which maintains dual proximity and therefore—in similar vein as the feasibility algorithm—only needs an oracle with precision $(Mn)^{-O(1)}$ for recursive calls.

$$
\begin{aligned}
& x \in W + d \\
& s \in W^\perp + c \\
& \|s - c\|_\infty \leq 16M^2 n \|c_{\Lambda(c,d)}\|_1 \qquad (\textit{F-Primal-Prox}(W, d, c, M, \varepsilon)) \\
& \|x_{\Lambda(x,s)}\|_\infty \leq 2\varepsilon n \|x\|_\infty \\
& s \geq 0
\end{aligned}
$$

Given an output (x, s) to this system with $\varepsilon \leq 1/(32M^4 n^4)$, we obtain a solution to $\textit{F-Primal}(W, d, c, M)$ as

$$
\tilde{x}_i = \begin{cases} 0 & \text{if } i \in \Lambda(x,s), \\ x_i & \text{otherwise,} \end{cases} \quad , \quad \tilde{d} = d - x + \tilde{x}, \quad \tilde{s} = s. \tag{12}
$$

We will assume that the following oracle (Oracle 2) is available, that returns a solution $(\tilde{c}, x, s)$ to the system $\textit{Prox-Opt}(W, d, c, M, \varepsilon)$, a primal or dual infeasibility certificate, or a lifting certificate. For the input, we require the nonnegativity $c \geq 0$. This will be perturbed to $\tilde{c}$, and (x, s) will be near-optimal and near-feasible primal and dual solutions with respect to the perturbed system, satisfying a primal proximity constraint.

The implementation of Oracle 2 is given in Section 8.3, using Theorem 5.3.

Lemma 7.3. *Assume we are given a matrix $A \in \mathbb{R}^{m \times n}$, vectors $c \in \mathbb{R}^n_+$ and $d \in \mathbb{R}^n$; let $W = \ker(A)$, and M be an estimate on κ_W. Further, let $0 < \varepsilon < 1$. There exists an $O(\Psi(A) \cdot \log(M + n) + nm^{\omega - 1 + o(1)})$ time algorithm, that either returns a solution to $\textit{Prox-Opt}(W, d, c, M, \varepsilon)$ or concludes that (ii), (iii) or (iv) should be the outcome of $\textit{Prox-Opt-Solver}(W, d, c, M, \varepsilon)$. These latter outcomes require an additional computational time $O(n^3 m)$.*

INNERLOOP$(W, d, c, M, \varepsilon)$ recursively calls itself and Oracle 2, while maintaining dual proximity. The Oracle will be called for the dual system.

For convenience we now state it with primal and dual side flipped:

$$
\begin{aligned}
& s \in W^\perp + c \\
& x \in W + \tilde{d} \\
& \|s_{\Lambda(s,x)}\|_\infty \leq \varepsilon \|c_{\Lambda(c,d)}\|_1 \\
& \|s - c\|_\infty \leq 3M^2 n \|c_{\Lambda(c,d)}\|_1 \qquad (\textit{Prox-Opt}(W^\perp, c, d, M, \varepsilon)) \\
& \|d - \tilde{d}\|_\infty \leq \frac{\varepsilon}{n} \|d/W\|_1 \\
& d - \tilde{d} \geq 0 \\
& x \geq 0
\end{aligned}
$$

27

Input : $W \subseteq \mathbb{R}^n$, $c \in \mathbb{R}^n_+$, $d \in \mathbb{R}^n$, $M \geq 2, \varepsilon > 0$.

Output: One of the following:

(i) A solution to the system

$$x \in W + d$$
$$s \in W^\perp + \tilde{c}$$
$$\|x_{\Lambda(x,s)}\|_\infty \leq \varepsilon \|d_{\Lambda(d,c)}\|_1$$
$$\|x - d\|_\infty \leq 3M^2 n \|d_{\Lambda(d,c)}\|_1 \qquad (\textit{Prox-Opt}(W, d, c, M, \varepsilon))$$
$$\|c - \tilde{c}\|_\infty \leq \frac{\varepsilon}{n} \|c/W^\perp\|_1$$
$$c - \tilde{c} \geq 0$$
$$s \geq 0$$

(ii) A vector $y \in W^\perp$, $y \geq 0$, $\langle d, y \rangle < 0$,

(iii) A vector $x \in W$, $x \geq 0$, $\langle c, x \rangle < 0$,

(iv) A lifting certificate of $M < \kappa_W = \kappa_{W^\perp}$.

Oracle 2: *Prox-Opt-Solver*$(W, d, c, M, \varepsilon)$

Lemma 7.4. *Let M be an estimate on κ_W, $0 < \varepsilon < 1/(32M^4 n^4)$. Let us define*

$$c' = \begin{cases} c & \text{if } \|c_{\Lambda(c,d)}\|_1 < \max\left\{ M\|c/W^\perp\|_1, \frac{\|c\|_\infty}{4M^2 n} \right\}, \\ c/W^\perp & \text{otherwise.} \end{cases}$$

Let (x, s) be a feasible solution to Prox-Opt$(W^\perp, c', d, M, \varepsilon)$ *and let*

$$I = \{i \in [n] : s_i \leq 16n^3 M^3 \|s_{\Lambda(s,x)}\|_1\}, \quad J = [n] \setminus I.$$

Then, the following hold:

(i) If $I = \emptyset$, then we must have $s \geq 0$ and $d \in W$, and $(0, s)$ is feasible to F-Primal-Prox$(W, d, c, M, \varepsilon)$.

(ii) If $I \neq \emptyset$, then let (w, z) be a solution to F-Primal-Prox$(W_I, x_I, s_I, M, \varepsilon)$, *and define*

$$\tilde{x} = (w, x_J) + d - \tilde{d}, \quad \text{and} \quad \tilde{s} = s + L_I^{W^\perp}(z - s_I).$$

Then either $(\tilde{x}, \tilde{s})$ is feasible to F-Primal-Prox$(W, d, c, M, \varepsilon)$ *or we obtain a lifting certificate of $M < \kappa_{W^\perp} = \kappa_W$.*

(iii) $J \neq \emptyset$.

Proof. Let us start with two simple claims.

Claim 7.4.1. $x_J = 0$.

Proof of Claim. The vector x is required to be nonnegative. Assume that for some $j \in J$, $x_j > 0$, and thus $j \in \Lambda(s, x)$. For such an index, $s_j > 16n^3 M^3 \|s_{\Lambda(s,x)}\|_1 \geq 16n^3 M^3 s_j$, a clear contradiction. ∎

Claim 7.4.2. $\|s - c\|_\infty \leq 8M^2n\|c_{\Lambda(c,d)}\|_1$.

Proof of Claim. If $c' = c$, then the stronger bound with coefficient $3M^2n$ is included in *Prox-Opt$(W^\perp, c', d, M, \varepsilon)$*. Assume $c' = c/W^\perp$. Then,

$$
\begin{aligned}
\|s - c\|_\infty &\leq \|s - c/W^\perp\|_\infty + \|c/W^\perp\|_\infty + \|c\|_\infty \\
&\leq 3M^2n\|c/W^\perp\|_1 + \|c/W^\perp\|_1 + \|c\|_\infty \\
&\leq \frac{3M^2n + 1}{M}\|c_{\Lambda(c,d)}\|_1 + 4M^2n\|c_{\Lambda(c,d)}\|_1 \\
&\leq 8M^2n\|c_{\Lambda(c,d)}\|_1 \,.
\end{aligned}
$$

The second inequality used the bound in $\|s - c'\|_\infty$ in the proximal solver, upper bounding the left hand side as $3M^2n\|c'\|_1$. The third inequality uses the bounds on $\|c/W^\perp\|$ and $\|c\|_\infty$ in the case when $c' = c/W^\perp$ is chosen. ∎

Part (i) Assume $I = \emptyset$ and $J = [n]$. Then, for every $j \in [n]$, $s_j > 0$ by definition, and $x_j = 0$ by Claim 7.4.1. Since $x = 0 \in W + \tilde{d}$, we have $\tilde{d} \in W$, and therefore $d - \tilde{d} \in W + d$. By Lemma 6.2, $\|d - \tilde{d}\|_\infty \geq \|d/W\|_1/n$. On the other hand, we have the upper bound $\|d - \tilde{d}\|_\infty \leq \varepsilon\|d/W\|_1/n$. This is only possible if $d/W = 0$, that is, $d \in W$. Using also Claim 7.4.2, we conclude that $(0, s)$ is feasible to *F-Primal-Prox$(W, d, c, M, \varepsilon)$*.

Part (ii) The vector $\tilde{s}$ is well-defined, since $z - s_I \in (W_I)^\perp = \pi_I(W^\perp)$. We have $\tilde{s} \in W^\perp + s = W^\perp + c' = W^\perp + c$ by definition. Further, $(w, 0_J) \in W + (x_I, 0_J)$, and therefore, $(w, x_J) \in W + x = W + \tilde{d}$. Then, $(w, x_J) - \tilde{d} + d \in W + d$.

Let us assume that $\|L_I^{W^\perp}(z - s_I)\|_\infty \leq M\|z - s_I\|_1$; otherwise, we can return a lifting certificate of $M < \kappa_{W^\perp}$. Thus,

$$
\|L_I^{W^\perp}(z - s_I)\|_\infty \leq M\|z - s_I\|_1 \leq 16M^3n^2\|s_{\Lambda(s_I, x_I)}\|_1 \leq 16M^3n^3\|s_{\Lambda(s,x)}\|_\infty, \quad (13)
$$

where the second inequality comes from the bound on $\|z - s_I\|_1$ in *F-Primal-Prox$(W_I, x'_I, s_I, M, \varepsilon)$*. We are ready to show $\tilde{s} \geq 0$. For $j \in I$, $\tilde{s}_j = z_j \geq 0$, and for $j \in J$, we have

$$
\tilde{s}_j \geq s_j - \|L_I^{W^\perp}(z - s_I)\|_\infty \geq 16M^3n^3\|s_{\Lambda(s,x)}\|_1 - 16M^3n^3\|s_{\Lambda(s,x)}\|_\infty \geq 0\,. \quad (14)
$$

We now turn to dual proximity:

$$
\begin{aligned}
\|\tilde{s} - c\|_\infty &\leq \|\tilde{s} - s\|_\infty + \|s - c\|_\infty \leq \|L_I^{W^\perp}(z - s_I)\|_\infty + 8M^2n\|c_{\Lambda(c,d)}\|_1 \\
&\leq 16M^3n^3\|s_{\Lambda(s,x)}\|_\infty + 8M^2n\|c_{\Lambda(c,d)}\|_1 \\
&\leq 16M^3n^3\varepsilon\|c_{\Lambda(c,d)}\|_1 + 8M^2n\|c_{\Lambda(c,d)}\|_1 \leq 16M^2n\|c_{\Lambda(c,d)}\|_1\,,
\end{aligned}
$$

using (13), Claim 7.4.2 and that $\varepsilon < 1/(2Mn^2)$.

It is left to show $\|\tilde{x}_{\Lambda(\tilde{x}, \tilde{s})}\|_\infty \leq 2\varepsilon n\|\tilde{x}\|_\infty$. Recall that $\tilde{x} = x + d - \tilde{d}$, and that $d - \tilde{d} \geq 0$. Further, $\|d - \tilde{d}\|_\infty \leq \varepsilon\|d/W\|_1/n \leq \varepsilon\|\tilde{x}\|_\infty$ using Lemma 6.2.

Let us fix an index $j \in [n]$ such that $\tilde{x}_j < 0$ or $\tilde{s}_j > 0$; our goal is to show $|\tilde{x}_j| \leq 2\varepsilon n\|\tilde{x}\|_\infty$.

Assume first $j \in J$. Then, $\tilde{x}_j = x_j + d_j - \tilde{d}_j = d_j - \tilde{d}_j$ by Claim 7.4.1. Hence, we obtain the stronger bound $|\tilde{x}_j| \leq \|d - \tilde{d}\|_\infty \leq \varepsilon\|\tilde{x}\|_\infty$. For the rest, let us assume $j \in I$. Then, $\tilde{x}_j = w_j + d_j - \tilde{d}_j$.

Claim 7.4.3. $|w_j| \leq 2\varepsilon(n-1)\|w\|_\infty$.

Proof of Claim. From the recursive call, we know that $\|w_{\Lambda(w,z)}\|_\infty \leq 2\varepsilon(n-1)\|w\|_\infty$, using that $|I| \leq n - 1$. The claim follows by showing that $j \in \Lambda(w, z)$. Indeed, $\tilde{s}_I = z$, and thus $\tilde{s}_j > 0$ means $j \in \text{supp}(z^+)$. If $\tilde{x}_j < 0$, then $0 > \tilde{x}_j = w_j + d_j - \tilde{d}_j \geq w_j$ by the nonnegativity of $d - \tilde{d}$. Hence, $j \in \text{supp}(w^-)$. ∎

Since $\tilde{x}_I - w = d_I - \tilde{d}_I$, we obtain the bound $\|\tilde{x}_I - w\|_\infty \le \varepsilon \|\tilde{x}\|_\infty$, implying $\|w\|_\infty \le (1+\varepsilon)\|\tilde{x}\|_\infty$. We can thus bound

$$
\begin{aligned}
|\tilde{x}_i| &\le |w_i| + \varepsilon \|\tilde{x}\|_\infty \le 2\varepsilon(n-1)\|w\|_\infty + \varepsilon\|\tilde{x}\|_\infty \\
&\le \varepsilon\left((2n-2)(1+\varepsilon)+1\right)\|\tilde{x}\|_\infty \le 2n\varepsilon\|\tilde{x}\|_\infty\,,
\end{aligned}
$$

using that $\varepsilon < 1/(2n)$.

Part (iii) By the definition of I, we have

$$
\|s_I\|_\infty < 16M^3 n^2 \|s_{\Lambda(s,x)}\|_1 \le 16 M^3 n^3 \|s_{\Lambda(s,x)}\|_\infty \le 16\varepsilon M^3 n^3 \|c_{\Lambda(c,d)}\|_1\,. \tag{15}
$$

First, let us assume $c' = c$. This can happen in two cases: if $\|c_{\Lambda(c,d)}\|_1 < M\|c/W^\perp\|_1$ or if $\|c_{\Lambda(c,d)}\|_1 < \|c\|_\infty/(4M^2 n)$. In the first case,

$$
\|s_I\|_\infty < 16\varepsilon M^3 n^3 \|c_{\Lambda(c,d)}\|_1 \le \frac{\|c_{\Lambda(c,d)}\|_1}{2Mn} \le \frac{\|c/W^\perp\|_1}{2n} \le \frac{\|s\|_2}{2\sqrt{n}}\,,
$$

showing that $I \ne [n]$. The second inequality uses the choice of $\varepsilon < 1/(32M^4 n^4)$, and the last inequality uses the minimum-norm property Lemma 6.2. In the second case, we can bound

$$
\begin{aligned}
\|s\|_\infty &\ge \|c\|_\infty - \|c - s\|_\infty \ge 4M^2 n \|c_{\Lambda(c,d)}\|_1 - 3M^2 n \|c_{\Lambda(c,d)}\|_1 = M^2 n \|c_{\Lambda(c,d)}\|_1 \\
&\ge 16 M^6 n^5 \|s_{\Lambda(s,x)}\|_\infty > \|s_I\|_\infty\,,
\end{aligned} \tag{16}
$$

again implying $I \ne [n]$.

Let us now turn to the case $c' = c/W^\perp$. From (15), we get

$$
\|s_I\|_\infty < 16\varepsilon M^3 n^3 \|c'_{\Lambda(c',d)}\|_1 \le 16\varepsilon M^3 n^3 \varepsilon \|c/W^\perp\|_1 \le \frac{\|c/W^\perp\|_2}{2M\sqrt{n}} \le \frac{\|s\|_2}{2M\sqrt{n}}\,,
$$

showing again $I \ne [n]$. $\qquad\square$

Algorithm 3 implements the recursive calls in accordance with Lemma 7.4. This is the algorithm asserted in Theorem 7.1.

Proof of Theorem 7.1. The output of Algorithm 3 can be transformed to a solution to the system *F-Primal*(W, d, c, M) using (12).

We show that the algorithm returns the claimed output and admits the running time bound. Correctness of the recursive calls follows from Lemma 7.4, and part (iii) also shows that $I \ne [n]$, i.e., the problem reduces in every call.

The runtime consists of n calls to the *Prox-Opt-Solver* which takes $O(\Psi(A)\log(\bar{\chi}_A + n) + nm^{\omega-1})$ each by Lemma 7.3. We have to maintain the subspaces W_I throughout as required in form $W_I = \ker(A')$ for some matrix A' in form $A' = A_B^{-1}A$ by Theorem 5.3 for some basis $B \subset [n]$. Even though *Prox-Opt-Solver* is called for the dual subspace $W_I^\perp$, a representation of either subspace suffices due to the symmetry in Theorem 5.3.

Let A denote the original matrix. Initially, we perform Gaussian elimination to obtain a basis $B \subseteq [n]$ and the form $A' = A_B^{-1}A$ in time $O(m^2 n)$. At any point in the algorithm let $H \subseteq [n]$ denote the current index subset; we let $W \subseteq \mathbb{R}^H$ denote the current subspace. We also maintain the basis $B \subseteq H$, and the form $A' = A_B^{-1}A_H$; thus $W = \ker(A'_H)$. At every iteration we recurse on $I \subset H$ and subspace W_I. The matrix representation gets updated by deleting the columns $H \setminus I$. For $I \cap B$, pivot operations are required to expand the set $B \cap I$ to a new full rank basis B. This requires $|H \setminus I|$ basis exchanges. Hence the total runtime is $O(n^2 m)$ for these operations. Recursively we invoke line 12 n times. By Lemma 2.3 this takes in total $n\min\{m^2 n, n^\omega\}$ time. $\qquad\square$

Algorithm 3: INNERLOOP

> **Input** : $W \subseteq \mathbb{R}^n$, $c \in \mathbb{R}^n$, $d \in \mathbb{R}^n_+$ such that *Primal-Dual(W,d,c)* is feasible, $M \geq 1$.
> **Output**: A solution $(\tilde{d}, \tilde{x}, \tilde{s})$ to *F-Primal-Prox(W,d,c,M,ε)* or lifting certificate of
> $M < \kappa_W$.

1 $\varepsilon \leftarrow 1/(32M^4 n^4)$;

2 **if** $\|c_{\Lambda(c,d)}\|_1 \geq \max\left\{ M\|c/W^\perp\|_1, \frac{\|c\|_\infty}{4M^2 n} \right\}$ **then**

3 $\quad \llcorner \; c \leftarrow c/W^\perp$

4 **if** *Prox-Opt-Solver($W^\perp, c, d, M, \varepsilon$) in (i)* **then**

5 $\quad (\tilde{d}, s, x) \leftarrow$ *Prox-Opt-Solver($W^\perp, c, d, M, \varepsilon$)* ;

6 $\quad I = \{i : s_i < 16(Mn)^3 \|s_{\Lambda(s,x)}\|_1\}$, $J = [n] \setminus I$;

7 $\quad$ **if** $I = \emptyset$ **then**

8 $\quad \quad \llcorner \;$ **return** (x, s)

9 $\quad$ **else**

10 $\quad \quad$ **if** INNERLOOP($W_I, x_I, s_I, M, \varepsilon$) *returns a solution* **then**

11 $\quad \quad \quad (w, z) \leftarrow$ INNERLOOP($W_I, x_I, s_I, M, \varepsilon$) ;

12 $\quad \quad \quad$ **if** $\|L_I^{W^\perp}(z - s_I)\|_\infty \leq M\|z - s_I\|_1$ **then**

13 $\quad \quad \quad \quad \tilde{x} \leftarrow (w, x_J) + d - \tilde{d}$;

14 $\quad \quad \quad \quad \tilde{s} \leftarrow s + L_I^{W^\perp}(z - s_I)$;

15 $\quad \quad \quad \quad$ **return** $(\tilde{x}, \tilde{s})$;

16 $\quad \quad \quad$ **else**

17 $\quad \quad \quad \quad \llcorner \;$ **return** Lifting certificate of $M < \kappa_W$

18 $\quad \quad$ **else**

19 $\quad \quad \quad$ **return** Lifting certificate of $M < \kappa_W$

20 **else**

21 $\quad \llcorner \;$ **return** Lifting certificate of $M < \kappa_W$

7.3 Certificate for the wrong guess

If the feasibility algorithm is applied to a feasible instance, it always provides a certificate of $M < \kappa_W$ if no feasible solution is found. In contrast, Algorithm 2 only provides the verdict $M < \kappa_W$ without providing the corresponding certificate. The INNERLOOP is able to provide a certificate, but if infeasibility is detected in the calls to the FEASIBILITY-ALGORITHM in lines Line 19 and Line 20 then this means that the partition $B \cup N$ is wrong. Note that failed calls to FEASIBILITY-ALGORITHM could also result in $M < \kappa_W$ for which the feasibility solvers provide a certificate. We are nonetheless able to provide a certificate of $M < \kappa_W$ via repeated application of Corollary 4.4.

Lemma 7.5. *In line 20 of Algorithm 2, the dual feasibility solver always succeeds to find a dual solution or finds a certificate of $M < \kappa_W$. If the primal feasibility solver in line 19 fails to find a primal feasible solution, then we can find a lifting certificate of $M < \kappa_W$ in $O(n^3 m^2)$ time.*

Proof. First, let us show that if the algorithm reaches line 20, then dual feasibility is guaranteed. By induction, we show that there exists a dual feasible solution in the original subspace W^0, supported on $I \cup N$, and that for the current subspace $W \subseteq \mathbb{R}^I$, every dual feasible solution extends to such a solution in W^0.

Initially, $[n] = I$, and we assume that the initial system is both primal and dual feasible. Every call to INNERLOOP delivers a solution $(\tilde{d}, \tilde{x}, \tilde{s})$ to system *F-Primal(W,d,c,M)*, supported on the current index set I. Here, $\tilde{s}$ is a dual feasible solution, and $\langle \tilde{x}, \tilde{s} \rangle = 0$.

The algorithm only moves indices to B where $\tilde{x}_i > 0$, and consequently, $\tilde{s}_i = 0$. By induction, $\tilde{s}$ extends to a dual feasible solution in the original subspace supported on $I \cup N$. Moreover, the new value of $c \in \mathbb{R}^I$ is set as $\tilde{s}_I$, verifying the second inductive hypothesis.

In contrast, primal feasibility may fail during the algorithm in case of $M < \kappa_W$. We now show how to 'backtrack' and identify a lifting certificate of $M < \kappa_W$. Let $k \leq m$ be the number of iterations of the while loop in line 3 of Algorithm 2. Let $B_0 = N_0 = \emptyset$, and for $i \in [k]$, let B_i and N_i be the sets B and N at the end of the i-th iteration. Similarly, use notation W^i, $\tilde{x}^i, I^i, I_L^i, I_M^i, I_S^i$ for the spaces, vectors and variables at the end of the i-th iteration.

Define $z^k = 0_{I_k}$. We iteratively perform following pull-back to obtain $z^i \geq 0$ from $z^{i+1} \geq 0$ for $i = k, \ldots, 0$. Consider the system $y \geq -\tilde{x}^i, y \in W^i$. Note that $z^i - \tilde{x}^i$ is a feasible solution. Therefore Corollary 4.4 finds in time $O(mn(n-m)^2 + n^2 m^{\omega-1}) = O(mn^3)$ a feasible solution y^i to this system such that $\|y\|_\infty \leq M\|(\tilde{x}^i)^-\|_1$ or a certificate $M < \kappa_W$. If the result is such a certificate, we terminate. Otherwise define

$$z^i = \tilde{x}^i + L_{I^i \cup I_S^{0,i} \cup I_M^i}^{W^{i-1}}(y^i, -\tilde{x}_{I_S^{0,i}}^i, 0_{I_M^i}). \tag{17}$$

The lift exists as $y^i \in W^i$ and $I_S^{0,i} \subseteq \mathrm{cl}(I_L^i)$. We have $z_{I^i}^i = x_{I^i}^i + y_{I^i}^i \geq 0$, $z_{I_S^{0,i}}^i = 0$, and $z_{I_M^i}^i = \tilde{x}_{I_M^i}^i \geq 0$. Furthermore, if the lift does not provide a certificate of $M < \kappa_W$, then for $\ell \in I_L^i$ we have

$$z_\ell^i > \frac{\|\tilde{x}\|_\infty}{2n} - M\|(y^i, \tilde{x}_{I_S^{0,i}}^i)\|_1 \geq \frac{\|\tilde{x}\|_\infty}{2n} - Mn \max\left\{ M\|(\tilde{x}^i)^-\|_\infty, \frac{\|\tilde{x}\|_\infty}{4n^2 M} \right\} > 0, \tag{18}$$

where the last inequality follows from the precision ε in INNERLOOP. We just have shown that $z^i \geq 0$. Iterating the process would lead to $z^0 \in W + d, z^0 \geq 0$ and $\mathrm{supp}(z^0) \subseteq B$, a contradiction. Therefore either an application of Corollary 4.4 or a computation as in (17) must have resulted in a certificate of $M < \kappa_W$. The lifts in (17) can be performed in time $m \min\{m^2 n, n^\omega\}$ by Lemma 2.3. Therefore, the runtime is dominated by the repeated application of Corollary 4.4, which takes $O(m^2 n^3)$. $\qquad \square$

8 Implementation of oracles and subroutines

8.1 A symmetric initialization system

Throughout, we let $A \in \mathbb{R}^{m \times n}$, $W = \ker(A)$, and let $b \in \mathbb{R}^m$, $c, d \in \mathbb{R}^n$ such that $Ad = b$. In order to apply interior point methods to (LP), one needs to work with an extended system where a near-central initial solution can be easily obtained—in particular, both primal and dual sides must be strictly feasible. A common approach is to use the self-dual homogenous initialization [YTM94]; however, this may signficantly increase the condition numbers $\bar{\chi}_A$ and κ_A. An alternative initialization that approximately preserves $\bar{\chi}_A$ and κ_A was proposed in [VY96], and also used in [DHNV20]. The drawback of this formulation is that the primal and dual side are not symmetric: we would have to prove several properties on the primal and dual side separately. However, as primal and dual are *nearly* symmetric, major parts of the proofs of these lemmas would be identical.

We now propose a symmetric modification of the system in [VY96] that also preserves the condition numbers approximately. Throughout, M is an estimate of the κ_W. Given an instance (W, c, d), we derive two other parameters from M.

$$M_P = 2\|c\|_1 M, \quad M_D = 2\|d\|_1 M. \tag{19}$$

The system is then defined as follows:

$$\min \langle c, x - \underline{x} \rangle + M_P \langle \mathbb{1}, x \rangle \qquad \max \langle y, b \rangle - M_D \langle \mathbb{1}, \bar{s} \rangle$$
$$Ax - A\underline{x} = b \qquad A^\top y + s - \bar{s} = c$$
$$x - \frac{1}{2}\underline{x} + \bar{x} = M_D \mathbb{1} \qquad -A^\top y + \frac{1}{2}\bar{s} + \underline{s} = M_P \mathbb{1} - c \qquad \text{(Init-LP)}$$
$$x, \bar{x}, \underline{x} \geq 0 \qquad s, \bar{s}, \underline{s} \geq 0.$$

Using the identity

$$\langle y, b \rangle = \langle y, Ad \rangle = \langle A^\top y, d \rangle = -\langle s - \bar{s}, d \rangle + \langle c, d \rangle \qquad (20)$$

an equivalent formulation of primal and dual of (Init-LP) is

$$\min \langle c, x - \underline{x} \rangle + M_P \langle x, \mathbb{1} \rangle \qquad \max \langle d, c \rangle - \langle d, s - \bar{s} \rangle - M_D \langle \mathbb{1}, \bar{s} \rangle$$
$$x - \underline{x} \in W + d \qquad s - \bar{s} \in W^\perp + c$$
$$x - \frac{1}{2}\underline{x} + \bar{x} = M_D \mathbb{1} \qquad s - \frac{1}{2}\bar{s} + \underline{s} = M_P \mathbb{1} \qquad \text{(Init-LP-sub)}$$
$$x, \bar{x}, \underline{x} \geq 0 \qquad s, \bar{s}, \underline{s} \geq 0.$$

which displays the desired symmetry via $x \sim s$, $\underline{x} \sim \bar{s}$ and $\bar{x} \sim \underline{s}$. In the following we will show that the system can be initalized centrally and that the condition number $\bar{\chi}$ does not increase by too much. Let us begin with the latter. We denote by $\hat{A}$ the primal constraint matrix of (Init-LP), that is

$$\hat{A} = \begin{pmatrix} A & -A & 0 \\ I & -\frac{1}{2}I & I \end{pmatrix}. \qquad (21)$$

Lemma 8.1. *Let $\hat{W} = \ker(\hat{A})$ for $\hat{A}$ as in (21). Then, $\kappa_{\hat{W}} \leq 4\kappa_W$.*

Proof. Let $g = (x, y, z) \in \mathbb{R}^{3n}$ denote a minimum support vector in $\hat{W}$. First, assume there is an index $i \in [n]$ such that x_i, y_i, z_i all have nonzero values. Then, let $g' = (e_i, e_i, -\frac{1}{2}e_i)$, where e_i is the i-th unit vector. We have $g' \in \hat{W}$, and $\mathrm{supp}(g') \subseteq \mathrm{supp}(g)$. By the minimality of g, this implies $g = \alpha g'$ for $\alpha \neq 0$; the ratio between the largest and the smallest absolute value entries is 2.

For the rest of the proof, we can assume that there is no index i such that x_i, y_i, z_i are all nonzero. By construction, $w = x - y \in W$. Let $T_{xy} \subseteq [n]$ denote the set of indices where $x_i y_i \neq 0$, $T_x \subseteq [n]$ the set with $x_i \neq 0$, $y_i = 0$, and $T_y \subseteq [n]$ the set with $y_i \neq 0$, $x_i = 0$. By our assumption, if $i \in T_{xy}$ then $z_i = 0$, and therefore $x_i = \frac{1}{2}y_i$. If $i \in T_x$ then $z_i = -x_i$, and if $i \in T_y$ then $z_i = \frac{1}{2}y_i$.

We claim that w is a minimum support vector in W. Indeed, if there is a smaller support vector $w' \in W$ with $\mathrm{supp}(w') \subsetneq \mathrm{supp}(w)$, then we can map it to $g' = (x', y', z') \in \hat{W}$ as follows. For each $i \in T_{xy}$, we set $x'_i = 2w'_i$, $y'_i = w'_i$. For each $i \in T_x$, we let $x'_i = w'_i$, $z'_i = -w'_i$, and for each $i \in T_y$, we let $y'_i = -w'_i$, $z'_i = -\frac{1}{2}w'_i$. We set all other coordinates of g' to 0. It is easy to verify that $g' \in \hat{W}$ and $\mathrm{supp}(g') \subsetneq \mathrm{supp}(g)$, giving a contradiction.

Hence, the largest ratio between the absolute value of elements of w is $\leq \kappa_W$. The same construction as described above can be used to map the entries of w to the entries of g. This implies a bound $\leq 4\kappa_W$ on the ratios between the elements of g, since each of (x_i, y_i, z_i) will be one of $w_i, -w_i, 2w_i$ and $-\frac{1}{2}w_i$. $\qquad \square$

Initial solutions While we use the black box results in Theorem 5.1, it is worth noting that for interior point methods, the system can be easily initialized near the central path with the following solutions.

$$
\begin{aligned}
(x, \underline{x}, \bar{x}) &= \frac{2}{3} M_D(\mathbb{1}_n, \mathbb{1}_n, \mathbb{1}_n) + (d, 0_n, -d) \\
(y, s, \underline{s}, \bar{s}) &= \frac{2}{3} M_P(0_m, \mathbb{1}_n, \mathbb{1}_n, \mathbb{1}_n) + (0_m, 0_n, -\frac{1}{2}c, -c)
\end{aligned}
\tag{22}
$$

The duality gap between these solutions is $\approx \frac{4}{3} n M_P M_D$.

Box constraints Theorem 5.1 requires upper bounds R_P and R_D on the diameters of the primal and dual feasible sets. The formulation (*Init-LP*) may include unbounded directions. However, we can impose trivial upper bounds

$$
(x, \underline{x}, \bar{x}) \leq 2n M_D \mathbb{1}_{3n} \quad \text{and} \quad (s, \underline{s}, \bar{s}) \leq 2n M_P \mathbb{1}_{3n} .
\tag{23}
$$

Any solution violating these bounds would have worse gap than the trivial solution (22); therefore, adding such bounds will not affect optimality.

8.2 Proofs of Theorem 5.2 and Theorem 5.3

In the proofs, we can assume $d = d/W$ and $c = c/W^\perp$; otherwise, we can replace d and c by their projections to the respective subspaces. These vectors d and c can be computed as projections in $\min O(m^\omega, mn^2)$ time.

The approximate LP setup We use Theorem 5.1 to the extended system (*Init-LP*), with additional box constraints (23) (represented via slack variables). Our aim in Theorem 5.3 is to obtain primal and dual solutions $(x, \underline{x}, \bar{x})$ and $(s, \underline{s}, \bar{s})$ with duality gap $\leq \varepsilon M \|c\|_1 \|d\|_1 / 2$. In Theorem 5.2 we use it for the special case $c = 0$; we now explain the more general case.

We note that, instead of using the final output of Theorem 5.1, one could obtain more direct algorithms by using the interior-point methods of [LS19, vdB20, vdBLSS20] directly on our extended system. We now give a black-box argument using Theorem 5.1, to demonstrate the compatibility of our results with any approximate LP solver, not just interior-point methods.

Both Theorem 5.2 and 5.3 require in the input $A_B^{-1} A$ for a nonsingular submatrix A_B. For simplicity of notation, we assume A is already in such a form, with $B = [m]$, that is, $A = (I_m | T)$ for some $T \in \mathbb{R}^{m \times (n-m)}$. (Such a form could be obtained via Gaussian elimination, but that might take more time than a single call of the approximate algorithm itself.) Also, we replace the terms M_P and M_D with $\hat{M}_P = M_P - \varepsilon \|c\|_1$ and $\hat{M}_D = M_D - \varepsilon \|d\|_1$ respectively. The motivation behind this shift will become clear at the end of the paragraph.

The extended system will be of the form $\min \langle \hat{c}, \hat{x} \rangle$ s.t. $\hat{A}\hat{x} = \hat{b}$, $\hat{x} \geq 0$ with $\hat{A}$ as in (21); we ignore the box constraints for the simplicity of presentation.

Let us now bound the parameters. We have the diameter bounds $R_P = O(n^{3/2} M_D) = O(n^{5/2} M \|d\|_1)$ and $R_D = O(n^{3/2} M_P) = O(n^{5/2} M \|c\|_1)$. Using the special form of A, by Proposition 2.5 we have $\|A\|_F \leq n \|A\|_{\max} \leq n\kappa_W$. For the matrix of the extended system (21), we also have $\|\hat{A}\|_F \leq O(n\kappa_W) = O(nM)$, if our guess M on κ_W was correct. Otherwise, we obtain a lifting certificate of $M < \kappa_W$ that can be derived from one of the circuits in the form of A.

We next bound $\|\hat{b}\|$ and $\|\hat{c}\|$. We have $\|b\|_\infty \leq \|A'\|_{\max} \|d\|_1 \leq n\kappa_W \|d\|_1$. If our guess $M > \kappa_W$ is correct, then this is upper bounded as $nM\|d\|_1$; otherwise, we obtain a lifting certificate of $M < \kappa_W$. The vector $\hat{b}$ also contains additional entries of $\hat{M}_D = (M - \varepsilon)\|d\|_1$; thus, we get $\|\hat{b}\|_2 \leq O(n^{3/2} M \|d\|_1)$. Similarly, we obtain $\|\hat{c}\|_2 \leq O(n^{3/2} M \|c\|_1)$.

The output of Theorem 5.1 maintains nonnegativity but allows for violation of the subspaces constraints $\hat{A}\hat{x} = \hat{b}$ and $\hat{A}^\top \hat{y} + \hat{s} = \hat{c}$. In Theorem 5.2 and Theorem 5.3, we require the points in

the right subspaces, but allow for violation of nonnegativity. We now show how fix the violations in the subspaces to obtain a proximal feasible solution to *Init-LP*.

We will use the algorithm in Theorem 5.1 with $\delta = \gamma \varepsilon n^{-4.5} M^{-2}$ for a suitable constant $\gamma > 0$, so that the optimality gap is bounded by at most $\gamma \varepsilon M \|c\|_1 \|d\|_1$, $\|\hat{A}\hat{x} - \hat{b}\|_2$ is bounded by $\gamma \varepsilon \|d\|_1 / n$, and $\|\hat{A}^\top \hat{y} + \hat{s} - \hat{c}\|$ is bounded by $\gamma \varepsilon \|c\|_1 / n$. We let $\hat{x} = (x', \underline{x}', \bar{x}')$, $\hat{y}$, and $\hat{s} = (s', \underline{s}', \underline{x}')$ denote the primal and dual solutions.

Recalling the form $A = (I_m|T)$, we can subtract $|A_i(x - \underline{x}) - b_i| \leq \gamma \varepsilon \|d\|_1 / n$ from either x'_i or $\underline{x}'_i$ to obtain a feasible solution to the equality system. By subsequently shifting up x' and $\underline{x}'$ by $2\gamma \varepsilon \|d\|_1 / n \mathbb{1}$ each, we maintain feasibility of the equality system and achieve nonnegativity of x' and $\underline{x}'$. Additionally, the equality $x - \frac{1}{2}\underline{x}' + \bar{x}' = M_D \mathbb{1}$ is satisfied. Thus the solution is feasible to the extended primal system (*Init-LP*).

The total adjustment to the original solution amounts to $2\gamma \varepsilon \|d\|_1 / n$ per coordinate at most. Therefore, the objective value changed by at most

$$3n \cdot \|\hat{c}\|_\infty \cdot 2\gamma \varepsilon \|d\|_1 / n = 6\gamma \varepsilon \hat{M}_P \|d\|_1 < 6\gamma \varepsilon M \|c\|_1 \|d\|_1 \,.$$

Adjusting the dual can be done equivalently, noting the symmetry in (*Init-LP-sub*). Note that this requires a matrix $B \in \mathbb{R}^{(n-m) \times n}$ such that $\text{Im}(B) = W^\perp$, which might be computed by Gram-Schmidt orthogonalisation. This could take longer than a single call of the approximate algorithm itself, therefore we assume that this matrix is given.

All together, we obtain feasible primal and dual solution $(x, \underline{x}, \bar{x})$, y, $(s, \underline{s}, \bar{s})$ to (*Init-LP*) with objective values within $\varepsilon M \|c\|_1 \|d\|_1$ if $\gamma \leq 1/13$ is chosen. The running time can be bounded as

$$O(\Psi(A) \log(n/\delta)) = O(\Psi(A) \log((M + n)/\varepsilon)) \,.$$

Theorem 5.2. *Let $A \in \mathbb{R}^{m \times n}$, $\text{rk}(A) = m$, $W = \ker(A)$, and $d \in \mathbb{R}^n$, and assume the matrix $A_B^{-1} A$ is provided for some $m \times m$ non-singular submatrix A_B. Let $0 < \varepsilon \leq 1/2$, and let $M \geq 2$ be an estimate of κ_W. Consider a linear feasibility problem $x \in W + d, x \geq 0$. There exists an algorithm that returns either of the following outcomes:*

(F1) A near-feasible solution $x \in W + d$, $\|x^-\|_1 \leq \varepsilon \|d/W\|_1$, $\|x\|_\infty \leq 2M\|d\|_1$.

(F2) A Farkas certificate of infeasibility: $s \in (W^\perp)_+$, $\langle d, s \rangle < 0$.

(F3) A lifting certificate of $M < \kappa_W$.

The running time is $O(\Psi(A) \log((M + n)/\varepsilon))$ for outcome (F1), and $O(\Psi(A) \log((M + n)/\varepsilon) + (n - m)m^2 + n^\omega)$ in outcomes (F2) and (F3).

Proof of Theorem 5.2. If $d = 0$, then we return $x = 0$. Otherwise, we use the approximate solver as described above, for systems of the form (*Init-LP*) with $c = 0$, $M_P = 1$, and $M_D = 2M\|d\|_1$. We thus obtain solutions $(x, \underline{x}, \bar{x})$ and $(y, s, \underline{s}, \bar{s})$ with duality gap $\leq \varepsilon \|d\|_1$. Assume the dual objective value is ≤ 0. In this case, the primal objective value is $\langle \mathbb{1}, \underline{x} \rangle \leq \varepsilon \|d\|_1$. Setting $\hat{x} = x - \underline{x}$, we get the required solution

$$\hat{x} \in W + d, \quad \|\hat{x}^-\|_1 \leq \varepsilon \|d\|_1, \quad \|\hat{x}\|_\infty \leq 2M\|d\|_1 \,. \tag{24}$$

Next, assume the dual objective value is $-\langle d, s - \bar{s} \rangle - M_D \langle \mathbb{1}, \bar{s} \rangle > 0$ for vector $s, \bar{s} \geq 0$, $s - \bar{s} \in W^\perp$. Then $\hat{s} = s - \bar{s}$ satisfies

$$\langle d, \hat{s} \rangle < -2M\|d\|_1 \|\hat{s}^-\|_1 \,. \tag{25}$$

We now apply Corollary 4.2 to $W^\perp$, the vector $\hat{s}$, and $J = \text{supp}(\hat{s}^-)$ to find a vector $z \in W^\perp$ that is sign-consistent with $\hat{s}$, $z_J = 0$, and $\|z - \hat{s}\|_\infty \leq M\|\hat{s}^-\|_1$ in $O((n - m)m^2 + n(n - m)^{\omega-1}) = O((n - m)m^2 + n^\omega)$ time. If no such vector is found, then we obtain a lifting certificate of $M < \kappa_{W^\perp} = \kappa_W$ as in (F3).

By the sign-consistency, $z \geq 0$, since $\hat{s}_i \geq 0$ for all $i \in [n] \setminus J$. We show that $\langle d, z \rangle < 0$, and consequently, z is a Farkas certificate as in (F2). This follows since

$$\langle d, z \rangle = \langle d, \hat{s} \rangle + \langle d, z - \hat{s} \rangle \leq \langle d, \hat{s} \rangle + \|d\|_1 \|z - \hat{s}\|_\infty < -2M\|d\|_1 \|\hat{s}^-\|_1 + M\|d\|_1 \|\hat{s}^-\|_1 \leq 0. \quad \square$$

Theorem 5.3. *Let $A \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(A) = m$, $W = \ker(A)$, and $d \in \mathbb{R}^n$, and assume the matrix $A_B^{-1}A$ is provided for some $m \times m$ non-singular submatrix A_B. Let $0 < \varepsilon \leq 1/2$, and let $M \geq 2$ be an estimate of κ_W. Consider a linear program of the form Primal-Dual(W, d, c). There exists an algorithm that returns either of the following outcomes:*

(M1) *A pair of primal and dual near-feasible and near-optimal solutions $x \in W + d$, $s \in W^\perp + c$, that is,*

$$\begin{aligned}
\|x^-\|_1 &\leq \varepsilon \|d/W\|_1, \\
\|s^-\|_1 &\leq \varepsilon \|c/W^\perp\|_1, \\
\|x \circ s\|_1 &\leq 5\varepsilon M \|d/W\|_1 \|c/W^\perp\|_1, \\
\|x\|_\infty &\leq 2M \|d/W\|_1, \text{ and} \\
\|s\|_\infty &\leq 2M \|c/W^\perp\|_1
\end{aligned} \tag{8}$$

Further, if $\kappa_W \leq M$, and both primal and dual sides of Primal-Dual(W, d, c) are feasible, and the optimum value is OPT, then

$$\langle c, x \rangle - 5\varepsilon M \|d/W\|_1 \|c/W^\perp\|_1 \leq OPT \leq \langle d, c - s \rangle + 5\varepsilon M \|d/W\|_1 \|c/W^\perp\|_1. \tag{9}$$

(M2) *A Farkas certificate of primal infeasibility: $s \in (W^\perp)_+$, $\langle d, s \rangle < 0$.*

(M3) *A Farkas certificate of dual infeasibility: $x \in W_+$, $\langle c, x \rangle < 0$.*

(M4) *A lifting certificate of $M < \kappa_W = \kappa_{W^\perp}$.*

The running time is $O(\Psi(A) \log((M + n)/\varepsilon))$ for outcome (M1). For (M2), (M3) and (M4) the runtime is $O(\Psi(A) \log((M + n)/\varepsilon) + n^2 m + n^\omega)$ if $d = 0$ or $c = 0$ and $O(\Psi(A) \log((M + n)/\varepsilon) + n^3 m)$ if $c \neq 0$ and $d \neq 0$.

Proof of Theorem 5.3. First, apply Theorem 5.2 to data $(W, d, \varepsilon/4, M)$ and $(W^\perp, c, \varepsilon/4, M)$. If either returns in (F3) we return in (M4). If the first call returns in (F2), we return in (M2), if the second call returns in (F2), we return in (M3). Note that the claimed runtime bounds are observed. From now on assume that both calls returned in (F1). Then we obtain vectors $\hat{x}$ and $\hat{s}$ such that

$$\begin{aligned}
\hat{x} \in W + d, \quad \|\hat{x}^-\| &\leq \frac{\varepsilon}{4} \|d/W\|_1, \qquad \|\hat{x}\|_\infty \leq 2M\|d\|_1, \\
\hat{s} \in W^\perp + c, \quad \|\hat{s}^-\| &\leq \frac{\varepsilon}{4} \|c/W^\perp\|_1, \qquad \|\hat{s}\|_\infty \leq 2M\|c\|_1.
\end{aligned} \tag{26}$$

Proceed by calling an approximate interior point solver as described above, with M_P and M_D defined as in (19). In time $O(\Psi(A) \log((M + n)/\varepsilon))$ we can obtain solutions $(x, \underline{x}, \bar{x})$ and $(y, s, \underline{s}, \bar{s})$ with normalized duality gap $\leq \varepsilon M \|c\|_1 \|d\|_1/2$.

We let $\tilde{x} = x - \underline{x}$ and $\tilde{s} = s - \bar{s}$. These fulfill $\tilde{x} \in W + d$ and $\tilde{s} \in W^\perp + c$. We distinguish two cases.

Case I: $\|\tilde{x}^-\|_1 \leq \varepsilon \|d\|_1$ and $\|\tilde{s}^-\|_1 \leq \varepsilon \|c\|_1$. In this case, we claim that $(\tilde{x}, \tilde{s})$ is a near-feasible near-optimal pair of solutions as required in (M1). First, we have upper bounds $\|\tilde{x}\|_\infty \leq 2M\|d\|_1$, $\|\tilde{s}\|_\infty \leq 2M\|c\|_1$ from

$$\begin{aligned}
\tilde{x} = x - \underline{x} \leq x - \frac{1}{2}\underline{x} + \bar{x} &= 2M\|d\|_1 \mathbb{1} \\
\tilde{s} = s - \bar{s} \leq s - \frac{1}{2}\bar{s} + \underline{s} &= 2M\|c\|_1 \mathbb{1}.
\end{aligned} \tag{27}$$

We now verify the bound on $\|x \circ s\|$. With the upper bounds on $\tilde{x}$ and $\tilde{s}$

$$\|\tilde{x} \circ \tilde{s}\|_1 \leq \|x \circ s\|_1 + \|\tilde{x}^- \circ \tilde{s}^-\|_1 + \|\tilde{x}^- \circ \tilde{s}\|_1 + \|\tilde{x} \circ \tilde{s}^-\|_1$$
$$\leq \left(\frac{\varepsilon}{2}M + \varepsilon^2 + 4\varepsilon M\right)\|c\|_1\|d\|_1 \leq 5\varepsilon M\|c\|_1\|d\|_1\,,$$

using the assumption $\varepsilon \leq 1/2$.

Finally, consider (9). If $\kappa_W \geq M$, and $x \in W + d$, $x \geq 0$ is feasible and bounded, then there exists an optimal solution x^* with $\|x^*\|_\infty \leq M_D$. We can map this to a solution of the extended system as $(x^*, 0, M_D \mathbb{1}_n - x^*)$, with the same objective value $\text{OPT} = \langle c, x^*\rangle$. By weak duality on the extended system, this is lower bounded by the dual objective value of $(\tilde{s}^+, \tilde{s}^-, M_P \mathbb{1}_n - \tilde{s}^+ + \frac{1}{2}\tilde{s}^-)$, that is $\langle d, c - \tilde{s}\rangle - M_D\|\tilde{s}^-\|_1$. The bound $\text{OPT} \leq \langle d, c - \tilde{s}\rangle + 5\varepsilon M\|c\|_1\|d\|_1$ follows by this assumption and the bound on the duality gap. The primal bound follows analogously.

Case II: $\|\tilde{x}^-\|_1 > \varepsilon\|d\|_1$ or $\|\tilde{s}^-\|_1 > \varepsilon\|c\|_1$. In this case, we use the approximately feasible solutions $\hat{x}$ and $\hat{s}$ to find a lifting certificate of $M > \kappa_W$ or $M > \kappa_{W^\perp}$.

Assume that $\|\tilde{x}^-\|_1 > \varepsilon\|d\|_1$. Let us define $\ell = \tilde{x} - 2M\|d\|_1\mathbb{1}$, and $u = \tilde{x} + \hat{x}^-$. Consider the system

$$v \in W\,, \quad \ell \leq v \leq u\,.$$

The vector $\tilde{x} - \hat{x}$ is a feasible solution. We now apply Corollary 4.4 to find a lifting certificate of $M > \kappa_W$ or another vector v feasible to the above system such that

$$\|v\|_\infty \leq M\|\ell^+ + u^-\|_1 = M\|u^-\|_1 \leq M\|\tilde{x}^-\|_1\,,$$

since $\ell \leq 0$ by (27), and $|u_i| \leq \tilde{x}_i^-$ whenever $u_i < 0$. We let $z = \tilde{x} - v$. Now,

$$z \in W + d\,, \quad -\hat{x}^- \leq z \leq 2M\|d\|_1\mathbb{1}\,, \quad \|z - \tilde{x}\|_\infty \leq M\|\tilde{x}^-\|_1\,.$$

In particular, $\|z^-\|_1 \leq \|\hat{x}^-\|_1 \leq \frac{\varepsilon}{4}\|d\|_1$. We can map z to a primal feasible solution $(z^+, z^-, M_D\mathbb{1} - z^+ + \frac{1}{2}z^-)$ of (Init-LP). The objective value of this solution is

$$\langle c, z\rangle + 2M\|c\|_1\|z^-\|_1 \leq \langle c, \tilde{x}\rangle + \langle c, z - \tilde{x}\rangle + \frac{\varepsilon}{2}M\|c\|_1\|d\|_1$$
$$\leq \langle c, \tilde{x}\rangle + \|c\|_1\|z - \tilde{x}\|_\infty + \frac{\varepsilon}{2}M\|c\|_1\|d\|_1$$
$$\leq \langle c, \tilde{x}\rangle + M\|c\|_1\|\tilde{x}^-\|_1 + \frac{\varepsilon}{2}M\|c\|_1\|d\|_1$$
$$< \langle c, \tilde{x}\rangle + 2M\|c\|_1\|\tilde{x}^-\|_1 - \frac{\varepsilon}{2}M\|c\|_1\|d\|_1$$
$$\leq \langle c, x - \underline{x}\rangle + 2M\|c\|_1\|\underline{x}\|_1 - \frac{\varepsilon}{2}M\|c\|_1\|d\|_1\,,$$

where the penultimate inequality used the assumption $\|\tilde{x}^-\|_1 > \varepsilon\|d\|_1$. This is a contradiction, since $\langle c, x - \underline{x}\rangle + 2M\|c\|_1\|\underline{x}\|_1$ is the objective value of the solution $(x, \underline{x}, \bar{x})$ that was chosen to be within $\frac{\varepsilon}{2}M\|c\|_1\|d\|_1$ from the optimum value.

Hence, such a feasible solution does not exist, and therefore, the algorithm in Corollary 4.4 must have terminated with a lifting certificate of $M < \kappa_W$ as in (F2). The analogous argument applies for the case $\|\tilde{s}^-\|_1 > \varepsilon\|c\|_1$. The time complexity bound follows: the additional running time term $O(n^3m + n^{\omega+1})$ comes from Corollary 4.4, and is only needed in Case II, that leads to (M4). When applied to the primal we get $O(nm(n-m)^2 + nm^{\omega-1})$ as in Corollary 4.4. When applied to the dual, the rank is $n - m$ and so the runtime is $O(nm^2(n-m) + n(n-m)^{\omega-1})$. Put together, it results in runtime $O(n^3m + n^\omega) = O(n^3m)$. $\qquad\square$

8.3 Implementation of the oracles

We proceed to show Lemma 6.1 and Lemma 7.3 on the implementations of Oracle 1 and Oracle 2, respectively. We restate both statements and oracles for convenience.

Lemma 6.1. *There exists an $O(\Psi(A) \cdot \log(M + n) + nm^{\omega-1+o(1)})$ time algorithm, that either returns a solution to Prox-Feas(W, d, M, ε), or concludes that (ii) or (iii) should be the outcome of Prox-Feas-Solver(W, d, M, ε). In the latter case, these outcomes can be obtained in additional time $O(nm^2 + n^{\omega+o(1)})$.*

Lemma 7.3. *Assume we are given a matrix $A \in \mathbb{R}^{m \times n}$, vectors $c \in \mathbb{R}^n_+$ and $d \in \mathbb{R}^n$; let $W = \ker(A)$, and M be an estimate on κ_W. Further, let $0 < \varepsilon < 1$. There exists an $O(\Psi(A) \cdot \log(M+n) + nm^{\omega-1+o(1)})$ time algorithm, that either returns a solution to Prox-Opt$(W, d, c, M, \varepsilon)$ or concludes that (ii), (iii) or (iv) should be the outcome of Prox-Opt-Solver$(W, d, c, M, \varepsilon)$. These latter outcomes require an additional computational time $O(n^3 m)$.*

Input : A subspace $W \subseteq \mathbb{R}^n$, given as $W = \ker(A)$ for $A \in \mathbb{R}^{m \times n}$, a vector $d \in \mathbb{R}^n$, $M, \varepsilon > 0$.

Output: One of the following three outcomes

  (i) A solution x to the system

$$
\begin{aligned}
x &\in W + d \\
\|x - d\|_\infty &\leq 3M^2 n \|d^-\|_1 \qquad\qquad (\textit{Prox-Feas}(W, d, M, \varepsilon)) \\
\|x^-\|_\infty &\leq \varepsilon \|d^-\|_1
\end{aligned}
$$

  (ii) A vector $y \in W^\perp$, $y \geq 0$, $\langle d, y \rangle < 0$,

  (iii) A subset $I \subseteq [n]$ and a vector $p \in \pi_I(W)$ such that $\|L_I^W(p)\|_\infty > M\|p\|_1$.

Oracle 1: *Prox-Feas-Solver(W, d, M, ε)*

We note that Oracle 1 is a special case of Oracle 2; thus, Lemma 6.1 follows as a special case of Lemma 7.3. We now give the proof of the latter statement.

Proof of Lemma 7.3. Let us define

$$
\tau = \|d_{\Lambda(d,c)}\|_1, \ I := \{i \in [n] : d_i \leq 2M\tau\}, \quad \text{and} \quad \varepsilon' = \frac{\varepsilon^2}{28M^3 n^3}.
$$

We run the algorithm in Theorem 5.3 on the system *Primal-Dual*$(\pi_I(W), d_I, c_I)$ and with ε'. Computing the corresponding constraing matrix can be done in $O(nm^{\omega-1})$. The reason that we do not consider the elements in $[n] \setminus I$ is that we need to bound the minimum-norm vector in the affine space in which we run the solver. Note that $\|d_I/\pi_I(W)\|_1 \leq 2n^{3/2}M\tau$ and that supp$(c) \subseteq I$ as $c \geq 0$. Let us discuss the potential outcomes of the algorithm. We can only be in case (M2), if the original system $x \in W + d$ was already infeasible. In particular, a Farkas certificate extends to this original system. (M3) can not happen as $c_{[n]\setminus I} = 0$ by definition of I and so c_I is a feasible dual solution to *Primal-Dual*$(\pi_I(W), d_I, c_I)$. If in case (M4), we output the corresponding lifting certificate of $M < \kappa_W$ and terminate.

Assume now we obtained approximately feasible and approximately optimal primal and dual

Input : $W \subseteq \mathbb{R}^n$, $c \in \mathbb{R}^n_+$, $d \in \mathbb{R}^n$, $M \geq 2, \varepsilon > 0$.

Output: One of the following:

(i) A solution to the system

$$x \in W + d$$
$$s \in W^\perp + \tilde{c}$$
$$\|x_{\Lambda(x,s)}\|_\infty \leq \varepsilon \|d_{\Lambda(d,c)}\|_1$$
$$\|x - d\|_\infty \leq 3M^2 n \|d_{\Lambda(d,c)}\|_1 \qquad (\text{Prox-Opt}(W,d,c,M,\varepsilon))$$
$$\|c - \tilde{c}\|_\infty \leq \frac{\varepsilon}{n}\|c/W^\perp\|_1$$
$$c - \tilde{c} \geq 0$$
$$s \geq 0$$

(ii) A vector $y \in W^\perp$, $y \geq 0$, $\langle d, y \rangle < 0$,

(iii) A vector $x \in W$, $x \geq 0$, $\langle c, x \rangle < 0$,

(iv) A lifting certificate of $M < \kappa_W = \kappa_{W^\perp}$.

Oracle 2: *Prox-Opt-Solver*$(W, d, c, M, \varepsilon)$

---

solutions $(\hat{x}, \hat{s})$ as in (M1) fulfilling

$$\|\hat{x}^-\|_1 \leq \varepsilon' \|d_I/\pi_I(W)\|_1 \leq \varepsilon' \cdot 2n^{3/2} M\tau,$$
$$\|\hat{s}^-\|_1 \leq \varepsilon' \|c/W^\perp\|_1,$$
$$\|\hat{x} \circ \hat{s}\|_1 \leq 10\varepsilon' M^2 n^{3/2} \tau \|c/W^\perp\|_1, \qquad (28)$$
$$\|\hat{x}\|_\infty \leq 4nM^2\tau, \text{ and}$$
$$\|\hat{s}\|_\infty \leq 2M\|c/W^\perp\|_1.$$

Further, extend $\hat{x}$ arbitrarily to an element $x' \in W + d$ with $x'_I = \hat{x}$ and set $s' = (0_{[n]\setminus I}, \hat{s})$.

We proceed by converting this solution into another one that is proximal to d without increasing the objective value respective to c. To this end consider, as in the proof of Corollary 3.3, a system $z \in W$, $\ell \leq z \leq u$ with

$$\ell = -d - (x')^- \quad \text{and} \quad u_i = \begin{cases} x'_i - d_i & \text{if } i \in \text{supp}(c), \\ \infty & \text{else.} \end{cases}$$

We have that

$$\|\ell^+ + u^-\|_1 \leq \|d_{\Lambda(d,c)}\|_1 + \|(x')^-\|_1 \leq \tau + \varepsilon' \cdot 2Mn^{3/2}\tau \leq 2\tau,$$

As $x' - d$ is a feasible solution we can apply Corollary 4.4. Assume it can find a feasible solution z with $\|z\|_\infty \leq M\|\ell^+ + u^-\|_1 \leq 2M\tau$, as otherwise we terminate with a certificate of $M < \kappa_W$. Defining $\tilde{x} = d + z$, we note that $\tilde{x}_{[n]\setminus I} \geq 0$ and $\langle c, \tilde{x} \rangle = \langle c, d + z \rangle \leq \langle c, x' \rangle$ hold as $c \geq 0$ and $d_i + z_i \leq x'_i$ for $i \in \text{supp}(c)$.

On the dual side we turn the proximal feasible vector s' into a feasible vector, using the feasible vector $c \geq 0$, again by applying Corollary 4.4. Consider the system

$$w \in W^\perp \cap \mathbb{R}^n_I, w \geq -s', \qquad (29)$$

of which $c - s'$ is a feasible solution. Therefore we are able to find a solution w with $\|w\|_\infty \leq M\|(s')^-\|_1 = M\|\hat{s}^-\|_1 \leq \varepsilon' M\|c/W^\perp\|_1$ or terminate with a certificate $M < \kappa_W$. In the former case, let $\bar{s} = s' + w \geq 0$ and define $\tilde{s}$ as

$$
\tilde{s}_i = \begin{cases} 0 & \text{if } \bar{s}_i \leq \frac{\varepsilon}{n}\|c/W^\perp\|_1\,, \\ \bar{s}_i & \text{otherwise.} \end{cases}
$$

Note that $\bar{s} \geq \tilde{s} \geq 0$ and so setting $\tilde{c} = c - \bar{s} + \tilde{s}$ gives $c - \tilde{c} \geq 0$, $\tilde{s} \in W^\perp + \tilde{c}$, and

$$
\|\tilde{c} - c\|_\infty = \|\tilde{s} - \bar{s}\|_\infty \leq \frac{\varepsilon}{n}\|c/W^\perp\|_1.
$$

Further we have

$$
\|\tilde{x}^-\|_1 \leq \|\hat{x}^-\|_1 \leq \varepsilon' \cdot 2Mn^{3/2}\tau, \text{ and } \|\tilde{s}\|_\infty \leq \|\bar{s}\|_\infty \leq \|\hat{s}\|_\infty + \|w\|_\infty \leq 3M\|c/W^\perp\|_1. \quad (30)
$$

To bound the duality gap beween $\tilde{x}$ and $\tilde{s}$, we note that $\langle \tilde{x}, \bar{s} \rangle \leq \langle x', \bar{s} \rangle$ as $\langle \tilde{x}, c \rangle \leq \langle x', c \rangle$ and $c - \bar{s} \in W^\perp$. Also recall that $\tilde{s} - \bar{s} \leq 0$. Therefore,

$$
\begin{aligned}
\langle \tilde{x}, \tilde{s} \rangle &= \langle \tilde{x}, \bar{s} \rangle + \langle \tilde{x}, \tilde{s} - \bar{s} \rangle \\
&\leq \langle x', \bar{s} \rangle + \|\tilde{x}^-\|_1\|\tilde{s} - \bar{s}\|_\infty \\
&= \langle x'_I, s'_I \rangle + \langle x'_I, \bar{s}_I - s'_I \rangle + \|\tilde{x}^-\|_1\|\tilde{s} - \bar{s}\|_\infty \\
&= \langle \hat{x}, \hat{s} \rangle + \langle \hat{x}, w_I \rangle + \|\tilde{x}^-\|_1\|\tilde{s} - \bar{s}\|_\infty \\
&\leq \|\hat{x} \circ \hat{s}\|_1 + \|\hat{x}\|_1\|w\|_\infty + \|\tilde{x}^-\|_1\|\tilde{s} - \bar{s}\|_\infty \\
&\leq 10\varepsilon' M^2 n^{3/2}\tau\|c/W^\perp\|_1 + 4n^2 M^2\tau \cdot \varepsilon' M\|c/W^\perp\|_1 + 2\varepsilon' Mn^{3/2}\tau \cdot \frac{\varepsilon}{n}\|c/W^\perp\|_1 \\
&\leq 16\varepsilon' M^3 n^2\tau\|c/W^\perp\|_1
\end{aligned} \quad (31)
$$

and by

$$
\langle \tilde{x}^-, \tilde{s} \rangle \leq \|\tilde{x}^-\|_1\|\tilde{s}\|_\infty \leq 6\varepsilon' M^2 n^{3/2}\tau\|c/W^\perp\|_1,
$$

we also get

$$
\|\tilde{x} \circ \tilde{s}\|_1 = \langle \tilde{x}, \tilde{s} \rangle + 2\langle \tilde{x}^-, \tilde{s} \rangle \leq 28\varepsilon' M^3 n^2\tau\|c/W^\perp\|_1, \quad (32)
$$

where the equality used $\tilde{s} \geq 0$. Let us verify the requirements of Prox-Opt$(W, d, c, M, \varepsilon)$. The bounds on $c - \tilde{c}$ were shown already. Using the bound on the norm of z we get the primal proximity as

$$
\|\tilde{x} - d\|_\infty = \|z\|_\infty \leq 2M\tau \leq 3M^2 n\tau\,.
$$

Let us now turn to the bound on $\|\tilde{x}_{\Lambda(\tilde{x}, \tilde{s})}\|_\infty$. First, let $j \in \text{supp}(\tilde{x}^-)$. As shown above we have $j \in I$ and further $\tilde{x}_j \geq x'_j$ and therefore

$$
|\tilde{x}_j| \leq |(x'_j)^-| \leq \|(x'_I)^-\|_1 = \|\hat{x}^-\|_1 \leq \varepsilon' \cdot 2Mn^{3/2}\tau < \varepsilon\tau\,.
$$

Next, let $j \in \text{supp}(\tilde{s}^+)$. Using (32) we obtain

$$
|\tilde{x}_j| \leq \frac{\|\tilde{x} \circ \tilde{s}\|_1}{\tilde{s}_j} \leq \frac{28\varepsilon' M^3 n^2\tau\|c/W^\perp\|_1}{\frac{\varepsilon}{n}\|c/W^\perp\|_1} \leq \varepsilon\tau, \quad (33)
$$

by choice of ε'. We conclude that all inequalities of Prox-Opt$(W, d, c, M, \varepsilon)$ are satisfied. $\qquad\square$

References

[BH74]   James R. Bunch and John E. Hopcroft. Triangular factorization and inversion by fast matrix multiplication. *Mathematics of Computation*, 28(125):231–231, January 1974. 10, 17

[CGST86]   William Cook, Albertus MH Gerards, Alexander Schrijver, and Éva Tardos. Sensitivity theorems in integer linear programming. *Mathematical Programming*, 34(3):251–264, 1986. 2, 14

[CLS19]   Michael B Cohen, Yin Tat Lee, and Zhao Song. Solving linear programs in the current matrix multiplication time. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 938–942, 2019. 3

[DHNV20]   Daniel Dadush, Sophie Huiberts, Bento Natura, and László A Végh. A scaling-invariant algorithm for linear programming whose running time depends only on the constraint matrix. In *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing (STOC)*, pages 761–774, 2020. 3, 7, 11, 12, 13, 32

[Dik67]   II Dikin. Iterative solution of problems of linear and quadratic programming. *Doklady Akademii Nauk*, 174(4):747–748, 1967. 10

[DVZ18]   Daniel Dadush, László A Végh, and Giacomo Zambelli. Geometric rescaling algorithms for submodular function minimization. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 832–848. SIAM, 2018. 2

[FT87]   András Frank and Éva Tardos. An application of simultaneous diophantine approximation in combinatorial optimization. *Combinatorica*, 7(1):49–65, 1987. 7

[GL97]   Clovis C. Gonzaga and Hugo J. Lara. A note on properties of condition numbers. *Linear Algebra and its Applications*, 261(1):269 – 273, 1997. 12

[GLS12]   Martin Grötschel, László Lovász, and Alexander Schrijver. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012. 7

[GT89]   A. V. Goldberg and R. E. Tarjan. Finding minimum-cost circulations by canceling negative cycles. *Journal of the ACM (JACM)*, 36(4):873–886, 1989. 2

[Hof52]   Alan J. Hoffman. On approximate solutions of systems of linear inequalities. *J. Res. Natl. Bur. Stand.*, 49(4):263–265, 1952. 2, 13

[HT02]   Jackie CK Ho and Levent Tunçel. Reconciliation of various complexity and condition measures for linear programming problems and a generalization of Tardos' theorem. In *Foundations of Computational Mathematics*, pages 93–147. World Scientific, 2002. 4, 10, 11, 13, 14

[IFF01]   Satoru Iwata, Lisa Fleischer, and Satoru Fujishige. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *Journal of the ACM (JACM)*, 48(4):761–777, 2001. 2

[IMH82]   Oscar H Ibarra, Shlomo Moran, and Roger Hui. A generalization of the fast LUP matrix decomposition algorithm and applications. *Journal of Algorithms*, 3(1):45–56, March 1982. 10, 17

[JSWZ20]   Shunhua Jiang, Zhao Song, Omri Weinstein, and Hengjie Zhang. Faster Dynamic Matrix Inverse for Faster LPs. *arXiv:2004.07470 [cs]*, April 2020. 3, 18, 19

[Kha79]   Leonid G Khachiyan. A polynomial algorithm in linear programming. In *Doklady Academii Nauk SSSR*, volume 244, pages 1093–1096, 1979. 2

[Kha95]   Leonid G. Khachiyan. On the complexity of approximating extremal determinants in matrices. *J. Complex.*, 11(1):138–153, March 1995. 7

[LS19]   Yin Tat Lee and Aaron Sidford. Solving linear programs with $\tilde{O}(\sqrt{\text{rank}})$ linear system solves. arXiv preprint 1910.08033, 2019. 3, 4, 18, 19, 34

[LSZ19]   Yin Tat Lee, Zhao Song, and Qiuyi Zhang. Solving empirical risk minimization in the current matrix multiplication time. In *COLT*, volume 99 of *Proceedings of Machine Learning Research*, pages 2140–2157, Phoenix, USA, 25–28 Jun 2019. PMLR. 3

[Meg83]   Nimrod Megiddo. Towards a genuinely polynomial algorithm for linear programming. *SIAM Journal on Computing*, 12(2):347–353, 1983. 2

[MT03]   Renato D. C. Monteiro and Takashi Tsuchiya. A variant of the Vavasis-Ye layered-step interior-point algorithm for linear programming. *SIAM Journal on Optimization*, 13(4):1054–1079, 2003. 3, 10

[MT05]   Renato D. C. Monteiro and Takashi Tsuchiya. A new iteration-complexity bound for the MTY predictor-corrector algorithm. *SIAM Journal on Optimization*, 15(2):319–347, 2005. 3

[MTY93]   Shinji Mizuno, Michael Todd, and Yinyu Ye. On adaptive-step primal-dual interior-point algorithms for linear programming. *Mathematics of Operations Research - MOR*, 18:964–981, 11 1993. 4

[O'L90]   Dianne P. O'Leary. On bounds for scaled projections and pseudoinverses. *Linear Algebra and its Applications*, 132:115–117, April 1990. 11

[Orl93]   J. B. Orlin. A faster strongly polynomial minimum cost flow algorithm. *Operations Research*, 41(2):338–350, 1993. 2

[OV17]   Neil Olver and László A. Végh. A simpler and faster strongly polynomial algorithm for generalized flow maximization. In *Proceedings of the Forty-Ninth Annual ACM Symposium on Theory of Computing (STOC)*, pages 100–111, 2017. 2

[PVZ20]   Javier Pena, Juan C Vera, and Luis F Zuluaga. New characterizations of hoffman constants for systems of linear constraints. *Mathematical Programming*, pages 1–31, 2020. 13

[Sch98]   Alexander Schrijver. *Theory of linear and integer programming*. John Wiley & Sons, 1998. 14

[Ste89]   G.W. Stewart. On scaled projections and pseudoinverses. *Linear Algebra and its Applications*, 112:189 – 193, 1989. 3, 10, 11

[Tar85]   Éva Tardos. A strongly polynomial minimum cost circulation algorithm. *Combinatorica*, 5(3):247–255, Sep 1985. 2, 5, 6, 14

[Tar86]   Éva Tardos. A strongly polynomial algorithm to solve combinatorial linear programs. *Operations Research*, pages 250–256, 1986. 2, 3

[Tod90]   Michael J. Todd. A Dantzig–Wolfe-like variant of Karmarkar's interior-point linear programming algorithm. *Operations Research*, 38(6):1006–1018, 1990. 10

[TTY01]   Michael J. Todd, Levent Tunçel, and Yinyu Ye. Characterizations, bounds, and probabilistic analysis of two complexity measures for linear programming problems. *Mathematical Programming*, 90(1):59–69, Mar 2001. 10

[Tun99]   Levent Tunçel. Approximating the complexity measure of Vavasis-Ye algorithm is NP-hard. *Mathematical Programming*, 86(1):219–223, Sep 1999. 7, 13

[Vav94]   Stephen A Vavasis. Stable numerical algorithms for equilibrium systems. *SIAM Journal on Matrix Analysis and Applications*, 15(4):1108–1131, 1994. 11, 12

[vdB20]   Jan van den Brand. A deterministic linear program solver in current matrix multiplication time. In *Proceedings of the Symposium on Discrete Algorithms (SODA)*, 2020. 3, 4, 5, 18, 19, 34

[vdBLSS20] Jan van den Brand, Yin Tat Lee, Aaron Sidford, and Zhao Song. Solving tall dense linear programs in nearly linear time, 2020. 3, 4, 18, 19, 34

[Vég17]   László A. Végh. A strongly polynomial algorithm for generalized flow maximization. *Mathematics of Operations Research*, 42(2):179–211, 2017. 2

[VY95]   Stephen A. Vavasis and Yinyu Ye. Condition numbers for polyhedra with real number data. *Operations Research Letters*, 17:209–214, 06 1995. 3

[VY96]   Stephen A. Vavasis and Yinyu Ye. A primal-dual interior point method whose running time depends only on the constraint matrix. *Mathematical Programming*, 74(1):79–120, 1996. 3, 7, 10, 32

[Ye05]   Yinyu Ye. A new complexity result on solving the Markov decision problem. *Mathematics of Operations Research*, 30(3):733–749, 2005. 2

[Ye06]   Yinyu Ye. Improved complexity results on solving real-number linear feasibility problems. *Mathematical Programming*, 106(2):339–363, April 2006. 3

[Ye11]   Yinyu Ye. The simplex and policy-iteration methods are strongly polynomial for the Markov decision problem with a fixed discount rate. *Mathematics of Operations Research*, 36(4):593–603, 2011. 2

[YTM94]   Yinyu Ye, Michael J. Todd, and Shinji Mizuno. An $O(\sqrt{n}L)$-iteration homogeneous and self-dual linear programming algorithm. *Mathematics of Operations Research*, 19(1):53–67, 1994. 32