Exact Linear Programming

Circuits, Curvature, and Diameter

by

Bento Natura

A thesis submitted for the degree of
Doctor of Philosophy
at
Department of Mathematics
The London School of Economics and Political Science

Acknowledgements

First and foremost, I thank my supervisor László A. Végh. For all the guidance and freedom that he gave me. For his mathematical advice, for so many inspiring research sessions and ideas. I am also thankful for his advice in navigating through the various stages of grad school and challenges in the last four years.

My utmost gratitude also goes out to my unofficial co-supervisor Daniel Dadush. For the many collaborations, for hosting me at CWI, the many hours of brainstorming, for shutting down bad ideas before they take roots, and for nurturing good ones.

Working with both Daniel and Laci in tandem is an amazing experience. The skills I acquired from learning from both of you will help me greatly going forward. I am looking forward to continuing to collaborate with you and edging a little closer to the holy grail.

Thanks to all my co-authors, without whom this thesis would not have been possible. Besides Daniel and Laci, I am especially grateful for the opportunity to work with Cedric, Daniel R., Farbod, Meike, Omri, Shunhua, Siad, Stefan, Stephan, and Xavier. I would be honored to work with you again.

I am thankful to the examiners Yin Tat Lee, Levent Tuncel, and Bernhard von Stengel for giving me comments and suggestions that have improved the presentation of the thesis, and for a stimulating discussion during the viva.

Further, I would like to thank Ahmad Abdi, Neil Olver, and Giacomo Zambelli. All of you have brought me great benefit in learning topics adjacent to my research and helped enjoy my time in the department. You and Laci alone make the Operations Research Group such an amazing place to be part of.

I would like to thank all the amazing people that make the Department of Mathematics at LSE such a great and welcoming environment. Credits are due to all the other PhD students and the administrative staff Kate Barker, Enfale Farooq, and Edward Perrin. A special thanks to Sarah Massey for so many things; let me representatively mention the collaboration on the blog and daily games.

Further thanks go to Stephan Held, Stefan Hougardy, and Jens Vygen at the Institute for Discrete Mathematics at the University of Bonn for sparking my interest in Discrete Mathematics and Optimization through exciting courses and projects throughout my Bachelor's and Master's. When I entered the university with solid foundations in algebra and analysis, I had never heard of the area in which I am now submitting this thesis. Another thanks to my Master's supervisor Stephan for encouraging me to pursue a PhD; it turned out to be the correct decision.

I am further thankful to an institute affiliated with the University of Bonn, the Hausdorff Institute of Mathematics, for hosting me at the trimester "Discrete Optimization", during which the research for one of the chapters of this thesis was undertaken.

I am grateful to all the anonymous reviewers for conferences and journals that helped to improve the presentation of the papers that became the backbone of this thesis.

These days we take LaTeX for typesetting for granted. But it is an immense success story of the open-source community, which still actively develops and improves packages and features. Special thanks go to Thomas Colcombet for developing `knowledge`, to Bastian Rieck for `mimosis`, and to R Schlicht for developing `microtype`.

Throughout my degree I was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement ScaleOpt–757481).

I thank everybody involved with "James Joyce", Lucile, Unlucky and all my other friends. To my family, I am forever grateful for their unconditional love and encouragement.

Declaration

I certify that the thesis I have presented for examination for the PhD degree of the London School of Economics and Political Science is solely my own work, with the exceptions outlined in the section below.

The copyright of this thesis rests with the author. Quotation from it is permitted, provided that full acknowledgement is made. In accordance with the regulations, I have deposited an electronic copy of it in LSE Theses Online held by the British Library of Political and Economic Science and have granted permission for my thesis to be made available for public reference. Otherwise, this thesis may not be reproduced without my prior written consent. I warrant that this authorisation does not, to the best of my belief, infringe the rights of any third party.

I declare that this thesis consists of 76,384 words.

Statement of co-authored work

Chapter 3 is based on [ENV22], Chapter 4 is based on [DHNV20], Chapter 5 is based on [All+22], Chapter 7 is based on [DNV20], Chapter 8 is based on [DKNV22; ENV22], and Chapter 9 is based on [DKNV21].

The contents of Chapter 6 and Section 9.3 are novel unpublished work.

[All+22] X. Allamigeon, D. Dadush, G. Loho, B. Natura, and L. A. Végh. *Interior point methods are not worse than Simplex*. To appear in Annual Symposium on Foundations of Computer Science (FOCS 2022). 2022. URL: https://arxiv.org/abs/2206.08810.

[DHNV20] D. Dadush, S. Huiberts, B. Natura, and L. 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 SIGACT Symposium on Theory of Computing*. STOC 2020. Association for Computing Machinery, Chicago, IL, USA, 2020, pp. 761–774. ISBN: 9781450369794. URL: https://doi.org/10.1145/3357713.3384326.

[DKNV21] D. Dadush, Z. K. Koh, B. Natura, and L. A. Végh. "An Accelerated Newton-Dinkelbach Method and Its Application to Two Variables per Inequality Systems". In: *29th Annual European Symposium on Algorithms (ESA 2021)*. Ed. by P. Mutzel, R. Pagh, and G. Herman. Vol. 204. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2021, 36:1–36:15. ISBN: 978-3-95977-204-4. URL: https://drops.dagstuhl.de/opus/volltexte/2021/14617.

[DKNV22] D. Dadush, Z. K. Koh, B. Natura, and L. A. Végh. "On Circuit Diameter Bounds via Circuit Imbalances". In: *Integer Programming and Combinatorial Optimization*. Springer International Publishing, 2022, pp. 140–153. URL: https://doi.org/10.1007/978-3-031-06901-7_11.

[DNV20] D. Dadush, B. Natura, and L. A. Végh. "Revisiting Tardos's Framework for Linear Programming: Faster Exact Solutions using Approximate Solvers". In: *61st IEEE Annual Symposium on Foundations of Computer Science, FOCS 2020, Durham, NC, USA, November 16-19, 2020*. IEEE, 2020, pp. 931–942. URL: https://doi.org/10.1109/FOCS46700.2020.00091.

[ENV22] F. Ekbatani, B. Natura, and L. A. Végh. "Circuit Imbalance Measures and Linear Programming". In: *Surveys in Combinatorics 2022*. London Mathematical Society Lecture Note Series. Cambridge University Press, 2022, pp. 64–114.

During my degree, I have also -authored the following published papers that are not part of this thesis:

[DHNR19] S. Daboul, S. Held, B. Natura, and D. Rotter. "Global Interconnect Optimization". In: *2019 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*. 2019, pp. 1–8.

[JNW22] S. Jiang, B. Natura, and O. Weinstein. "A Faster Interior-Point Method for Sum-Of-Squares Optimization". In: *49th International Colloquium on Automata, Languages, and Programming (ICALP 2022)*. Ed. by M. Bojańczyk, E. Merelli, and D. P. Woodruff. Vol. 229. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2022, 79:1–79:20. ISBN: 978-3-95977-235-8. URL: https://drops.dagstuhl.de/opus/volltexte/2022/16420.

[NNW22] B. Natura, M. Neuwohner, and S. Weltge. *The Pareto cover problem*. In Annual European Symposium on Algorithms (ESA 2022). 2022. URL: https://arxiv.org/abs/2202.08035.

Bento Natura

Abstract

We study Linear Programming (LP) and present novel algorithms. In particular, we study LP in the context of circuits, which are support-minimal vectors of linear spaces. Our results will be stated in terms of the circuit imbalance (CI), which is the worst-case ratio of nonzero entries of circuits and whose properties we study in detail. We present following results with logarithmic dependency on CI. (i) A scaling-invariant Interior-Point Method, which solves LP in time that is polynomial in the dimensions, answering an open question by Monteiro-Tsuchiya in the affirmative. This closes a long line of work by Vavasis-Ye and Monteiro-Tsuchiya; (ii) We introduce a new polynomial-time path-following interior point method where the number of iterations admits a singly exponential upper bound. This complements recent results, that path-following method must take at least exponentially many iterations; (iii) We further provide similar upper bounds on a natural notion of curvature of the central path; (iv) A black-box algorithm that requires only quadratically many calls to an approximate LP solver to solve LP exactly. This significantly strengthens the framework by Tardos, which requires exact solvers and whose runtime is logarithmic in the maximum subdeterminant of the constraint matrix. The maximum subdeterminant is exponentially bigger than CI, already for fundamental combinatorial problems such as matchings; (v) Furthermore, we obtain a circuit diameter that is quadratic in the number of variables, giving the first polynomial bound for general LP where CI is exponential. Unlike in the simplex method, one does not have to augment around the edges of the polyhedron: Augmentations can be in any circuit direction; (vi) Lastly, we present an accelerated version of the Newton–Dinkelbach method, which extends the black-box framework to certain classes of fractional and parametric optimization problems. Using the Bregman divergence as a potential in conjunction with combinatorial arguments, we obtain improved runtimes over the non-accelerated version.

Contents

1 Introduction

> In real life, when emotions and sentiments are involved and the very continuity of life is at stake, there are no quantitative theories, linear programming, and applied mechanics available to solve those problems.
>
> *Some Mistakes Have No Pardon*
> Girdhar Joshi

Fortunately, we do not study real life here. Even if this thesis is submitted for the degree of Doctor of Philosophy, we do not make an attempt at discussing the relation of the most important real life problems to emotions and sentiments. The study of LP can still solve various problems, even if they might be unrelated to emotions, sentiments, and the context of real life. This does not degrade their importance in the world.

> Linear programming is viewed as a revolutionary development giving man the ability to state general objectives and to find, by means of the simplex method, optimal policy decisions for a broad class of practical decision problems of great complexity.
>
> (George Dantzig)

This is the positive perspective we need. From here, we focus on the mathematics of LP.

Linear Programming (LP) in matrix formulation. We will use LPs in the following *standard* primal and dual form for $\mathbf{A} \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.

System 1.1. *Linear Program (LP)*

Data: A triple $(\mathbf{A}, b, c)$, where $\mathbf{A}$ is an $m \times n$ matrix and $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$.

Primal

$$\begin{aligned} \min_{x} \quad & \langle c, x \rangle \\ \text{s.t.} \quad & \mathbf{A}x = b, \\ & x \geq \mathbf{0} \end{aligned}$$

Dual

$$\begin{aligned} \max_{y,s} \quad & \langle y, b \rangle \\ \text{s.t.} \quad & \mathbf{A}^\top y + s = c, \\ & s \geq \mathbf{0} \end{aligned}$$

Linear Programming in subspace formulation Since our main focus is on properties of subspaces, it will be more natural to think about linear programming in the following *subspace formulation*. For $\mathbf{A}$, b and c as above, let $W = \ker(\mathbf{A}) \subseteq \mathbb{R}^n$. We can write System 1.1 in the following equivalent form, where d is a vector fulfilling $\mathbf{A}d = b$:

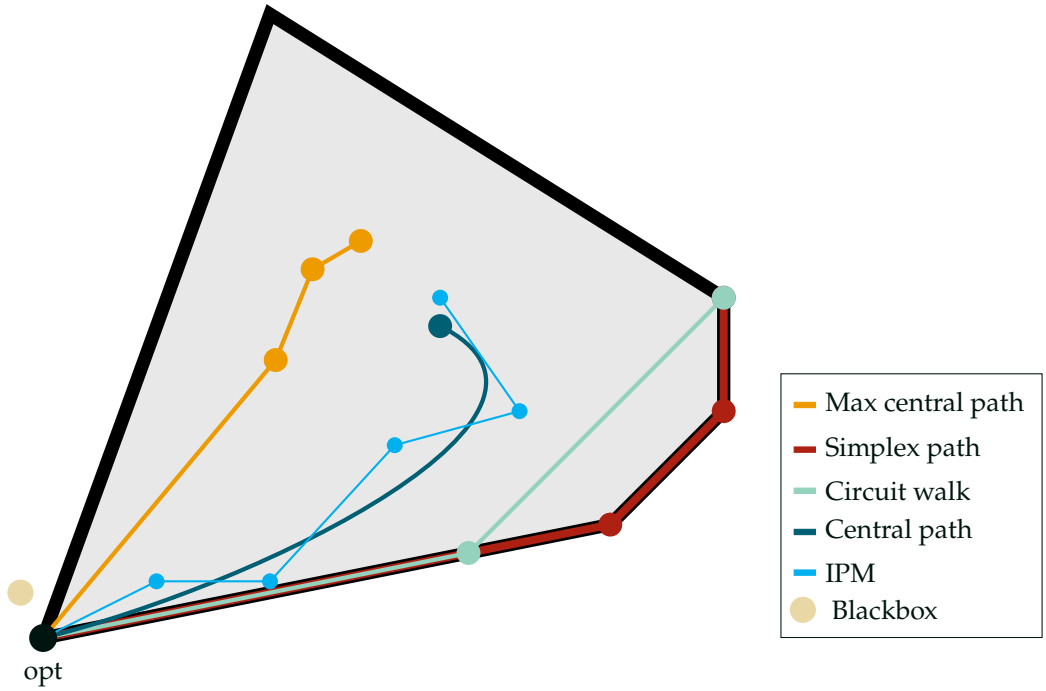


Figure 1.1: The several geometric objects considered in the thesis.

System 1.2. *Linear Program (LP) – subspace formulation*

Data: A triple (W, d, c), where $W \subseteq \mathbb{R}^n$ is a subspace, and $d, c \in \mathbb{R}^n$.

Primal

$$\begin{aligned} \min_x \quad & \langle c, x \rangle \\ \text{s.t.} \quad & x \in W + d, \\ & x \geq \mathbf{0} \end{aligned}$$

Dual

$$\begin{aligned} \max_s \quad & \langle d, c - s \rangle \\ \text{s.t.} \quad & s \in W^\perp + c, \\ & s \geq \mathbf{0} \end{aligned}$$

In this thesis we present several exact algorithms with varying methods, as depicted in Figure 1.1. For the primal and dual feasible regions we write

$$\begin{aligned} \mathcal{P} &:= \{ x \in \mathbb{R}^n : \mathbf{A}x = b, x \geq \mathbf{0} \} = \{ x \in \mathbb{R}^n : x \in W + d, x \geq \mathbf{0} \} \quad \text{and} \\ \mathcal{D} &:= \left\{ s \in \mathbb{R}^n : \exists y \colon \mathbf{A}^\top y + s = c \,,\ s \geq \mathbf{0} \right\} = \left\{ s \in \mathbb{R}^n : s \in W^\perp + c, s \geq \mathbf{0} \right\}. \end{aligned} \tag{1.1}$$

An outstanding open question is the existence of a strongly polynomial algorithm for LP, listed by Smale as one of the most prominent mathematical challenges for the 21st century [Sma98]. Such an algorithm amounts to solving LP using $\text{poly}(n, m)$ basic arithmetic operations in the *real model of computation*.[1] Known strongly polynomially solvable LP problems classes include: feasibility for two variable per inequality systems [Meg83], the minimum-cost circulation problem [Tar85], the maximum generalized flow problem [OV20; Vég17], and discounted Markov decision problems [Ye05; Ye11].

[1] In the bit-complexity model, a further requirement is that the algorithm must be in PSPACE.

Algorithmic Perspective

Linear Programming has a long history and is a central tool in optimization in both theory and practice. The study of LP dates back at least as far as the early 19th century to Fourier. In the mid 20th century it was formalized by Kantorovich. Shortly after, the first efficient algorithms were developed.

The Simplex Method. The *simplex method* is the oldest of the practical algorithms for LP and was invented by Dantzig in 1947. Its fundamental role earned it a place in a list of the top 10 algorithms of the 20th century [Nas00]. The simplex method traverses a path formed by vertices and edges of $\mathcal{P}$ according to a certain *pivot rule*. Albeit efficient in practice, there is no polynomial-time variant known, and there are exponential worst case examples for several pivot rules. The first such construction was given by Klee and Minty [KM72] for Dantzig's pivot rule. While the simplex method may be exponential, it is never worse: for any non-cycling pivot rule, the number of pivot steps can be bounded by the number of bases, at most $\binom{n}{m} < 2^n$.

The Ellipsoid Method. Khachiyan [Kha79] used the *ellipsoid method* to give the first polynomial time LP algorithm in the bit-complexity model, that is, polynomial in the bit description length of $(\mathbf{A}, b, c)$.

On a high level, the ellipsoid method aims to find a feasible solution in a polytope, by encircling the feasible region by an ellipsoid. If the center of the ellipsoid is not contained in the polytope, then a *separation oracle* provides a *separating hyperplane*, which allows cutting off a part of the ellipsoid to recurse with an ellipsoid of smaller size that still fully contains the feasible region. These methods are called *cutting plane methods*.

The excellent performance of cutting plane methods in theory led to a hype, even though people were not immediately able to translate the theoretical guarantees into a practical algorithm able to compete with the by then well-established simplex method. It remains a challenge today to design practical cutting plane methods. In theory, more breakthroughs were achieved in recent years [JLSW20; LSW15].

Interior Point Methods. The advent of Interior-Point-Methods (IPM) brought the first class of algorithms for LP that performs well in both, theory and practice. Karmarkar [Kar84] was the first to devise such a method in 1984. Contrary to the simplex method, it solves LP by traversing the interior of the polytope. IPM follow a central path which corresponds to the set of optimal solutions to problems of the form $\min \langle c, x\rangle + \mu F(x), x \in \mathcal{P}$, where $F(x)$ is a so-called barrier function defined which penalizes getting to close the boundary of $\mathcal{P}$. A standard barrier for LP is the log-barrier, whose so called central path is the set of solutions defined in (CP), parametrized by μ.

$$\begin{aligned} x(\mu)_i s(\mu)_i &= \mu, \quad \forall i \in [n] \\ \mathbf{A}x(\mu) &= b,\ x(\mu) > \mathbf{0}, \\ \mathbf{A}^\top y(\mu) + s(\mu) &= c,\ s(\mu) > \mathbf{0}, \end{aligned} \tag{CP}$$

IPM are at the heart of recent breakthrough results for LP, see [Bra20; Che+22; CLS19; LS19] to only name a few which appeared during the author's PhD studies.

Geometric Perspective

Related to algorithms, but allowing for separate study, are certain geometric properties of LP.

The *combinatorial diameter* of a polyhedron $P \subseteq \mathbb{R}^d$ is the diameter of the vertex-edge graph associated with P. Hirsch's famous conjecture from 1957 asserted that the combinatorial diameter of a polytope (a bounded polyhedron) in d dimensions with n facets is at most $n - d$. This was disproved by Santos in 2012 [San12]. The polynomial Hirsch conjecture, i.e., finding a poly(n, d) bound on the combinatorial diameter remains a central question in the theory of linear programming. In particular is such a bound necessary to devise a strongly polynomial simplex algorithm.

The first quasipolynomial bound was given by Kalai and Kleitman [Kal92; KK92], see [Suk17] for the best current bound and an overview of the literature. Dyer and Frieze [DF94] showed the polynomial Hirsch conjecture for Totally Unimodular (TU) matrices.

As a natural relaxation of the combinatorial diameter, Borgwardt, Finhold, and Hemmecke [BFH15] initiated the study of circuit diameters. Consider a polytope in the standard equality form $\mathcal{P}$. All edge directions of $\mathcal{P}$ are elementary vectors, and the set of elementary vectors $\mathcal{E}(\mathbf{A})$ equals the set of all possible edge directions of $\mathcal{P}$ for varying $b \in \mathbb{R}^n$ [ST97].

A *circuit walk* is a set of consecutive points $x^{(1)}, x^{(2)}, \dots, x^{(k+1)} \in \mathcal{P}$ such that for each $i = 1, \dots, k$, $x^{(i+1)} = x^{(i)} + g^{(i)}$ for $g^{(i)} \in \mathcal{E}(\mathbf{A})$, and further, $x^{(i)} + (1+\varepsilon)g^{(i)} \notin \mathcal{P}$ for any $\varepsilon > 0$, i.e., each consecutive circuit step is *maximal*. The *circuit diameter* of $\mathcal{P}$ is the minimum length of a circuit walk between any two vertices $x, y \in \mathcal{P}$. Note that, in contrast to walks in the vertex-edge graph, circuit walks are non-reversible and the minimum length from x to y may be different from the one from y to x; this is due to the maximality requirement. The circuit-analogue of the Hirsch conjecture, formulated in [BFH15], asserts that the circuit diameter of a polytope in d dimensions with n facets is at most $n - d$; this may be true even for unbounded polyhedra, see [BSY18]. For $\mathcal{P}$, $d = n - m$ and hence the conjectured bound is m.

Circuit diameter bounds have been shown for some combinatorial polytopes such as dual transportation polyhedra [BFH15], matching, travelling salesman, and fractional stable set polytopes [KPS19]. The paper [BDF16] introduced several other variants of the circuit diameter, and explored the relation between them.

Another geometric object that has been studied is the central path and a natural notion of its curvature [SSZ91], defined as

$$\mathcal{I}(0, \infty) := \int_0^\infty \frac{\Upsilon(\nu)}{\nu} \, \mathrm{d}\nu, \quad \Upsilon(\nu) := \sqrt{\|\nu \dot{x}(\nu) \dot{s}(\nu)\|}\,. \tag{1.2}$$

It was related very closely tied to the number of iterations an IPM requires if the error one allows while approximately following the paths converges to 0 [MT08].

Condition numbers of the constraint matrix

Towards the goal of finding more classes of LP that are strongly polynomially solvable, the principal line of attack has been to develop LP algorithms whose running time is bounded in terms of natural *condition measures*. Such condition measures attempt to measure the "intrinsic complexity" of LPs. An important line of work in this area has been to parametrize LPs by the "niceness" of their solutions (e.g., the depth of the most interior point), where relevant examples include the Goffin measure [Gof80] for conic systems and Renegar's distance to ill-posedness for general LPs [Ren94; Ren95], and bounded ratios between the nonzero entries in basic feasible solutions [Chu14; KM13].

Parametrizing by the constraint matrix. A second line of research, and one of the main topics of this thesis, focuses on the complexity of the constraint matrix $\mathbf{A}$. In a seminal work, Vavasis

and Ye [VY96] introduced a new type of interior-point method that optimally solves System 1.1 within $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}}+n))$ iterations, where the condition number $\bar{\chi}_{\mathbf{A}}$ controls the size of solutions to certain linear systems related to the kernel of $\mathbf{A}$ (see Section 4.2 for the formal definition).

Monteiro and Tsuchiya [MT03], noting that the central path is invariant under rescalings of the columns of $\mathbf{A}$ and c, asked whether there exists an LP algorithm depending instead on the measure $\bar{\chi}^*_{\mathbf{A}}$, defined as the minimum $\bar{\chi}_{\mathbf{AD}}$ value achievable by a column rescaling $\mathbf{AD}$ of $\mathbf{A}$, and gave strong evidence that this should be the case. We resolve this open question affirmatively.

Theorem I (Theorem 3.4.7)**.** *There exists a strongly polynomial algorithm, that given a matrix* $\mathbf{A}$ *computes a rescaling* $\mathbf{D}$ *such that* $\bar{\chi}_{\mathbf{AD}} \leq \text{poly}(n, \bar{\chi}^*_{\mathbf{A}})$.

This result is in surprising contrast to that of Tunçel [Tun99], who showed NP-hardness for approximating $\bar{\chi}_{\mathbf{A}}$ to within $2^{\text{poly}(\text{rk}(\mathbf{A}))}$.

While this resolves Monteiro and Tsuchiya's question by appropriate preprocessing, it falls short of providing either a truly *scaling invariant* algorithm or an improvement upon the base Layered-Least-Squares (LLS) analysis. In this vein, as a second main contribution in this theory we develop a scaling invariant LLS algorithm.

Theorem II (Theorem 4.3.16)**.** *There exists a scaling-invariant IPM that solves LP in* $\widetilde{O}(n^{2.5}\log(\bar{\chi}^*_{\mathbf{A}}))$ *iterations, each of which can be computed in strongly polynomial* $\text{poly}(n, m)$ *time.*

Besides being scaling-invariant and hence depending on $\bar{\chi}^*$ instead of $\bar{\chi}$, this also constitutes a $n/\log n$ improvement on the iteration complexity bound of the original Vavasis-Ye Algorithm in [VY96] (VY) algorithm and follow-up works by Monteiro and Tsuchiya.

As other IPM our algorithm follows very closely the central path (CP) induced by the barrier. In particular, matching the result on the number of iterations of the VY algorithm Monteiro and Tsuchiya bounded the total curvature of the central path by $O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}}+n))$. We show that their result can naturally be improved based on our improvement on the number of iterations in our scaling-invariant IPM.

Theorem III (Theorem 6.3.3)**.** *The total curvature of the central path is bounded by* $\mathcal{I}(0,\infty) = \widetilde{O}(n^{2.5}\log\bar{\chi}^*_{\mathbf{A}})$.

However, while this results in an IPM with logarithmic dependency on $\log\bar{\chi}^*$, they do not admit even an exponential dependency on the dimensions m and n only, as $\bar{\chi}^*$ can be arbitrarily big even in fixed dimension. To the extent of our knowledge, no variant of the ellipsoid or interior point methods have been shown to admit a bound $f(n)$ on the number of iterations for any function $f:\mathbb{N}\to\mathbb{N}$ prior to 2022. We design an IPM that finally admits an exponential iteration bound.

Theorem IV (Informal, Theorem 5.1.2)**.** *There exists an IPM that solves LP within* $O(n^{1.5}2^n\log(n))$ *iterations. Each iteration can be implemented in strongly polynomial time* $\text{poly}(n, m)$.

Moreover, the running time of the algorithm is within a polynomial factor of any path following method.

We also show how this algorithmic bound can be translated into a purely geometric bound on the curvature of the central path.

Theorem V (Theorem 6.2.4)**.** *The total curvature of the central path is bounded by* $\mathcal{I}(0,\infty) = O(n^{1.5}2^n\log(n))$.

Proximity– based solvers. The first breakthrough in the area of conditioned matrices predates Vavasis and Ye by 10 years and was achieved by Tardos [Tar86]. She showed that if $\mathbf{A}$ has integer entries and all square submatrices of $\mathbf{A}$ have determinant at most Δ in absolute value, then System 1.1 can be solved in $\text{poly}(n, m, \log \Delta)$ arithmetic operations, independent of the encoding length of the vectors b and c. At the heart of this approach lie proximity results. Such proximity results allow to deduce information about optimal solutions to LP based on nearly-optimal solutions to slightly perturbed problems. We introduce a new condition number, $\kappa_{\mathbf{A}}$, which is the largest ratio of two non-zero coordinates in support-minimal elements in the kernel of $\mathbf{A}$, which fulfills $\kappa_{\mathbf{A}} \leq \Delta_{\mathbf{A}}$ for all $\mathbf{A} \in \mathbb{Z}^{m \times n}$. We provide an in-depth study of its properties as well as related condition numbers in Chapter 3 and extend Tardos' results to $\kappa_{\mathbf{A}}$, while requiring weaker oracle assumptions.

Theorem VI (Informal, Theorem 7.8.1)**.** *There exist algorithms that for feasible primal-dual instances of LP can find*

(i) a feasible point within m calls to any approximate solver to LP with precision $(n\kappa)^{-O(1)}$, and

(ii) an optimal point within mn calls to any approximate solver to LP with precision $(n\kappa)^{-O(1)}$.

The running time is dominated by the calls to the approximate solvers.

For the dual $\mathcal{D}$ with integer constraint matrix $\mathbf{A}$, polynomial diameter bounds were given in terms of the maximum subdeterminant $\Delta_{\mathbf{A}}$ [Bon+14; BR13; DH16; EV17]. These arguments can be strengthened to using a parametrization by a 'discrete curvature measure' $\delta_{\mathbf{A}^\top} \geq 1/(n\Delta_{\mathbf{A}}^2)$. The best such bound was given by Dadush and Hähnle [DH16] as $O(d^3 \log(d/\delta_{\mathbf{A}^\top})/\delta_{\mathbf{A}^\top})$, using a shadow vertex simplex algorithm.

By relating $\delta_{\mathbf{A}^\top}$ to $\kappa_{\mathbf{A}}$ we are able to show that the diameter is also polynomially bounded in $\kappa_{\mathbf{A}}$.

Theorem VII (Theorem 3.3.12)**.** *The diameter of $\mathcal{P}$ is bounded by $O(n^3 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$.*

Unlike for the simplex method, it can be shown that there is always a circuit direction which relative to the optimal solution improves the objective geometrically. Similarly to the blackbox approach, this can be used to show constructive bounds on the circuit diameter.

Theorem VIII (Theorem 8.1.4)**.** *The circuit diameter of $\mathcal{P}$ is $O(m^2 \log(\kappa_{\mathbf{A}} + n))$.*

Interestingly, we can extend these results to capacitated LP formulations without significantly increasing the diameter.

System 1.3. *Capacitated Linear Program (CLP)*

Data: A quadruple $(\mathbf{A}, b, c, u)$, where $\mathbf{A}$ is an $m \times n$ matrix and $b \in \mathbb{R}^m$ and $c, u \in \mathbb{R}^n$.

Capacitated Primal

$$\begin{aligned} \min_{x} \quad & \langle c, x \rangle \\ \text{s.t.} \quad & \mathbf{A}x = b, \\ & \mathbf{0} \leq x \leq u \end{aligned}$$

Capacitated Dual

$$\begin{aligned} \max_{y,s} \quad & \langle y, b \rangle - \langle u, t \rangle \\ \text{s.t.} \quad & \mathbf{A}^\top y + s - t = c, \\ & s, t \geq \mathbf{0} \end{aligned}$$

Theorem IX (Theorem 8.1.5)**.** *The circuit diameter of the feasible region in Capacitated-Primal is $O(m^2 \log(\kappa_{\mathbf{A}} + n) + n \log(n))$.*

Beyond Linear Programs

System 1.4. *Fractional Linear Program (FLP)*

Data: A tuple $(\mathbf{A}, b, c, q)$, where $\mathbf{A}$ is an $m \times n$ matrix and $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$, $d \in \mathbb{R}^n$ and $\langle q, x \rangle > 0$ for all $x \in$ Primal$(\mathbf{A}, b)$.

$$\min_x \frac{\langle c, x \rangle}{\langle q, x \rangle} \quad \text{s.t.} \quad \mathbf{A}x = b, \quad x \geq \mathbf{0}\,. \tag{1.3}$$

In the last chapter of this thesis we look into Fractional Linear Programming. The discrete Newton method solves Fractional LP by reducing it to a sequence of LP with objective function $\langle c, x \rangle - \lambda \langle q, x \rangle, \lambda \in \mathbb{R}$. We develop a novel accelerated discrete Newton method, for which we show improved running times and simplified analyses for problems such as linear fractional combinatorial optimization and parametric submodular function minimization.

We also develop a proximity-based solver for Fractional LP in terms of $\kappa_{\mathbf{A}}$.

Theorem X (Theorem 9.3.1)**.** *System 1.4 can be solved within $O(n \log \kappa_{\mathbf{A}})$ calls to an exact solver to LP with the same constraint matrix $\mathbf{A}$.*

2 Preliminaries

We let $\mathbb{N} = \{1, 2, \ldots\}$ denote the positive integers, $\mathbb{Z}$ the integers, $\mathbb{Q}$ the rational numbers and $\mathbb{R}$ the real numbers. Further, let $\overline{\mathbb{R}} \coloneqq \mathbb{R} \cup \{\pm\infty\}$, $\mathbb{R}_{++}$ the set of positive reals, and $\mathbb{R}_{+}$ the set of nonnegative reals. We let $[n] \coloneqq \{1, \ldots, n\}$. For $k \in \mathbb{N}$, a number $q \in \mathbb{Q}$ is $1/k$-*integral* if it is an integer multiple of $1/k$. Let $\mathbb{P} \subseteq \mathbb{N}$ denote the set of primes.

For a prime number $p \in \mathbb{P}$, the *p-adic valuation* for $\mathbb{Z}$ is the function $\nu_p \colon \mathbb{Z} \to \mathbb{N} \cup \{\infty\}$ defined by

$$\nu_p(n) \coloneqq \begin{cases} \max\{\, v \in \mathbb{N} : p^v \mid n \,\} & \text{if } n \neq 0 \\ \infty & \text{if } n = 0. \end{cases} \tag{2.1}$$

Linear Algebra. We denote the support of a vector $x \in \mathbb{R}^n$ by $\operatorname{supp}(x) \coloneqq \{i \in [n] : x_i \neq 0\}$. We let $\mathbf{1}_n$ denote the n-dimensional all-ones vector, or simply $\mathbf{1}$, whenever the dimension is clear from the context. Analogously $\mathbf{0}_n$ (and $\mathbf{0}$) are the all-zero vectors. Let e^i denote the i-th vector in the standard basis.

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 $v^+ \coloneqq \max\{v, \mathbf{0}_n\}$ and $v^- \coloneqq \max\{-v, \mathbf{0}_n\}$; note that both v^+ and v^- are nonnegative vectors. For two vectors $x, y \in \mathbb{R}^n$, we let $\langle x, y \rangle = x^\top y$ denote their scalar product and we let $[x, y] = \{\, (1-\lambda)x + \lambda y : 0 \leq \lambda \leq 1 \,\}$ denote the line-segment connecting x and y. We use the notation $xy \in \mathbb{R}^n$ to denote the *Hadamard product* $xy = (x_i y_i)_{i \in [n]}$. Further, with $p \in \mathbb{Q}$, we also use the notation x^p to denote the vector $(x_i^p)_{i \in [n]}$. Similarly, for $x \in \mathbb{R}^n, y \in (\mathbb{R} \setminus \{0\})^n$, we let x/y denote the vector $(x_i/y_i)_{i \in [n]}$.

For sets $S, T \subseteq \mathbb{R}$ we let $S \cdot T = \{\, st : s \in S, t \in T \,\}$.

We let $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ denote the n-dimensional identity matrix and $\mathfrak{D}_n$ denote the set of all positive definite $n \times n$ diagonal matrices. For a vector $v \in \mathbb{R}^n$, we denote by $\operatorname{diag}(v)$ the diagonal matrix whose i-th diagonal entry is v_i.

For a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, let $\mathbf{A}_1, \mathbf{A}_2, \ldots, \mathbf{A}_n \in \mathbb{R}^n$ denote the column vectors, and $\mathbf{A}^1, \mathbf{A}^2, \ldots, \mathbf{A}^m \in \mathbb{R}^m$ denote the transposed row vectors.

For $I \subseteq [m]$ and $J \subseteq [n]$ we let $\mathbf{A}_{I,J}$ denote the submatrix of $\mathbf{A}$ restricted to the set of rows in I and columns in J. We also use $\mathbf{A}_{I,\bullet} = \mathbf{A}_{I,[n]}$ and $\mathbf{A}_J = \mathbf{A}_{\bullet,J} = \mathbf{A}_{[m],J}$. For $i \in [n]$, we set $\mathbf{A}_{\leq i} \coloneqq \mathbf{A}_{\{j \in [n] : j \leq i\}}$ and $\mathbf{A}_{\geq i} \coloneqq \mathbf{A}_{\{j \in [n] : j \geq i\}}$. We define $\sigma_i(\mathbf{A})$ to be the i-th smallest singular value of $\mathbf{A}$. We let $\mathbf{A}^+ \in \mathbb{R}^{n \times m}$ denote the Moore–Penrose *pseudo-inverse* of $\mathbf{A}$. The matrix $\mathbf{A}$ is said to be in *basis form* for a basis B if $\mathbf{A}_B = \mathbf{I}_m$. We denote by $\operatorname{rk}(\mathbf{A})$ its *rank*, by $\ker(\mathbf{A})$ its *kernel* and by $\operatorname{im}(\mathbf{A})$ its *image*. Analogously, let $\operatorname{range}(\cdot)$ denote the range of a matrix or an operator.

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 $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\|\mathbf{A}\|$ will refer to the $\ell_2 \to \ell_2$ operator norm, $\|\mathbf{A}\|_F = \sqrt{\sum_{i,j} |\mathbf{A}_{ij}|^2}$ to the *Frobenius norm*, and $\|\mathbf{A}\|_{\max} = \max_{i,j} |\mathbf{A}_{ij}|$ to the *max-norm*.

For an index subset $I \subseteq [n]$, we use $\pi_I \colon \mathbb{R}^n \to \mathbb{R}^I$ 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 $\mathbb{R}^n_I = \left\{ x \in \mathbb{R}^n : x_{[n] \setminus I} = \mathbf{0} \right\}$.

For a subspace $W \subseteq \mathbb{R}^n$, we let $W_I = \pi_I(W \cap \mathbb{R}^n_I)$.

We will often use the following identity:

Proposition 2.0.1. *For $\emptyset \neq I \subseteq [n]$ and a linear subspace $W \subseteq \mathbb{R}^n$, $\pi_I(W)^\perp = \pi_I(W^\perp \cap \mathbb{R}^n_I)$ holds.*

For a subspace $W \subseteq \mathbb{R}^n$, we define by $\Pi_W : \mathbb{R}^n \to \mathbb{R}^n$ the *orthogonal projection* onto W and let $W_+ := W \cap \mathbb{R}^n_+$.

For a vector $x \in \mathbb{R}^n$ and a real threshold $\tau \in \mathbb{R}_+$ define the *essential support* $\operatorname{ess}(x, \tau) \in \mathbb{R}^n$ as

$$\operatorname{ess}(x,\tau)_i := \begin{cases} x_i & \text{if } |x_i| \geq \tau, \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

Given a concave function $f : \mathbb{R} \to \overline{\mathbb{R}}$, let $\operatorname{dom}(f) := \{ x : -\infty < f(x) < \infty \}$ be the *effective domain* of f. For a point $x_0 \in \operatorname{dom}(f)$, denote the set of *supergradients* of f at x_0 as $\partial f(x_0) := \{ g : f(x) \leq f(x_0) + g(x - x_0)\ \forall x \in \mathbb{R} \}$. If x_0 is in the interior of $\operatorname{dom}(f)$, then $\partial f(x_0) = [f'_-(x_0), f'_+(x_0)]$, where $f'_-(x_0)$ and $f'_+(x_0)$ are the left and right derivatives.

The linear spaces $\{0\}$ and $\mathbb{R}^n$ will be called *trivial subspaces*; all other subspaces are *nontrivial*. A linear subspace of $\mathbb{R}^n$ is a *rational linear space* if it admits a basis of rational vectors. Equivalently, a rational linear space can be represented as the image of a rational matrix. For an integer vector $v \in \mathbb{Z}^n$, let $\operatorname{lcm}(v)$ denote the *least common multiple* and $\gcd(v)$ denote the *greatest common divisor* of the entries $|v_i|$, $i \in [n]$.

For a set of vectors $V \subseteq \mathbb{R}^n$ we let $\operatorname{span}(V)$ denote the linear space spanned by the vectors in V. For a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\operatorname{span}(\mathbf{A}) \subseteq \mathbb{R}^m$ is the subspace spanned by the columns of $\mathbf{A}$.

Matroids. A *matroid* on a finite ground set E is given as $\mathcal{M} = (E, \mathcal{I})$, where $\mathcal{I} \subseteq 2^E$ is a nonempty collection of independent sets. This collection is required to satisfy the *independence axioms*:

(I1) *Non-Emptiness:* $\emptyset \in \mathcal{I}$.

(I2) *Monotonicity:* if $X \in \mathcal{I}$ then $Y \in \mathcal{I}$ for all $Y \subseteq X$, and

(I3) *Exchange property:* if $X, Y \in \mathcal{I}$, $|X| < |Y|$, then there exists a $y \in Y \setminus X$ such that $X \cup \{y\} \in \mathcal{I}$.

Bases and Circuits. For a linear subspace $W \subseteq \mathbb{R}^n$ and a matrix $\mathbf{A}$ such that $W = \ker(\mathbf{A})$, a *circuit* $C \subseteq [n]$ is an inclusion-wise minimal dependent set of columns of $\mathbf{A}$. Analogously, a *basis* $B \subseteq [n]$ is an inclusion-wise maximal independent set of columns of $\mathbf{A}$.

These notions only depend on the subspace W, and not on the particular representation $\mathbf{A}$; 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. Equivalently, B is a basis if $W \cap \mathbb{R}^n_B = \{\mathbf{0}\}$, but no strict superset of B has this property. The set of bases of W is denoted by $\mathcal{B}_W$. The set of circuits of W is denoted by $\mathcal{C}_W$.

For a subset $I \subseteq [n]$, we let $\operatorname{cl}(I)$ denote its *closure* in the matroidal sense.

That is, $\operatorname{cl}(I) = J$ is the unique maximal set containing $J \supseteq I$ such that $\operatorname{rk}(\mathbf{A}_J) = \operatorname{rk}(\mathbf{A}_I)$. Equivalently,

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

For a linear space $W \subseteq \mathbb{R}^n$, $g \in W$ is an *elementary vector* if g is a support minimal nonzero vector in W, that is, no $h \in W \setminus \{\mathbf{0}\}$ exists such that $\operatorname{supp}(h) \subsetneq \operatorname{supp}(g)$, where supp denotes the support of a vector. A *circuit* in W is the support of some elementary vector; these are precisely the circuits in the associated linear matroid $\mathcal{M}(W)$. We let $\mathcal{E}(W) \subseteq W$ and $\mathcal{C}_W \subseteq 2^n$ denote the set of elementary vectors and circuits in the space W, respectively.

A *circuit basis* of a subspace $W \subseteq \mathbb{R}^n$ is a set $\mathcal{F} \subseteq \mathcal{E}(W)$ of $\operatorname{rk}(W)$ linearly independent elementary vectors, i.e., $\operatorname{span}(\mathcal{F}) = W$.

Linear matroids. For a linear subspace $W \subseteq \mathbb{R}^n$, let $\mathcal{M}(W) = ([n], \mathcal{I})$ denote the associated linear matroid, i.e., the matroid defined by the set of circuits $\mathcal{C}_W$. Here, $\mathcal{I}$ denotes the set of independent sets; $S \in \mathcal{I}$ if and only if there exists no $z \in W \setminus \{0\}$ with $\operatorname{supp}(z) \subseteq S$; the maximal independent sets are the bases. We refer the reader to [Sch03, Chapter 39] or [Fra11, Chapter 5] for relevant definitions and background on matroid theory.

Assume $\operatorname{rk}(\mathbf{A}) = m$ and $W = \ker(\mathbf{A})$ for $\mathbf{A} \in \mathbb{R}^{m \times n}$. Then $B \subseteq [n]$, $|B| = m$ is a basis in $\mathcal{M}(\mathbf{A}) \coloneqq \mathcal{M}(W)$ if and only if $\mathbf{A}_B$ is nonsingular; then, $\mathbf{A}' = \mathbf{A}_B^{-1}\mathbf{A}$ is in basis form for B such that $\ker(\mathbf{A}') = W$.

The matroid $\mathcal{M}$ is separable if the ground set $[n]$ can be partitioned into two nonempty subsets $[n] = S \cup T$ such that $I \in \mathcal{I}$ if and only if $I \cap S, I \cap T \in \mathcal{I}$. In this case, the matroid is the direct sum of its restrictions to S and T. In particular, every circuit is fully contained in S or in T. For the linear matroid $\mathcal{M}(\mathbf{A})$, separability means that $\ker(\mathbf{A}) = \ker(\mathbf{A}_S) \oplus \ker(\mathbf{A}_T)$. In this case, we have $\kappa_{\mathbf{A}} = \max\{\kappa_{\mathbf{A}_S}, \kappa_{\mathbf{A}_T}\}$ and $\dot{\kappa}_{\mathbf{A}} = \operatorname{lcm}\{\dot{\kappa}_{\mathbf{A}_S}, \dot{\kappa}_{\mathbf{A}_T}\}$; solving 1.1 can be decomposed into two subproblems, restricted to the columns in $\mathbf{A}_S$ and in $\mathbf{A}_T$.

Thus, for most concepts and problems considered in this thesis, we can focus on the *non-separable* components of $\mathcal{M}(W)$. The following characterization will turn out to be very useful, see e.g. [Fra11, Theorem 5.2.5].

Proposition 2.0.2 (See Proposition 3.2.21)**.** *A matroid $\mathcal{M} = ([n], \mathcal{I})$ is non-separable if and only if for any $i, j \in [n]$, there exists a circuit containing i and j.*

Conformal circuit decompositions. We say that the vector $y \in \mathbb{R}^n$ *conforms* to $x \in \mathbb{R}^n$ if $x_i y_i > 0$ whenever $y_i \neq 0$. Given a subspace $W \subseteq \mathbb{R}^n$, a conformal circuit decomposition of a vector $z \in W$ is a decomposition

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

where $h \leq n$ and $g^1, g^2, \ldots, g^h \in \mathcal{E}(W)$ are elementary vectors that are conformal with z. A fundamental result on elementary vectors asserts the existence of a conformal circuit decomposition, see e.g. [Ful68; Roc69].

We write $y \sqsubseteq x$ if y conforms to x and $|y_i| \leq |x_i|$ for all $i \in [n]$.

Lemma 2.0.3. *For every subspace $W \subseteq \mathbb{R}^n$, every $z \in W$ admits a conformal circuit decomposition.*

Proof. Let $F \subseteq W$ be the set of vectors conformal with z. The set 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 $\{\, \alpha g : \alpha \geq 0 \,\}$ for $g \in \mathcal{E}(W)$. As $z \in F$, it can be written as a conic combination of at most n rays by the Minkowski–Weyl theorem. Such a decomposition yields a conformal circuit decomposition. □

Computational Model. We use the real model of computation, allowing basic arithmetic operations $+, -, \times, /$, comparisons, and square root computations. Exact square root computations could be omitted by using approximate square roots; we assume exact computations for simplicity.

Throughout, we use $\widetilde{O}(\cdot)$ to hide polylogarithmic factors in m and n, and we use $\operatorname{poly}(\cdot)$ to denote polynomial running time in the arguments.

LP formulation We write the feasible regions of $\operatorname{Primal}(W, d)$ and $\operatorname{Dual}(W, c)$ as

$$\mathcal{P} \coloneqq \{\, x \in \mathbb{R}^n : x \in W + d, x \geq \mathbf{0} \,\}, \quad \mathcal{D} \coloneqq \left\{\, s \in \mathbb{R}^n : s \in W^\perp + c, s \geq \mathbf{0} \,\right\}. \tag{2.3}$$

3 Circuit imbalance measures and linear programming

We study properties and applications of various circuit imbalance measures associated with linear spaces. These measures describe possible ratios between nonzero entries of support-minimal nonzero vectors of the space. The fractional circuit imbalance measure turns out to be a crucial parameter in the context of linear programming, and two integer variants can be used to describe integrality properties of associated polyhedra.

We give an overview of the properties of these measures, and survey classical and recent applications, in particular, for linear programming algorithms with running time dependence on the constraint matrix only, and for circuit augmentation algorithms. We also present new bounds on the diameter and circuit diameter of polyhedra in terms of the fractional circuit imbalance measure.

This chapter is based on joint work with Farbod Ekbatani and László A. Végh [ENV22].

Contents

3.1 Introduction

Recall that an elementary vector in a subspace is a support-minimal nonzero vector. Elementary vectors were first studied in the 1960s by Camion [Cam64], Tutte [Tut65], Fulkerson [Ful68], and Rockafellar [Roc69]. Circuits play a crucial role in matroid theory and have been extremely well studied. For regular subspaces (i.e., kernels of totally unimodular matrices), elementary vectors have ±1 entries; this fact has been at the heart of several arguments in network optimization since the 1950s.

The focus of this chapter is on various circuit imbalance measures. We give an overview of classical and recent applications, and their relationship with other condition measures. We will mainly focus on applications in linear programming, mentioning in passing also their relevance to integer programming.

Three circuit imbalance measures. There are multiple ways to quantify how 'imbalanced' elementary vectors of a subspace can be. We define three different measures that capture various fractionality and integrality properties.

For every $C \in \mathcal{C}_W$, the elementary vectors with support C form a one-dimensional subspace of W. We pick a representative $g^{C,W} \in \mathcal{E}(W)$ from this subspace. If W is not a rational subspace, we select $g^{C,W}$ arbitrarily. For rational subspaces, we select $g^{C,W}$ as an integer vector with the largest common divisor of the coordinates being 1; this choice is unique up to multiplication by -1. When clear from the context, we omit the index W and simply write g^C. We now define the fractional circuit imbalance measure and two variants of integer circuit imbalance measure.

Definition 3.1.1 (Circuit imbalances)**.** *For a non-trivial linear subspace $W \subseteq \mathbb{R}^n$, let us define the following notions:*

- *The fractional circuit imbalance measure of W is*

$$\kappa_W := \max\left\{ \left| \frac{g^C_j}{g^C_i} \right| : C \in \mathcal{C}_W, i, j \in C \right\}.$$

- *If W is a rational linear space, the lcm-circuit imbalance measure is*

$$\dot{\kappa}_W := \operatorname{lcm}\left\{ \operatorname{lcm}(g^C) : C \in \mathcal{C}_W \right\}.$$

- *If W is a rational linear space, the* max-circuit imbalance measure *is*

$$\bar{\kappa}_W := \max\left\{ \|g^C\|_\infty : C \in \mathcal{C}_W \right\}.$$

For trivial subspaces W, we define $\kappa_W = \dot{\kappa}_W = \bar{\kappa}_W = 1$. Further, we say that the rational subspace W is anchored *if every vector g^C, $C \in \mathcal{C}_W$ has a ± 1 entry.*

Equivalently, in an anchored subspace every elementary vector $g \in \mathcal{E}(W)$ has a nonzero entry such that all other entries are integer multiples of this entry.

The term circuit imbalance measure will refer to the fractional measure κ_W. Note that $1 \le \kappa_W \le \bar{\kappa}_W \le \dot{\kappa}_W$ and $\kappa_W = 1$ implies $\bar{\kappa}_W = \dot{\kappa}_W = 1$. This case plays a distinguished role and turns out to be equivalent to W being a regular linear space (see Theorem 3.2.4).

Another important case is when $\dot{\kappa}_W = p^\alpha$ is a prime power. In this case, W is anchored, and $\kappa_W = \bar{\kappa}_W = \dot{\kappa}_W$. The linear space will often be represented as $W = \ker(\mathbf{A})$ for a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$. We will use $\mathcal{E}(\mathbf{A})$, $\mathcal{C}_\mathbf{A}$, $\kappa_\mathbf{A}$, $\dot{\kappa}_\mathbf{A}$, $\bar{\kappa}_\mathbf{A}$ to refer to the corresponding quantities in $\ker(\mathbf{A})$.

An earlier systematic study of elementary vectors was done in Lee's work [Lee89]. He mainly focused on the max-circuit imbalance measure; we give a quick comparison to the results in Section 3.2. The fractional circuit imbalance measure played a key role in the paper [DHNV20] on layered-least-squares interior point methods; it turns out to be a close proxy to the well-studied condition number $\bar{\chi}_W$. As far as the authors are aware, the lcm-circuit imbalance measure has not been explicitly studied previously.

Overview and contributions. Section 3.2 gives an overview of fundamental properties of κ_W and $\dot{\kappa}_W$. In particular, Section 3.2.1 relates circuit imbalances to subdeterminant bounds. We note that many extensions of totally unimodular matrices focus on matrices with bounded subdeterminants. Working with circuit imbalances directly can often lead to stronger and conceptually cleaner results. Section 3.2.2 presents an extension of the Hoffman-Kruskal characterization of TU matrices. Section 3.2.3 shows an important self-duality property of κ_W and $\dot{\kappa}_W$. Section 3.2.4 studies 'nice' matrix representations of subspaces with given lcm-circuit imbalances. Section 3.2.6 proves a multiplicative triangle-inequality for κ_W. Many of these results were previously shown by Lee [Lee89], Appa and Kotnyek [AK04], and by Dadush et al. [DHNV20]. We present them in a unified framework, extend some of the results, and provide new proofs.

Section 3.3 reveals connections between κ_W and the well-studied condition numbers $\bar{\chi}$ studied in the context of interior point methods, and δ studied—among other topics—in the analysis of the shadow simplex method. In particular, we show that previous diameter bounds for polyhedra can be translated to strong diameter bounds in terms of the condition number κ_W (Theorem 3.3.12).

Section 3.4 studies the best possible values of κ_W that can be achieved by rescaling the variables. We present the algorithm and min-max characterization from [DHNV20]. Further, we characterize when a subspace can be rescaled to a regular one; we also give a new proof of a theorem from [Lee89].

Section 3.6 gives an outlook to integer programming, showing the relationship between the max-circuit imbalance and Graver bases. Finally, Section 3.7 formulates a conjecture on circuit decompositions with bounded fractionality.

3.2 Properties of the imbalance measures

Comparison to well-scaled frames. Lee's work [Lee89] on *'well-scaled frames'* investigated the following closely related concepts. For a set $S \subseteq \mathbb{Q}$ the rational linear space W is *S-regular* if for every elementary vector $g \in \mathcal{E}(F)$, there exists a $\lambda \neq 0$ such that all nonzero entries of λg are in S. For $S = \{-k, \dots, k\}$, the subspace is called *k-regular*. For $k, \Omega \in \mathbb{N}$, a subspace is *k-adic of order Ω* if it is S-regular for $S = \{\pm 1, \pm k, \dots, \pm k^{\Omega}\}$. The *frame* of the subspace W refers to the set of elementary vectors $\mathcal{E}(W)$.

Using our terminology, a subspace is k-regular if and only if $\bar{\kappa}_W = k$, and every k-adic subspace is anchored. Many of the properties in this section were explicitly or implicitly shown in Lee [Lee89]. However, it turns out that many properties are simpler and more natural to state in terms of either κ_W and $\dot{\kappa}_W$. Roughly speaking, the fractional circuit imbalance κ_W is the key quantity of interest for continuous properties, particularly relevant for proximity results in linear programming. On the other hand, the lcm-circuit imbalance $\dot{\kappa}_W$ captures most clearly the integrality properties. The max-circuit imbalance $\bar{\kappa}_W$ interpolates between these two, although, as already noted by Lee, it is the right quantity for proximity results in integer programming (see Section 3.6).

Appa and Kotnyek [AK04] also use the term k-regularity in a different sense, as a natural extension of unimodularity. This turns out to be strongly related to $\dot{\kappa}_W$; see Lemma 3.2.3 and Corollary 3.2.9.

The key lemma on basis forms. The following simple proposition turns out to be extremely useful in deriving properties of κ_W and $\dot{\kappa}_W$. The first statement is from [DNV20].

Proposition 3.2.1. *For every matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(\mathbf{A}) = m$,*

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

Moreover, for each nonsingular $\mathbf{A}_B$, all nonzero entries of $\mathbf{A}_B^{-1}\mathbf{A}$ have absolute values between $1/\kappa_{\mathbf{A}}$ and $\kappa_{\mathbf{A}}$ and are $1/\dot{\kappa}_{\mathbf{A}}$-integral.

Proof. Consider the matrix $\mathbf{A}' = \mathbf{A}_B^{-1}\mathbf{A}$ for any non-singular $m \times m$ submatrix $\mathbf{A}_B$. Let us renumber the columns such that B corresponds to the first m columns. Then, for every $m+1 \le j \le n$, the jth column of $\mathbf{A}'$ corresponds to an elementary vector g where $g_j = 1$, and $g_i = -\mathbf{A}'_{ij}$ for $i \in [m]$. Hence, $\|\mathbf{A}'\|_{\max}$ gives a lower bound on $\kappa_{\mathbf{A}}$. This also implies that all nonzero entries are between $1/\kappa_{\mathbf{A}}$ and $\kappa_{\mathbf{A}}$. To see that all entries of A' are $1/\dot{\kappa}_A$-integral, note that $g = g'/\alpha$ for a vector g' where all entries are integer divisors of $\dot{\kappa}_{\mathbf{A}}$. Since $g_j = 1$, it follows that α itself is an integer divisor of $\dot{\kappa}_{\mathbf{A}}$.

To see that the maximum in the first statement is achieved, take the elementary vector g^C that attains the maximum in the definition of $\kappa_{\mathbf{A}}$; let g^C_j be the minimum absolute value element. Let us select a basis B such that $C \setminus \{j\} \subseteq B$. Then, the largest absolute value in the j-th column of $\mathbf{A}_B^{-1}\mathbf{A}$ will be $\kappa_{\mathbf{A}}$. □

3.2.1 Bounds on subdeterminants

For an integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$, we define

$$\begin{aligned} \Delta_{\mathbf{A}} &:= \max\left\{ |\det(\mathbf{B})| : \mathbf{B} \text{ is a nonsingular submatrix of } \mathbf{A} \right\}, \text{ and} \\ \dot{\Delta}_{\mathbf{A}} &:= \mathrm{lcm}\left\{ |\det(\mathbf{B})| : \mathbf{B} \text{ is a nonsingular submatrix of } \mathbf{A} \right\}. \end{aligned} \tag{3.1}$$

The matrix is TU if $\Delta_{\mathbf{A}} = 1$: thus, all subdeterminants are 0 or ±1. This class of matrices plays a foundational role in combinatorial optimization, see e.g., [Sch98, Chapters 19-20]. A significant example is the node-arc incidence matrix of a directed graph. A key property is that they define integer polyhedra, see Theorem 3.2.5 below. A polynomial-time algorithm is known to decide whether a matrix is TU, based on the deep decomposition theorem by Seymour from 1980 [Sey80].

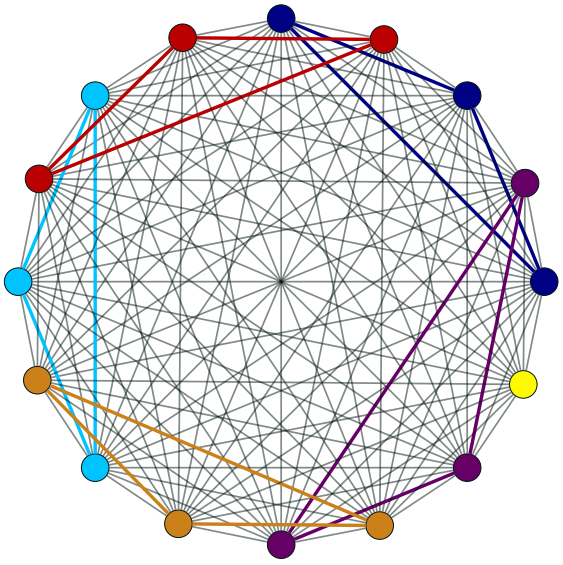
The next statement is implicit in [Lee89, Proposition 5.3].

Proposition 3.2.2. *For every integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$, $\bar{\kappa}_{\mathbf{A}} \le \Delta_{\mathbf{A}}$ and $\dot{\kappa}_{\mathbf{A}} \le \dot{\Delta}_{\mathbf{A}}$.*

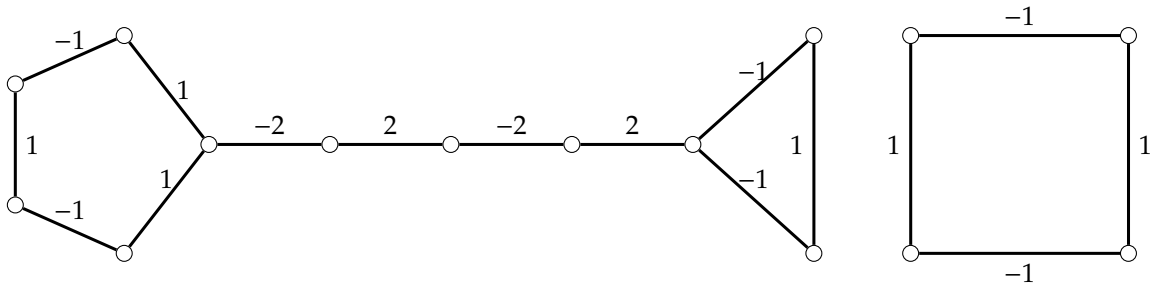
Proof. Let $C \in \mathcal{C}_{\mathbf{A}}$ be a circuit, and select a submatrix $\hat{\mathbf{A}} \in \mathbb{Z}^{(|C|-1)\times|C|}$ of $\mathbf{A}$ where the columns are indexed by C, and the rows are linearly independent. Let $\hat{\mathbf{A}}_{-i}$ be the square submatrix resulting from deleting the column corresponding to i from $\hat{\mathbf{A}}$. From Cramer's rule, we see that $|g^C_i| = |\det(\hat{\mathbf{A}}_{-i})|/\alpha$ for some $\alpha \in \mathbb{Q}$, $\alpha \ge 1$. This implies both claims $\bar{\kappa}_{\mathbf{A}} \le \Delta_{\mathbf{A}}$ and $\dot{\kappa}_{\mathbf{A}} \le \dot{\Delta}_{\mathbf{A}}$. □

In Propositions 3.2.18 and 3.2.19, we show that for any matrix $\mathbf{A} \in \mathbb{Q}^{m\times n}$ there exists a matrix $\tilde{\mathbf{A}} \in \mathbb{Z}^{m\times n}$ such that $\ker(\mathbf{A}) = \ker(\tilde{\mathbf{A}})$ and $\dot{\Delta}_{\tilde{\mathbf{A}}} \le (\dot{\kappa}_{\mathbf{A}})^m$.

To see an example where $\Delta_{\mathbf{A}}$ can be much larger than $\kappa_{\mathbf{A}}$, let $\mathbf{A} \in \mathbb{Z}^{n\times\binom{n}{2}}$ be the node-edge incidence matrix of a complete undirected graph on n nodes (see Figure 3.1); assume n is divisible by 3. The determinant corresponding to any submatrix corresponding to an odd cycle is ±2. Let H be an edge set of $\frac{n}{3}$ node-disjoint triangles. Then A_H is a square submatrix with determinant $\pm 2^{n/3}$. In fact, $\Delta_{\mathbf{A}} = 2^{n/3}$ in this case, since $\Delta_{\mathbf{A}}$ for a node-edge incidence matrix equals the maximum number of node disjoint odd cycles, see [GKS95]. On the other hand, $\kappa_{\mathbf{A}} = \bar{\kappa}_{\mathbf{A}} = \dot{\kappa}_{\mathbf{A}} \in \{1, 2\}$ for the



(a) Vertex disjoint triangles



(b) the two types of circuits for incidence matrix in undirected graphs

Figure 3.1: Showcased difference in κ and Δ.

incidence matrix $\mathbf{A}$ of any undirected graph; see Section 3.2.2. The difference between the two is displayed in Figure 3.1.

For TU-matrices, the converse of Proposition 3.2.2 is also true. In 1956, Heller and Tompkins [Hel57; HT56] introduced the *Dantzig property*. A matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ has the Dantzig property if $\mathbf{A}_B^{-1}\mathbf{A}$ is a $0, \pm 1$-matrix for every nonsingular $m \times m$ submatrix $\mathbf{A}_B$. According to Proposition 3.2.1, this is equivalent to $\kappa_{\mathbf{A}} = 1$. Theorem 3.2.4 below can be attributed to Cederbaum [Ced57, Proposition (v)]; see also Camion's PhD thesis [Cam64, Theorem 2.4.5(f)]. The key is the following lemma that we formulate for general $1/\dot{\kappa}_{\mathbf{A}}$ for later use.

Lemma 3.2.3. *Let* $\mathbf{A} = \begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix} \in \mathbb{R}^{m\times n}$. *Then, for any nonsingular square submatrix* $\mathbf{M}$ *of* $\mathbf{A}$, *the inverse* $\mathbf{M}^{-1}$ *is* $1/\dot{\kappa}_{\mathbf{A}}$*-integral, with non-zero entries between* $1/\kappa_{\mathbf{A}}$ *and* $\kappa_{\mathbf{A}}$ *in absolute value.*

Proof. Let $\mathbf{M}$ be any $k \times k$ nonsingular submatrix of $\mathbf{A}$; w.l.o.g., let us assume that it uses the first k rows of $\mathbf{A}$. Let B be the set of columns of $\mathbf{M}$, along with the $m - k$ additional columns $i \in [k+1, m]$, i.e., the last $m - k$ unit vectors from $\mathbf{I}_m$. Thus, $\mathbf{A}_B \in \mathbb{R}^{m\times m}$ is also nonsingular. After permuting the columns, this can be written in the form

$$\mathbf{A}_B = \begin{bmatrix}\mathbf{M} & \mathbf{0} \\ \mathbf{L} & \mathbf{I}_{m-k}\end{bmatrix} \tag{3.2}$$

for some $\mathbf{L} \in \mathbb{Z}^{(m-k)\times k}$. We now use Proposition 3.2.1 for $\tilde{\mathbf{A}} = \mathbf{A}_B^{-1}\mathbf{A}$. Note that the first m columns of $\tilde{\mathbf{A}}$ correspond to $\mathbf{A}_B^{-1}$. Moreover, we see that

$$\mathbf{A}_B^{-1} = \begin{bmatrix} \mathbf{M}^{-1} & \mathbf{0} \\ -\mathbf{L}\mathbf{M}^{-1} & \mathbf{I}_{m-k} \end{bmatrix} \tag{3.3}$$

Thus, $\mathbf{M}^{-1}$ is $1/\dot{\kappa}_{\mathbf{A}}$-integral, with non-zero entries between $1/\kappa_{\mathbf{A}}$ and $\kappa_{\mathbf{A}}$ completing the proof. □

Appa and Kotnyek define k-regular matrices as follows: a rational matrix $\mathbf{A}' \in \mathbb{R}^{m\times n}$ is k-regular if and only if the inverse of all nonsingular submatrices is $1/k$-integral. From the above statement, it follows that $\mathbf{A}'$ is k-regular in this sense for $k = \kappa_{[\mathbf{I}_m\ \mathbf{A}']}$. See also Corollary 3.2.9.

Theorem 3.2.4 (Cederbaum, 1957)**.** *Let $W \subset \mathbb{R}^n$ be a linear subspace. Then, the following are equivalent.*

(i) $\kappa_W = \bar{\kappa}_W = \dot{\kappa}_W = 1$.

(ii) There exists a TU matrix $\mathbf{A}$*, such that* $W = \ker(\mathbf{A})$.

(iii) For any matrix $\mathbf{A}$ *in basis form such that* $W = \ker(\mathbf{A})$*,* $\mathbf{A}$ *is a TU-matrix.*

Proof. (iii) ⇒ (ii) is straightforward, and (ii) ⇒ (i) follows by Proposition 3.2.2. It remains to show (i) ⇒ (iii). Let $\mathrm{rk}(W) = n - m$, and consider any $\mathbf{A} \in \mathbb{R}^{m\times n}$ in basis form such that $W = \ker(\mathbf{A})$. For simplicity of notation, assume the basis is formed by the first m columns, that is, $\mathbf{A} = [\mathbf{I}_m\ \mathbf{A}']$ for some $\mathbf{A}' \in \mathbb{R}^{m\times(n-m)}$.

Proposition 3.2.1 implies that all entries of $\mathbf{A}$ are 0 and ±1. Consider any nonsingular square submatrix $\mathbf{M}$ of $\mathbf{A}$. By Lemma 3.2.3, $\mathbf{M}^{-1}$ is also a 0, ±1 matrix. Consequently, both $\det(\mathbf{M})$ and $\det(\mathbf{M}^{-1})$ are nonzero integers, which implies that $|\det(\mathbf{M})| = 1$, as required. □

3.2.2 Fractional integrality characterization

Hoffman and Kruskal [HK56] gave the following characterization of TU matrices. A polyhedron $P \subseteq \mathbb{R}^n$ is *integral* if all vertices (i..e., basic feasible solutions) are integer.

Theorem 3.2.5 (Hoffman and Kruskal, 1956)**.** *An integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ is TU if and only if for every $b \in \mathbb{Z}^m$, the polyhedron*

$$\{\, x \in \mathbb{R}^n : \mathbf{A}x \le b, x \ge \mathbf{0} \,\} \tag{3.4}$$

is integral.

Since $\dot{\kappa}$ is a property of the subspace, it will be more convenient to work with the standard equality form of an LP. Here as well as in Section 3.3.2, we use the following straightforward correspondence between the two forms. Recall that an *edge* of a polyhedron is a bounded one dimensional face; every edge is incident to exactly two vertices. The following statement is standard and easy to verify.

Lemma 3.2.6. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ be of the form $\mathbf{A} = \begin{bmatrix} \mathbf{A}' & \mathbf{I}_m \end{bmatrix}$ for $\mathbf{A}' = \mathbb{R}^{m\times(n-m)}$. For a vector $b \in \mathbb{R}^m$, let*

$$P_b := \{\, x \in \mathbb{R}^n : \mathbf{A}x = b, x \ge \mathbf{0} \,\} \quad \text{and} \quad P'_b := \{\, x' \in \mathbb{R}^{n-m} : \mathbf{A}'x' \le b, x' \ge \mathbf{0} \,\}. \tag{3.5}$$

Let $I = [n-m]$ denote the index set of $\mathbf{A}'$. Then, $P'_b = \pi_I(P_b)$, i.e., P'_b is the projection of P_b to the coordinates in I. For every vertex x of P_b, $x' = x_I$ is a vertex of P'_b, and conversely, for every vertex x' of P'_b, there exists a unique vertex x of P such that $x_I = x'$. There is a one-to-one correspondence between the edges of P_b and P'_b. Further, if $b \in \mathbb{Z}^m$, then P_b is $1/k$-integral if and only if P'_b is $1/k$-integral.

Using Theorem 3.2.4 and Lemma 3.2.6, we can formulate Theorem 3.2.5 in subspace language.

Corollary 3.2.7. *Let $W \subseteq \mathbb{R}^n$ be a linear space. Then, $\kappa_W = 1$ if and only if for every $d \in \mathbb{Z}^n$, the polyhedron*

$$\{ x \in \mathbb{R}^n : x \in W + d, x \geq \mathbf{0} \} \tag{3.6}$$

is integral.

Proof. Let $n' = n - m = \dim(W)$. W.l.o.g., assume the last m variables form a basis, and let us represent W in a basis form as $W = \ker(\mathbf{A})$ for $A = \begin{bmatrix} \mathbf{A} & \mathbf{I}_m \end{bmatrix}$, where $\mathbf{A}' \in \mathbb{R}^{m \times n'}$. It follows by Theorem 3.2.4 that $\kappa_W = 1$ if and only if A is TU, which is further equivalent to $\mathbf{A}'$ being TU.

Further, note that the system $\{ x \in \mathbb{R}^n : x \in W + d, x \geq 0 \}$ coincides with

$$P_b = \left\{ x \in \mathbb{R}^n : \begin{bmatrix} \mathbf{A}' & \mathbf{I}_m \end{bmatrix} x = b, x \geq \mathbf{0} \right\}, \tag{3.7}$$

where $b = \mathbf{A}d$.

Note that $b = \mathbf{A}d$ is integer whenever $d \in \mathbb{Z}^m$. Moreover, we can obtain every integer vector in $b \in \mathbb{Z}^m$ this way, since $\mathbf{A}$ contains an identity matrix. According to Lemma 3.2.6, P_b is integral if and only if $P'_b = \{x \in \mathbb{R}^{n-m} : \mathbf{A}'x' \leq b, x' \geq \mathbf{0}\}$ is integral. The claim follows by Theorem 3.2.5. □

We provide the following natural generalization. Related statements, although in substantially more complicated forms, were given in [Lee89, Proposition 6.1 and 6.2].

Theorem 3.2.8. *Let $W \subseteq \mathbb{R}^n$ be a linear space. Then, $\dot{\kappa}_W$ is the smallest integer $k \in \mathbb{Z}$ such that for every $d \in \mathbb{Z}^n$, the polyhedron $\{ x \in \mathbb{R}^n : x \in W + d, x \geq \mathbf{0} \}$ is $1/k$-integral.*

Proof. Let $\dim(W) = n - m$, and let us represent $W = \ker(\mathbf{A})$ for $\mathbf{A} \in \mathbb{R}^{m \times n}$. Then, $x \in W + d$, $x \geq \mathbf{0}$ can be written as $\mathbf{A}x = \mathbf{A}d$, $x \geq \mathbf{0}$. Let x be a basic feasible solution (i.e., vertex) of this system. Then, $x = \mathbf{A}_B^{-1}\mathbf{A}d$. By Proposition 3.2.1, $\mathbf{A}_B^{-1}\mathbf{A}$ is $1/\dot{\kappa}_W$-integral. Thus, if $d \in \mathbb{Z}^n$ then x must be also $1/\dot{\kappa}_W$-integral.

Let us now show the converse direction. Assume $\{ x \in \mathbb{R}^n : x \in W + d, x \geq 0 \}$ is $1/k$-integral for every $d \in \mathbb{Z}^n$. For a contradiction, assume there exists a circuit $C \in \mathcal{C}_W$ such that the entries of the elementary vector are not all divisors of k (or that g^C is not even a rational vector if W is not a rational space). In particular, select an index $\ell \in C$ such that $g^C_\ell \nmid k$, or such that $(1/g^C_\ell)g^C$is not rational.

Let us select a basis $B \subseteq [n]$ such that $C \setminus B = \{\ell\}$. For simplicity of notation, let $B = [m]$. We can represent $W = \ker(\mathbf{A})$ in a basis form as $\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & \mathbf{A}' \end{bmatrix}$. Let $g \in \mathbb{R}^n$ be defined by $g_\ell = 1$, $g_j = -\mathbf{A}_{j\ell}$ for $j \in B$ and $g_j = 0$ otherwise; thus, $g = (1/g^C_\ell)g^C$.

Let us pick an integer $t \in \mathbb{N}$, $t \geq \|g\|_\infty$, and define $d \in \mathbb{Z}^n$ by $d_j = t$ for $j \in B$, $d_\ell = -1$, and $d_j = 0$ otherwise. Then, the basic solution of $x \in W + d$, $x \geq 0$ corresponding to the basis B is obtained as $x_j = t + g_j$ for $j \in B$ and $x_j = 0$ for $j \in [n] \setminus B$. The choice of t guarantees $x \geq 0$. By the assumption, x is $1/k$-integer, and therefore g is also $1/k$-integer. Recall that $g = (1/g^C_\ell)g^C$, where either $g^C \in \mathbb{Z}^n$ with $\operatorname{lcm}(g^C) = 1$ and $g^C_\ell \nmid k$, or g is not rational. Both cases give a contradiction. □

Using again Lemma 3.2.6, we can write this theorem in a form similar to the Hoffman-Kruskal theorem.

Corollary 3.2.9. *Let $\mathbf{A} = \begin{bmatrix} \mathbf{A}' & \mathbf{I}_m \end{bmatrix} \in \mathbb{R}^{m \times n}$. Then, $\dot{\kappa}_\mathbf{A}$ is the smallest value k such that for every $b \in \mathbb{Z}^m$, the polyhedron*

$$\{ x' \in \mathbb{R}^{n-m} : \mathbf{A}'x' \leq b, x' \geq \mathbf{0} \} \tag{3.8}$$

is $1/k$-integral.

Appa and Kotnyek [AK04, Theorem 17] show that k-regularity of $\mathbf{A}'$ (in the sense that the inverse of every square submatrix is $1/k$-integral) is equivalent to the property above.

Subspaces with $\dot{\kappa}_{\mathbf{A}} = 2$. The case $\dot{\kappa}_W = 2$ is a particularly interesting class. As already noted, it includes incidence matrices of undirected graphs, and according to Theorem 3.2.8, it corresponds to half-integer polytopes. This class includes the following matrices, first studied by Edmonds and Johnson [EJ70]; the following result follows e.g., from [AK04; GS86; HMNT93].

Theorem 3.2.10. *Let $\mathbf{A} \in \mathbb{Z}^{m\times n}$ such that for each column $j \in [n]$, $\sum_{i=1}^{m} |\mathbf{A}_{ij}| \leq 2$. Then $\dot{\kappa}_{\mathbf{A}} \in \{1, 2\}$.*

Appa and Kotnyek [AK04] define *binet matrices* as $\mathbf{A}' = \mathbf{A}_B^{-1}\mathbf{A}$ for a matrix $\mathbf{A}$ as in Theorem 3.2.10 for a basis B. These matrices have $\dot{\kappa}_{\mathbf{A}'} \in \{1, 2\}$ since they define the same subspace.

Deciding whether a matrix has $\dot{\kappa}_{\mathbf{A}} = 2$ (or more generally, $\dot{\kappa}_{\mathbf{A}} = k$ for a fixed constant k) is an interesting open question: is it possible to extend Seymour's decomposition [Sey80] from TU matrices? The matrices in Theorem 3.2.10 could be a natural building block of such a decomposition.

3.2.3 Self-duality

We next show that both κ_W and $\dot{\kappa}_W$ are self-dual. These rely on the following duality property of circuits. We introduce the following more refined quantities that will also come useful later on.

Definition 3.2.11 (Pairwise Circuit Imbalances)**.** *For a space $W \subseteq \mathbb{R}^n$ and variables $i, j \in [n]$ we define*

$$\begin{aligned} \mathcal{K}_{ij}^W &:= \left\{ \left|\frac{g_j}{g_i}\right| : \{i,j\} \subseteq \operatorname{supp}(g), g \in \mathcal{E}(W) \right\}, \quad \kappa_{ij}^W := \max \mathcal{K}_{ij}^W, \\ \dot{\mathcal{K}}_{ij}^W &:= \left\{ \operatorname{lcm}(p,q) : p, q \in \mathbb{N}, \gcd(p,q) = 1, \frac{p}{q} \in \mathcal{K}_{ij}^W \right\}. \end{aligned} \tag{3.9}$$

We call κ_{ij}^W the pairwise imbalance between i and j.

Cleary, $\kappa_W = \max_{i,j\in[n]} \kappa_{ij}^W$ for a nontrivial linear space W. We use the following simple lemma.

Lemma 3.2.12. *Consider a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ in basis form for $B \subseteq [n]$, i.e., $\mathbf{A}_B = \mathbf{I}_m$. Let $W = \ker(\mathbf{A})$; thus, $W^\perp = \operatorname{span}(\mathbf{A}^\top)$. The following hold.*

(i) The rows of $\mathbf{A}$ form a circuit basis of $W^\perp$, denoted as $\mathcal{E}_B(W^\perp)$.

(ii) For any two rows $\mathbf{A}^i, \mathbf{A}^j$, $i, j \in B$, $i \neq j$, and $k \in [n] \setminus B$, the vector $h = \mathbf{A}_{jk}\mathbf{A}^i - \mathbf{A}_{ik}\mathbf{A}^j$ fulfills $h \in \mathcal{E}(W^\perp)$.

Proof. For part (i), the rows are linearly independent and span $W^\perp$. Therefore, every $g \in W^\perp$ must have $\operatorname{supp}(g) \cap B \neq \emptyset$, and if $\operatorname{supp}(g) \cap B = \{i\}$ then $g = g_i\mathbf{A}^i$. These two facts imply that each $\mathbf{A}^i$ is support minimal in $W^\perp$, that is, $\mathbf{A}^i \in \mathcal{E}(W^\perp)$.

For part (ii), there is nothing to prove if $\mathbf{A}_{ik} = 0$ or $\mathbf{A}_{jk} = 0$; for the rest, assume both are nonzero. Assume for a contradiction $h \notin \mathcal{E}(W^\perp)$; thus, there exists a $g \in W^\perp$, $g \neq 0$ and $\operatorname{supp}(g) \subsetneq \operatorname{supp}(h)$. We have $\operatorname{supp}(h) \cap B = \{i, j\}$. If $\operatorname{supp}(g) \cap B \subsetneq \{i, j\}$, as above we get that $g = g_i\mathbf{A}^i$ or $g = g_j\mathbf{A}^j$, a contradiction since $h_k = 0$ but $\mathbf{A}_{ik}, \mathbf{A}_{jk} \neq 0$. Hence, $\operatorname{supp}(g) \cap B = \{i, j\}$. By part (i), we have $g = g_i\mathbf{A}^i + g_j\mathbf{A}^j$; and since $h_k = 0$ it follows that $g_i/g_j = -\mathbf{A}_{jk}/\mathbf{A}_{ik}$; thus, g is a scalar multiple of h, a contradiction. □

Lemma 3.2.13. *For any $i, j \in [n]$ we have $\mathcal{K}_{ij}^W = \left\{ \alpha^{-1} : \alpha \in \mathcal{K}_{ji}^{W^\perp} \right\}$. Equivalently: for every elementary vector $g \in \mathcal{E}(W)$ with indices $i, j \in \operatorname{supp}(g)$ there exists an elementary vector $h \in \mathcal{E}(W^\perp)$ such that $|h_i/h_j| = |g_j/g_i|$.*

Proof. Let $g \in \mathcal{E}(W)$ such that $i, j \in \operatorname{supp}(g)$. If $\operatorname{supp}(g) = \{i, j\}$ then any $h \in \mathcal{E}(W^\perp)$ with $i \in \operatorname{supp}(h)$ fulfills $g_i h_i + g_j h_j = \langle g, h \rangle = 0$, so $j \in \operatorname{supp}(h)$ and $|h_i/h_j| = |g_j/g_i|$.

Else, there exists $k \in \operatorname{supp}(g) \setminus \{i, j\}$. Let us select a basis B of $\mathcal{M}(W)$ with $\operatorname{supp}(g) \setminus B = \{k\}$. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a matrix in basis form for B with $\ker(\mathbf{A}) = W$, and let $h = \mathbf{A}_{jk}\mathbf{A}^i - \mathbf{A}_{ik}\mathbf{A}^j$, an elementary vector in $\mathcal{E}(W^\perp)$ by Lemma 3.2.12(ii).

By the construction, $|h_i/h_j| = |\mathbf{A}_{jk}/\mathbf{A}_{ik}|$. On the other hand, $\langle g, \mathbf{A}^i \rangle = 0$ and $\operatorname{supp}(g) \setminus B = \{k\}$ implies $g_i = -g_k \mathbf{A}_{ik}$ and similarly $\langle g, \mathbf{A}^j \rangle = 0$ implies $g_j = -g_k \mathbf{A}_{jk}$. The claim follows. □

For κ_W, duality is immediate from the above:

Proposition 3.2.14 ([DHNV20]). *For any linear subspace $W \subseteq \mathbb{R}^n$, we have $\kappa_W = \kappa_{W^\perp}$.*

Let us now show duality also for $\dot{\kappa}_W$; this was shown in [Lee89, Lemma 2.1] in a slightly different form.

Proposition 3.2.15. *For any rational linear subspace $W \subseteq \mathbb{R}^n$, we have $\dot{\kappa}_W = \dot{\kappa}_{W^\perp}$.*

Proof. Recall the p-adic valuation $\nu_p(n)$ defined in (2.1). It suffices to show that $\nu_p(\dot{\kappa}_W) = \nu_p(\dot{\kappa}_{W^\perp})$ for any prime $p \in \mathbb{P}$. We can reformulate as

$$\begin{aligned}\nu_p(\dot{\kappa}_W) &= \nu_p\Big(\operatorname{lcm}\big\{\operatorname{lcm}(g^C) : C \in \mathcal{C}_W\big\}\Big)\\ &= \max\big\{\nu_p(\operatorname{lcm}(g^C)) : C \in \mathcal{C}_W\big\}\\ &= \max\Big\{\nu_p(\alpha) : i, j \in [n], \alpha \in \dot{\mathcal{K}}_{ij}^W\Big\}.\end{aligned}$$

Lemma 3.2.13 implies that the last expression is the same for W and $W^\perp$. □

We next show that κ_W and $\dot{\kappa}_W$ are monotone under projections and restrictions of the subspace.

Lemma 3.2.16. *For any linear subspace $W \subseteq \mathbb{R}^n$, $J \subseteq [n]$ and $i, j \in J$, we have*

$$\mathcal{K}_{ij}^{\pi_J(W)} \subseteq \mathcal{K}_{ij}^W, \quad \mathcal{K}_{ij}^{W_J} \subseteq \mathcal{K}_{ij}^W, \quad \dot{\mathcal{K}}_{ij}^{\pi_J(W)} \subseteq \dot{\mathcal{K}}_{ij}^W, \quad \text{and} \quad \dot{\mathcal{K}}_{ij}^{W_J} \subseteq \dot{\mathcal{K}}_{ij}^W.$$

Proof. Let $g \in \mathcal{E}(W_J)$. Then $(g, 0_{[n]\setminus J}) \in \mathcal{E}(W)$ and so $\mathcal{K}_{ij}^{W_J} \subseteq \mathcal{K}_{ij}^W$. Note that $\pi_J(W) = ((W^\perp)_J)^\perp$ and so by Lemma 3.2.13,

$$\mathcal{K}_{ij}^{\pi_J(W)} = \left\{\alpha^{-1} : \alpha \in \mathcal{K}_{ji}^{(W^\perp)_J}\right\} \subseteq \left\{\alpha^{-1} : \alpha \in \mathcal{K}_{ji}^{W^\perp}\right\} = \mathcal{K}_{ij}^W. \tag{3.10}$$

The same arguments extend to $\dot{\mathcal{K}}_{ij}$. □

Proposition 3.2.17. *For any linear subspace $W \subseteq \mathbb{R}^n$ and $J \subseteq [n]$, we have*

$$\kappa_{W_J} \le \kappa_W, \quad \kappa_{\pi_J(W)} \le \kappa_W, \quad \dot{\kappa}_{W_J} \le \dot{\kappa}_W, \quad \text{and} \quad \dot{\kappa}_{\pi_J(W)} \le \dot{\kappa}_W.$$

3.2.4 Matrix representations

Proposition 3.2.1 already tells us that any rational matrix of the form $\mathbf{A} = \begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix}$ is $1/\dot{\kappa}_{\mathbf{A}}$-integral, and according to Lemma 3.2.3, the inverse of every non-singular square submatrix of $\mathbf{A}$ is also $1/\dot{\kappa}_{\mathbf{A}}$-integral. It is natural to ask whether every linear subspace W can be represented as $W = \ker(\mathbf{A})$ for an *integer* matrix $\mathbf{A}$ with the same property on the inverse matrices.

We show that this is true if the dual space is anchored but false in general. Recall that this means that every elementary vector g^C, $C \in \mathcal{C}_{W^\perp}$ has a ± 1 entry. In particular, $\dot{\kappa}_W = p^\alpha$ for some prime number $p \in \mathbb{P}$ implies that both W and $W^\perp$ are anchored; in this case we also have $\kappa_W = \dot{\kappa}_W$.

In [Lee89, Section 7], it is shown that if B is a basis minimizing $|\det(\mathbf{A}_B)|$ for a full rank $\mathbf{A} \in \mathbb{R}^{m\times n}$, then every nonzero entry in $\mathbf{A}_B^{-1}\mathbf{A}$ is at least 1 in absolute value. Moreover, a simple greedy algorithm is proposed (called 1-OPT) that finds such a basis within m pivots for k-adic spaces. Our next statement can be seen as the variant of this for anchored-spaces, using the lcm-circuit imbalance $\dot{\kappa}_{\mathbf{A}}$. We note that finding a basis minimizing $|\det(\mathbf{A}_B)|$ is computationally hard in general [Kha95b].

Proposition 3.2.18. *Let $W \subseteq \mathbb{R}^n$, $\dim(W) = n - m$ be a rational subspace such that $W^\perp$ is an anchored space. Then there exists an integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ such that $\ker(\mathbf{A}) = W$, and*

(i) All entries of $\mathbf{A}$ divide $\dot{\kappa}_W$.

(ii) For all non-singular submatrices $\mathbf{M}$ of $\mathbf{A}$, $\mathbf{M}^{-1}$ is $\frac{1}{\dot{\kappa}_W}$-integral.

(iii) $\dot{\Delta}_{\mathbf{A}}$ is an integer divisor of $(\dot{\kappa}_W)^m$.

Proof. Let $\bar{\mathbf{A}} \in \mathbb{Q}^{m\times n}$ be an arbitrary matrix with $\ker(\bar{\mathbf{A}}) = W$. By performing row operations we can convert $\bar{\mathbf{A}}$ into $\mathbf{A} = \begin{bmatrix}\mathbf{D} & \mathbf{A}'\end{bmatrix} \in \mathbb{Z}^{m\times n}$ where $\mathbf{D} \in \mathfrak{D}_m$ is positive diagonal and $\mathbf{A}' \in \mathbb{Z}^{m\times(n-m)}$ (after possibly permuting the columns). If $\mathbf{D} = \mathbf{I}_m$, then we are already done. Property (i) follows by Proposition 3.2.1; property (ii) follows by Lemma 3.2.3, and property (iii) holds since $\det(\mathbf{M}) \cdot \det(\mathbf{M}^{-1}) = 1$, $\det(\mathbf{M}) \in \mathbb{Z}$, and $\det(\mathbf{M}^{-1})$ is $\frac{1}{(\dot{\kappa}_W)^m}$-integral.

If $\mathbf{D}$ is not the identity matrix, then we show that $\mathbf{A}$ can be brought to the form $\begin{bmatrix}\mathbf{I}_m & \mathbf{A}''\end{bmatrix}$ with an integer $\mathbf{A}''$ by performing further basis exchanges. Let us assume that $\gcd(\mathbf{A}^i) = 1$ for all rows $\mathbf{A}^i$, $i \in [m]$. By Lemma 3.2.13, $\mathbf{A}^i \in \mathcal{E}(W^\perp)$. Assume $\mathbf{D}_{ii} = \mathbf{A}_{ii} > 1$ for some $i \in [m]$. As $\mathbf{A}^i$ is a circuit and $W^\perp$ is anchored, there exists an index $k \in [n]$ such that $|\mathbf{A}_{ik}| = 1$.

Let us perform a basis exchange between columns i and k. That is, subtract integer multiples of row i from the other rows to turn column k into e^i. We then swap columns i and k and obtain the matrix again in the form $\begin{bmatrix}\mathbf{D}' & \mathbf{A}''\end{bmatrix}$. Notice that the matrix remains integral, $\mathbf{D}'_{ii} = 1$, and $\mathbf{D}'_{jj} = \mathbf{D}_{jj}$ for $j \in [m]$, $j \neq i$. Hence, repeating this procedure at most n times, we can convert the matrix to the integer form $\begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix}$, completing the proof. □

Note that the proof gives an algorithm to find such a basis representation using a Gaussian elimination and at most m additional pivot operations. If $W^\perp$ is not anchored, we show the following weaker statement.

Proposition 3.2.19. *Let $W \subseteq \mathbb{R}^n$, $\dim(W) = n - m$ be a rational subspace. Then there exists an integer matrix $\mathbf{A} \in \mathbb{Z}^{m\times n}$ with $\ker(\mathbf{A}) = W$ such that*

(i) All entries of $\mathbf{A}$ divide $\dot{\kappa}_W$;

(ii) For all non-singular submatrices $\mathbf{M}$ of $\mathbf{A}$, $\mathbf{M}^{-1}$ is $\frac{1}{(\dot{\kappa}_W)^2}$-integral.

(iii) $\dot{\Delta}_{\mathbf{A}}$ is an integer divisor of $(\dot{\kappa}_W)^m$.

Proof. The proof is an easy consequence of Proposition 3.2.1 and Lemma 3.2.3. Consider any basis form $\mathbf{A} = \begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix}$ with $\ker(\mathbf{A}) = W$ (after possibly permuting the columns). According to Proposition 3.2.1, all entries of $\mathbf{A}$ are $1/\dot{\kappa}_W$ integral. By Lemma 3.2.13, the rows $\mathbf{A}^i \in \mathcal{E}(W^\perp)$ for $i \in [m]$. We can write $\mathbf{A}^i = g^i/d_i$ for some $g^i \in \mathcal{E}(W^\perp) \cap \mathbb{Z}^m$ and $d_i \in \mathbb{Q}$ such that $\gcd(g^i) = 1$

for each $i \in [m]$. By the definition of $\dot{\kappa}_{\mathbf{A}}$, the entries of each g^i are divisors of $\dot{\kappa}_{\mathbf{A}}$. Since $\mathbf{A}_{ii} = 1$ it follows that $d_i \in \mathbb{Z}$ and $d_i \mid \dot{\kappa}_{\mathbf{A}}$. Let $\mathbf{D} \in \mathfrak{D}_m$ be the diagonal matrix with entries $\mathbf{D}_{ii} = d_i$. Then, $\bar{\mathbf{A}} = \mathbf{DA}$ is an integer matrix where all entries divide $\dot{\kappa}_{\mathbf{A}}$, proving (i). Part (ii) follows by Lemma 3.2.3 and noting that the subdeterminants get multiplied by a submatrix $\mathbf{D}^{-1}$.

For part (iii), let us start use a basis B such that $|\det(\mathbf{A}_B)|$ is *maximal*; w.l.o.g. assume $B = [m]$. Then, in the basis form $\begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix}$ for B, all subdeterminants are ≤ 1. This holds as for any submatrix $\mathbf{M} \in \mathbb{Q}^{k\times k}$ of $\mathbf{A}'$ with $\det(\mathbf{M}) \neq 0$ we have that augmenting the columns of $\mathbf{M}$ by the columns $i \in B$ such that i is not a row of $\mathbf{M}$ results in a basis B_M with $|\det(\mathbf{M})| = \left|\det\left(\begin{bmatrix}\mathbf{I}_m & \mathbf{A}'\end{bmatrix}_{B_M}\right)\right| \leq \det(\mathbf{I}_m) = 1$ by assumption on B. After multiplying by $\mathbf{D}$ as above, $\bar{\mathbf{A}} = \mathbf{DA}$, all subdeterminants will be $\leq \det(\mathbf{D}) \leq (\dot{\kappa}_{\mathbf{A}})^m$. □

Note that parts (i) and (ii) are true for any choice of the basis form, whereas (iii) requires one to select $\mathbf{A}_B$ with maximum determinant. The maximum subdeterminant is NP-hard even to approximate better than c^m for some $c > 1$ [DEFM14]. However, it is easy to see that even if we start with an arbitrary basis, then $\dot{\Delta}_{\mathbf{A}} \mid (\dot{\kappa}_W)^{2m}$, since every subdeterminant of $\mathbf{A}_B^{-1}\mathbf{A}$ is at most $(\dot{\kappa}_W)^m$ follows by Lemma 3.2.3.

We now give an example to illustrate why Proposition 3.2.18(ii) cannot hold for arbitrary values of $\dot{\kappa}_W$. The proof is given in the Appendix.

Example 3.2.20. *Consider the matrix*

$$\mathbf{A} = \begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix}. \tag{3.11}$$

For this matrix $\dot{\kappa}_{\mathbf{A}} = 5850 = 2\times 3^2 \times 5^2 \times 13$ *holds, and there exists no* $\tilde{\mathbf{A}} \in \mathbb{Z}^{2\times 4}$ *such that* $\ker(\tilde{\mathbf{A}}) = \ker(\mathbf{A})$ *and the inverse of every nonsingular* 2×2 *submatrix of* $\tilde{\mathbf{A}}$ *is* $1/5850$*-integral.*

3.2.5 A basic matroid property

We need some simple concepts and results from matroid theory. We refer the reader to [Sch03, Chapter 39] or [Fra11, Chapter 5] for definitions and background. Let $\mathcal{M} = ([n], \mathcal{I})$ be a matroid on ground set $[n]$ with independent sets $\mathcal{I} \subseteq 2^{[n]}$. The rank $\mathrm{rk}(S)$ of a set $S \subseteq [n]$ is the maximum size of an independent set contained in S. The maximal independent sets are called *bases*. All bases have the same cardinality $\mathrm{rk}([n])$.

For the matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, we will work with the linear matroid $\mathcal{M}(\mathbf{A}) = ([n], \mathcal{I}(\mathbf{A}))$, where a subset $I \subseteq [n]$ is independent if the columns $\{\mathbf{A}_i : i \in I\}$ are linearly independent. Note that $\mathrm{rk}([n]) = m$ under the assumption that A has full row rank.

Recall that the circuits of the matroid are the inclusion-wise minimal non-independent sets. Let $I \in \mathcal{I}$ be an independent set, and $i \in [n] \setminus I$ such that $I \cup \{i\} \notin \mathcal{I}$. Then, there exists a unique circuit $C(I, i) \subseteq I \cup \{i\}$ that is called the *fundamental circuit* of i with respect to I. Note that $i \in C(I, i)$.

Recall that the matroid $\mathcal{M}$ is separable if the ground set $[n]$ can be partitioned to two nonempty subsets $[n] = S \cup T$ such that $I \in \mathcal{I}$ if and only if $I \cap S, I \cap T \in \mathcal{I}$. In this case, the matroid is the direct sum of its restrictions to S and T. In particular, every circuit is fully contained in S or in T.

For the linear matroid $\mathcal{M}(\mathbf{A})$, separability means that $\ker(\mathbf{A}) = \ker(\mathbf{A}_S) \times \ker(\mathbf{A}_T)$. In this case, solving System 1.1 can be decomposed into two subproblems, restricted to the columns in $\mathbf{A}_S$ and in $\mathbf{A}_T$, and $\kappa_{\mathbf{A}} = \max\{\kappa_{\mathbf{A}_S}, \kappa_{\mathbf{A}_T}\}$.

Hence, we can focus on *non-separable* matroids. The following characterization is well-known, see e.g. [Fra11, Theorems 5.2.5, 5.2.7–5.2.9]. For a hypergraph $H = ([n], \mathcal{E})$, we define the underlying

graph $H_G = ([n], E)$ such that $(i, j) \in E$ if there is a hyperedge $S \in \mathcal{E}$ with $i, j \in S$. That is, we add a clique corresponding to each hyperedge. The hypergraph is called *connected* if the underlying graph $G = ([n], E)$ is connected.

Proposition 3.2.21. *For a matroid $\mathcal{M} = ([n], \mathcal{I})$, the following are equivalent:*

(i) *$\mathcal{M}$ is non-separable.*

(ii) *The hypergraph of the circuits is connected.*

(iii) *For any base B of $\mathcal{M}$, the hypergraph formed by the fundamental circuits $\mathcal{C}^B = \{ C(B, i) : i \in [n] \setminus B \}$ is connected.*

(iv) *For any $i, j \in [n]$, there exists a circuit containing i and j.*

Proof. The implications (i) $\Leftrightarrow$ (ii), (iii) $\Rightarrow$ (ii), and (iv) $\Rightarrow$ (ii) are immediate from the definitions.

For the implication (ii) $\Rightarrow$ (iii), assume for a contradiction that the hypergraph of the fundamental circuits with respect to B is not connected. This means that we can partition $[n] = S \cup T$ such that for each $i \in S$, $C(B, i) \subseteq S$, and for each $i \in T$, $C(B, i) \subseteq T$. Consequently, $\mathrm{rk}(S) = |B \cap S|$, $\mathrm{rk}(T) = |B \cap T|$, and therefore $\mathrm{rk}([n]) = \mathrm{rk}(S) + \mathrm{rk}(T)$. It is easy to see that this property is equivalent to separability to S and T; see e.g. [Fra11, Theorem 5.2.7] for a proof.

Finally, for the implication (ii) $\Rightarrow$ (iv), consider the undirected graph $([n], E)$ where $(i, j) \in E$ if there is a circuit containing both i and j. This graph is transitive according to [Fra11, Theorem 5.2.5]: if $(i, j), (j, k) \in E$, then also $(i, k) \in E$. Consequently, whenever $([n], E)$ is connected, it must be a complete directed graph. □

3.2.6 The triangle inequality

An interesting additional fact about circuit imbalances is that the logarithm of the weights satisfy the triangle inequality; this was shown in [DHNV20]. Here, we formulate a stronger version and give a simpler proof. Throughout, we assume that $\mathcal{M}(W)$ is non-separable. Thus, according to Proposition 3.2.21, for any $i, j \in [n]$ there is a circuit $C \in \mathcal{C}_W$ with $i, j \in C$.

Theorem 3.2.22. *Let $W \subseteq \mathbb{R}^n$ be a linear space, and assume $\mathcal{M}(W)$ is non-separable. Then,*

(i) *for any distinct $i, j, k \in [n]$, $\mathcal{K}^W_{ij} \subseteq \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$; and*

(ii) *for any distinct $i, j, k \in [n]$, $\kappa_{ij} \leq \kappa_{ik} \cdot \kappa_{kj}$.*

Proof of part (i) in [DHNV20]. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a full-rank matrix with $W = \ker(\mathbf{A})$. If $C = \{i, j\}$, then the columns $\mathbf{A}_i, \mathbf{A}_j$ are linearly dependent. Writing $\mathbf{A}_i = \lambda \mathbf{A}_j$, we have $\lambda = -g^C_j / g^C_i$. Let h be any circuit solution with $i, k \in \mathrm{supp}(h)$, and hence $j \notin \mathrm{supp}(h)$. By assumption, the vector $h' = h - h_i e^i + \lambda h_i e^j$ will satisfy $\mathbf{A}h' = \mathbf{0}$ and have $i \notin \mathrm{supp}(h')$, $j, k \in \mathrm{supp}(h')$. We know that h' is a circuit solution, because any circuit $C' \subset \mathrm{supp}(h')$ could, by the above process in reverse, be used to produce a kernel solution with strictly smaller support than h, contradicting the assumption that h is a circuit solution. Now we have $|h'_j / h'_k| \cdot |h_k / h_i| = |h'_j / h_i| = |\lambda|$ by construction. Thus, h and h' are the circuit solutions we are looking for.

Now assume $C \neq \{i, j\}$. If $k \in C$, the statement is trivially true with $C = C_1 = C_2$, so assume $k \notin C$. Pick $l \in C$, $l \notin \{i, j\}$ and set $B = C \setminus \{l\}$. Assume without loss of generality that $B \subseteq [m]$ and apply row operations to $\mathbf{A}$ such that $\mathbf{A}_{B,B} = \mathbf{I}_{B \times B}$ is an identity submatrix and $\mathbf{A}_{[m] \setminus B, B} = \mathbf{0}$. Then the column $\mathbf{A}_l$ has support given by B, for otherwise g^C could not be in the kernel. The given circuit solution satisfies $g^C_t = -\mathbf{A}_{t,l} g^C_l$ for all $t \in B$, and in particular $g^C_j / g^C_i = \mathbf{A}_{j,l} / \mathbf{A}_{i,l}$.

Take any circuit solution $h \in \ker(\mathbf{A})$ such that $l, k \in \operatorname{supp}(h)$ and such that $C \cup \operatorname{supp}(h)$ is inclusion-wise minimal. Such a vector exists by Proposition 3.2.21(iv). Now let $J = \operatorname{supp}(h) \setminus C$. Because $\mathbf{A}_{[m]\setminus B,C} = \mathbf{0}$ and $\mathbf{A}h = \mathbf{0}$, we must have $\mathbf{0} \neq h_J \in \ker(\mathbf{A}_{[m]\setminus B,J})$. We show that we can uniquely lift any vector $x \in \ker(\mathbf{A}_{B,C\cup\{k\}})$ to a vector $x' \in \ker(\mathbf{A}_{C\cup J})$ with $x'_{C\cup k} = x$. Since this lift will send circuit solutions to circuit solutions by uniqueness, it suffices to find our desired circuits as solutions to the smaller linear system.

We first prove that $\dim(\ker(\mathbf{A}_{[m]\setminus B,J})) = 1$. For suppose that $\dim(\ker(\mathbf{A}_{[m]\setminus B,J})) \geq 2$, then $|J| \geq 2$ and there would exist some vector $y \in \ker(\mathbf{A}_{[m]\setminus B,J})$ linearly independent of h_J with $k \in \operatorname{supp}(y)$. This vector could be uniquely lifted to a vector $\bar{y} \in \ker(\mathbf{A})$, and we could then find a linear combination $h + \alpha\bar{y}$ such that $\operatorname{supp}(h + \alpha\bar{y}) \subsetneq C \cup J$ but $l, k \in \operatorname{supp}(h + \alpha\bar{y})$. The existence of such a vector contradicts the minimality of $C \cup \operatorname{supp}(h)$. As such, we know that $\dim(\ker(\mathbf{A}_{[m]\setminus B,J})) = 1$.

This clear linear relation between any two entries in J for any vector in $\ker(\mathbf{A}_{[m]\setminus B,J})$ implies that we can apply row operations to $\mathbf{A}$ such that $\mathbf{A}_{B,J}$ has non-zero entries only in the column $\mathbf{A}_{B,\{k\}}$. Note that these row operations leave $\mathbf{A}_C$ unchanged because $\mathbf{A}_{[m]\setminus B,C} = 0$. From this, we can see that any element in $\ker(\mathbf{A}_{B,C\cup\{k\}})$ can be uniquely lifted to an element in $\ker(\mathbf{A}_{C\cup J})$. Hence we can focus on $\ker(\mathbf{A}_{B,C\cup\{k\}})$.

If $\mathbf{A}_{i,k} = \mathbf{A}_{j,k} = 0$, then any $x \in \ker(\mathbf{A}_{B,C\cup\{k\}})$ satisfies $x_i + \mathbf{A}_{i,l}x_l = x_j + \mathbf{A}_{j,l}x_l = 0$ and, in particular, any circuit $l, k \in \bar{C} \subset C \cup \{k\}$ contains $\{i, j\} \subset \bar{C}$ and fulfills $|g^C_j / g^C_i| = |\mathbf{A}_{j,l}/\mathbf{A}_{i,l}| = |g^{\bar{C}}_j / g^{\bar{C}}_i| = |g^{\bar{C}}_j / g^{\bar{C}}_k||g^{\bar{C}}_k / g^{\bar{C}}_i|$. Choosing $C_1 = C_2 = \bar{C}$ concludes the case.

Otherwise, we know that $\mathbf{A}_{i,k} \neq 0$ or $\mathbf{A}_{j,k} \neq 0$, meaning that $\ker(\mathbf{A}_{\{i,j\},\{i,j,l,k\}})$ contains at least one circuit solution with k in its support. Observe that any circuit in $\ker(\mathbf{A}_{\{i,j\},\{i,j,l,k\}})$ can be lifted uniquely to an element in $\ker(\mathbf{A}_{B,C\cup\{k\}})$ since $\mathbf{A}_{B,B}$ is an identity matrix and we can set the entries of $B \setminus \{i, j\}$ individually to satisfy the equalities. Note that this lifted vector is a circuit as well, again by uniqueness of the lift. Hence, we may restrict our attention to the matrix $\mathbf{A}_{\{i,j\},\{i,j,l,k\}}$. If the columns $\mathbf{A}_{\{i,j\},k}, \mathbf{A}_{\{i,j\},l}$ are linearly dependent, then any circuit solution to $\mathbf{A}_{\{i,j\},\{i,j,l\}}x = 0, x_l \neq 0$, such as $g^C_{\{i,j,l\}}$, is easily transformed into a circuit solution to $\mathbf{A}_{\{i,j\},\{i,j,k\}}x = 0, x_k \neq 0$, and we are done.

If $\mathbf{A}_{\{i,j\},k}, \mathbf{A}_{\{i,j\},l}$ are independent, we can write $\mathbf{A}_{\{i,j\},\{i,j,l,k\}} = \left(\begin{smallmatrix} 1 & 0 & a & c \\ 0 & 1 & b & d \end{smallmatrix}\right)$, where $g^C_j / g^C_i = b/a$. For $\alpha = ad - bc$, which is non-zero since $\alpha = \det(\left(\begin{smallmatrix} a & c \\ b & d \end{smallmatrix}\right)) \neq 0$ by the independence assumption, we can check that $(\alpha, 0, -d, b)^\top$ and $(0, \alpha, c, -a)^\top$ are the circuits we are looking for. □

The alternative proof of Theorem 3.2.22 relies on the following technical lemma that analyzes the scenario when almost all vectors in W are elementary.

Lemma 3.2.23. *Let $W \subseteq \mathbb{R}^n$ be a subspace s.t. $\mathcal{M}(W)$ is non-separable.*

(i) If $\mathcal{E}(W) = \{g \in W \setminus \{0\} : \operatorname{supp}(g) \neq [n]\}$, then $\mathcal{K}^W_{ij} \subseteq \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$.

(ii) If there exists $g \in \mathcal{E}(W)$ such that $|\operatorname{supp}(g)| = n - 1$, then

$$\mathcal{E}(W) = \{ g \in W \setminus \{0\} : \operatorname{supp}(g) \neq [n] \}.$$

Proof. For part (i), let $\delta \in \mathcal{K}^W_{ij}$ and let $g \in \mathcal{E}(W)$ such that $\{i, j\} \subseteq \operatorname{supp}(g)$ and $|g_j/g_i| = \delta$. If $k \in \operatorname{supp}(g)$, then $|g_j/g_i| = |g_k/g_i| \cdot |g_j/g_k|$ shows the claim.

Assume $k \notin \operatorname{supp}(g)$, and pick $h \in \mathcal{E}(W)$ such that $\{i, k\} \subseteq \operatorname{supp}(h)$ and let $\tilde{h} = h_j g - g_j h$; such a h exists by Proposition 3.2.21. Then $\tilde{h}_j = 0$ and $\tilde{h}_k \neq 0$, so $\tilde{h} \in \mathcal{E}(W)$ by the assumption. If $\tilde{h}_i = 0$ then $h_j g_i = g_j h_i$ and so $\{i, j, k\} \subseteq \operatorname{supp}(h)$ with $h_j/h_i = g_j/g_i$, therefore h certifies the statement

as $|h_j/h_i| = |h_k/h_i| \cdot |h_j/h_k|$. Otherwise, $\tilde{h}_i \neq 0$ and $h' := \tilde{h}_i g - g_i \tilde{h}$ fulfills $h' \in \mathcal{E}(W)$ as $h'_i = 0$, $\{j,k\} \subseteq \operatorname{supp}(h')$. Now, using that $\tilde{h}_j = 0$ and $g_k = 0$ it is easy to see that

$$\left|\frac{\tilde{h}_k}{\tilde{h}_i} \cdot \frac{h'_j}{h'_k}\right| = \left|\frac{\tilde{h}_k}{\tilde{h}_i} \cdot \frac{\tilde{h}_i g_j - g_i \tilde{h}_j}{\tilde{h}_i g_k - g_i \tilde{h}_k}\right| = \left|\frac{\tilde{h}_k}{\tilde{h}_i} \cdot \frac{\tilde{h}_i g_j}{g_i \tilde{h}_k}\right| = \left|\frac{g_j}{g_i}\right|. \tag{3.12}$$

We now turn to part (ii). Since there exists $g \in \mathcal{E}(W)$ with $\operatorname{supp}(g) \neq n$, we cannot have $[n] \in \mathcal{C}(W)$.

Let $g \in \mathcal{E}(W)$ and $i \in [n]$ such that $\operatorname{supp}(g) = [n] \setminus \{i\}$. Consider any $h \in W$, $\operatorname{supp}(h) \neq \operatorname{supp}(g)$ such that $\operatorname{supp}(h) \neq [n]$. If $h \notin \mathcal{E}(W)$ there exists $\ell \in \mathcal{E}(W)$ such that $\operatorname{supp}(\ell) \subsetneq \operatorname{supp}(h)$. We must have $i \in \operatorname{supp}(\ell)$, since $\operatorname{supp}(\ell) \setminus \operatorname{supp}(g) \neq \emptyset$. Then $\tilde{h} := h_i \ell - \ell_i h$ fulfills $\tilde{h} \neq \mathbf{0}$, $\tilde{h}_i = 0$ and $\operatorname{supp}(\tilde{h}) \subsetneq [n] \setminus \{i\}$, a contradiction to $g \in \mathcal{E}(W)$. □

Proof of Theorem 3.2.22. Part (ii) immediately follows from part (i), when taking $C \in \mathcal{C}_W$ such that $|g^C_j / g^C_i| = \kappa_{ij}$. We now prove part (i).

Let $\delta \in \mathcal{K}^W_{ij}$ and $C \in \mathcal{C}_W$ such that $i,j \in C$ and for $g = g^C$, $|g_j/g_i| = \delta$. If $k \in C$ then $|g_j/g_i| = |g_k/g_i| \cdot |g_j/g_k| \in \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$. Otherwise, let us select $C' \in \mathcal{C}_W$ such that $i,k \in C'$, and $|C \cup C'|$ is minimal. Let $h = g^{C'}$ and $J = C' \setminus (C \cup \{k\})$.

Claim 3.2.23.1. *Let $G = (C \cup C') \setminus J$. Then for the space $\hat{W} := \pi_G(W_{C \cup C'})$ we have that $g_G, h_G \in \mathcal{E}(\hat{W})$.*

Proof. The statement that $h_G \in \mathcal{E}(\hat{W})$ is clear as $h_{C \cup C'} \in \mathcal{E}(W_{C \cup C'})$ and the variables we project out J fulfill $J \subseteq \operatorname{supp}(h)$. For the statement on g_G assume that there exists $\hat{g} \in \mathcal{E}(\hat{W})$ such that $\operatorname{supp}(\hat{g}) \subsetneq \operatorname{supp}(g_G)$. Then there exists a lift $\tilde{g} \in \mathcal{E}(W_{C \cup C'})$ of $\hat{g}$ and some $\ell \in J$ such that $\ell \in \operatorname{supp}(\tilde{g})$; note also that $\tilde{g}_k = g_k = 0$. The vector $\hat{h} := h_\ell \tilde{g} - \tilde{g}_\ell h$ fulfills $\ell \notin \operatorname{supp}(\hat{h})$ and $k \in \operatorname{supp}(\hat{h})$.

Now pick any circuit $\tilde{h} \in \mathcal{E}(\hat{W})$ such that $k \in \operatorname{supp}(\tilde{h})$ and $\operatorname{supp}(\tilde{h}) \subseteq \operatorname{supp}(\hat{h})$. Note that $J \cup \{k\}$ is independent, as $J \cup \{k\} \subseteq C' \setminus \{i\} \subsetneq C'$. Therefore, $\operatorname{supp}(\tilde{h}) \cap \operatorname{supp}(g) \neq \emptyset$. Hence, for $T := C \cup \operatorname{supp}(\tilde{h})$ we have that $\mathcal{M}(W_T)$ is non-separable. In particular there exists a circuit $h' \in \mathcal{E}(W_T)$ such that $i,k \in \operatorname{supp}(h')$. As $T \subseteq (C \cup C') \setminus \{\ell\}$, this is a contradiction to the minimal choice of C'. ■

As $\operatorname{supp}(h_D) \cup \operatorname{supp}(g_D) = D$ and $\operatorname{supp}(h_D) \cap \operatorname{supp}(g_D) \neq \emptyset$ we have that $\mathcal{E}(W')$ is non-separable. Further $|\operatorname{supp}(g_D)| = |D| - 1$, so we can apply Lemma 3.2.23 to learn $\delta \in \mathcal{K}^{W'}_{ij} \subseteq \mathcal{K}^{W'}_{ik} \cdot \mathcal{K}^{W'}_{kj}$. We can conclude $\delta \in \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$ from Lemma 3.2.16. □

If $\kappa_W = 1$, then the reverse inclusion $\mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj} \subseteq \mathcal{K}^W_{ij}$ trivially holds, since 1 is the only element in these sets. In Proposition 3.4.9, we give a necessary and sufficient condition for $\mathcal{K}^W_{ij} = \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$.

One may ask under which circumstances an element $\alpha \in \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$ is also contained in $\mathcal{K}^W_{ij}$. We give a partial answer by stating a sufficient condition in a restrictive setting. For a basis B of $\mathcal{M}(W)$, recall $\mathcal{E}_B(W^\perp)$ from Lemma 3.2.12. Then, Lemmas 3.2.12 and 3.2.13 together imply:

Lemma 3.2.24. *Given a basis $B \subseteq [n]$ in $\mathcal{M}(W)$ and $g,h \in \mathcal{E}_B \subseteq \mathcal{E}(W^\perp)$ such that $i \in \operatorname{supp}(g) \cap B$, $j \in \operatorname{supp}(h) \cap B$ and $k \in \operatorname{supp}(g) \cap \operatorname{supp}(h)$. Then $|h_j/h_k| \cdot |g_k/g_i| \in \mathcal{K}^W_{ij}$.*

3.3 Connections to other condition numbers

3.3.1 The condition number $\bar{\chi}$ and the lifting operator

For a full row rank matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, the condition number $\bar{\chi}_{\mathbf{A}}$ can be defined in the following two equivalent ways:

$$\begin{aligned}\bar{\chi}_{\mathbf{A}} &:= \sup\left\{ \left\| \mathbf{A}^\top (\mathbf{A}\mathbf{D}\mathbf{A}^\top)^{-1}\mathbf{A}\mathbf{D} \right\| : \mathbf{D} \in \mathfrak{D}_n \right\} \\ &= \sup\left\{ \frac{\|\mathbf{A}^\top y\|}{\|p\|} : y \text{ minimizes } \left\|\mathbf{D}^{1/2}(\mathbf{A}^\top y - p)\right\| \text{ for some } 0 \neq p \in \mathbb{R}^n \text{ and } \mathbf{D} \in \mathfrak{D}_n \right\}.\end{aligned} \tag{3.13}$$

We provide a further definition which might help in developing some intuition:

$$\bar{\chi}_{\mathbf{A}} = \sup\left\{ \left\|\mathbf{D}^{-1}\Pi_{\mathbf{D}\mathbf{A}^\top}\mathbf{D}\right\| : \mathbf{D} \in \mathfrak{D}_n \right\}. \tag{3.14}$$

The interpretation in Equation (3.14) is as follows. If the matrix in the argument is applied to a vector $y \in \mathbb{R}^n$, then we first rescale the entries according to the norm corresponding to $\mathbf{D}$ to obtain $\mathbf{D}y$. Then, an orthogonal projection of $\mathbf{D}y$ onto the rescaled space $\mathrm{im}(\mathbf{D}\mathbf{A}^\top)$ is performed. Finally, the rescaling is undone via multiplication by $\mathbf{D}^{-1}$.

This condition number was first studied by Dikin [Dik67], Stewart [Ste89], and Todd [Tod90]. There is an extensive literature on the properties and applications of $\bar{\chi}_{\mathbf{A}}$, as well as its relations to other condition numbers. In particular, it plays a key role in layered-least-squares interior point methods, see Chapter 4. We refer the reader to the papers [HT02; MT03; VY96] for further results and references.

It is important to note that—similarly to $\kappa_{\mathbf{A}}$ and $\dot{\kappa}_{\mathbf{A}}$—$\bar{\chi}_{\mathbf{A}}$ only depends on the subspace $W = \ker(\mathbf{A})$. Hence, we can also write $\bar{\chi}_W$ for a subspace $W \subseteq \mathbb{R}^n$, defined to be equal to $\bar{\chi}_{\mathbf{A}}$ for some matrix $\mathbf{A} \in \mathbb{R}^{k\times n}$ with $W = \ker(\mathbf{A})$. We will use the notations $\bar{\chi}_{\mathbf{A}}$ and $\bar{\chi}_W$ interchangeably. The following characterization reveals the connection between $\kappa_{\mathbf{A}}$ and $\bar{\chi}_{\mathbf{A}}$.

Proposition 3.3.1 ([TTY01]). *For a full row rank matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$,*

$$\bar{\chi}_{\mathbf{A}} = \max\left\{ \|\mathbf{A}_B^{-1}\mathbf{A}\| : \mathbf{A}_B \text{ is a non-singular } m \times m\text{-submatrix of } \mathbf{A} \right\}. \tag{3.15}$$

Together with Proposition 3.2.1, this shows that the difference between $\bar{\chi}_{\mathbf{A}}$ and $\kappa_{\mathbf{A}}$ is in using ℓ_2 instead of ℓ_∞ norm. This immediately implies the upper bound and a slightly weaker lower bound in the next theorem.

Approximating the condition number $\bar{\chi}_{\mathbf{A}}$ is known to be hard; by the same token, $\bar{\kappa}_{\mathbf{A}}$ also cannot be approximated by any polynomial factor. The proof relies on the hardness of approximating the minimum subdeterminant by Khachiyan [Kha95b].

Theorem 3.3.2 (Tunçel [Tun99]). *Approximating $\bar{\chi}_{\mathbf{A}}$ up to a factor of $2^{\mathrm{poly}(n)}$ is NP-hard.*

The next lemma summarizes some important known properties of $\bar{\chi}_{\mathbf{A}}$.

Proposition 3.3.3. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$ with full row rank and $W = \ker(\mathbf{A})$.*

(i) If the entries of $\mathbf{A}$ are all integers, then $\bar{\chi}_{\mathbf{A}}$ is bounded by $2^{O(L_{\mathbf{A}})}$, where $L_{\mathbf{A}}$ is the input bit length of $\mathbf{A}$.

(ii) $\bar{\chi}_W = \bar{\chi}_{W^\perp}$.

Proof. Part (i) was proved in [VY96, Lemma 24]. The duality statement (ii) was shown in [GL97]. □

In connection with $\bar{\chi}_{\mathbf{A}}$, it is worth mentioning the *lifting map*, a key concept in the algorithms presented in Chapter 4.

Definition 3.3.4. *For $\emptyset \neq I \subseteq [n]$ we define $L_I^W : \pi_I(W) \to W$ by*

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

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 $W \cap \mathbb{R}^n_{[n]\setminus I}$.

Lemma 3.3.5. *Let $W \subseteq \mathbb{R}^n$ be an $(n-m)$-dimensional linear subspace. Let the columns of $\mathbf{B} \in \mathbb{R}^{n\times(n-m)}$ denote an orthonormal basis of W. Then, viewing L_I^W as a matrix in $\mathbb{R}^{n\times|I|}$,*

$$L_I^W = \mathbf{B}\mathbf{B}^\dagger_{I,\bullet}.$$

Proof. If $p \in \pi_I(W)$, then $p = \mathbf{B}_{I,\bullet} y$ for some $y \in \mathbb{R}^{n-m}$. By the well-known property of the pseudo-inverse we get $\mathbf{B}^\dagger_{I,\bullet} p = \arg\min_{p=\mathbf{B}_{I,\bullet} y} \|y\|$. This solution satisfies $\pi_I(\mathbf{B}\mathbf{B}^\dagger_{I,\bullet} p) = p$ and $\mathbf{B}\mathbf{B}^\dagger_{I,\bullet} p \in W$. Since the columns of $\mathbf{B}$ form an orthonormal basis of W, we have $\|\mathbf{B}\mathbf{B}^\dagger_{I,\bullet} p\| = \|\mathbf{B}^\dagger_{I,\bullet} p\|$. Consequently, $\mathbf{B}\mathbf{B}^\dagger_{I,\bullet} p$ is the minimum-norm point with the above properties. □

The condition number $\bar{\chi}_W$ can be equivalently defined as the maximum norm of any lifting map for an index subset.

Proposition 3.3.6 ([DHNV20; OLe90; Ste89])**.** *For a linear subspace $W \subseteq \mathbb{R}^n$,*

$$\bar{\chi}_W = \max\left\{ \|L_I^W\| : I \subseteq [n], I \neq \emptyset \right\}. \tag{3.16}$$

Proof. With Lemma 3.3.5 the statement is equivalent to

$$\bar{\chi}_W = \max\left\{ \|\mathbf{B}\mathbf{B}^\dagger_{I,\bullet}\| : \emptyset \neq I \subseteq [n] \right\}. \tag{3.17}$$

The direction $\geq$ was proved in [Ste89], and the direction $\leq$ in [OLe90]. □

Even though L_I^W is defined with respect to the ℓ_2-norm, it can also be used to characterize κ_W.

Proposition 3.3.7 ([DNV20])**.** *For a linear subspace $W \subseteq \mathbb{R}^n$,*

$$\kappa_W = \max\left\{ \frac{\|L_I^W(p)\|_\infty}{\|p\|_1} : I \subseteq [n], I \neq \emptyset, p \in \pi_I(W) \setminus \{\mathbf{0}\} \right\}. \tag{3.18}$$

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 conformal decomposition $z = \sum_{k=1}^h g^k$ as in Lemma 2.0.3. For each $k \in [h]$, let $C_k = \operatorname{supp}(g^k)$. 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.

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

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

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 elementary vector and $i, j \in C$ such that $\kappa_W = |g^C_j / g^C_i|$.

Let us set $I = ([n] \setminus C) \cup \{i\}$, and define $p_k = 0$ if $k \in [n] \setminus C$ and $p_i = g^C_i$. Then $p \in \pi_I(W)$, and the unique extension to W is g^C; thus, $L^W_I(p) = g^C$. We have $\|L^W_I(p)\|_\infty = |g^C_j|$. Noting that $\|p\|_1 = |g^C_i|$, it follows that $\kappa_W = \|L^W_I(p)\|_\infty / \|p\|_1$. □

The following theorem closely relates κ and $\bar{\chi}$. Related results have already appeared in [HT02; Vav94].

Theorem 3.3.8 ([DHNV20; DNV20]). *For a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ we have $\sqrt{1 + \kappa^2_{\mathbf{A}}} \le \bar{\chi}_{\mathbf{A}} \le n\kappa_{\mathbf{A}}$.*

Proof. For the first inequality, let $C \in \mathcal{C}_W$ be the circuit and $i \neq j \in C$ such that $|g_j/g_i| = \kappa_W$ for the corresponding solution $g = g^C$. Let us use the characterization of $\bar{\chi}_W$ in Proposition 3.3.6. Let $I = ([n] \setminus C) \cup \{i\}$, and $p = g_i e^i$, that is, the vector with $p_i = g_i$ and $p_k = 0$ for $k \neq i$. Then, the unique vector $z \in W$ such that $z_I = p$ is $z = g$. Therefore,

$$\bar{\chi}_W \ge \min_{z \in W, z_I = p} \frac{\|z\|}{\|p\|} = \frac{\|g\|}{|g_i|} \ge \frac{\sqrt{|g_i|^2 + |g_j|^2}}{|g_i|} = \sqrt{1 + \kappa^2_W}\,.$$

The second inequality is immediate from Proposition 3.3.6 and Proposition 3.3.7, and the inequalities between ℓ_1, ℓ_2, and ℓ_∞ norms. The proof of the slightly weaker $\bar{\chi}_W \le \sqrt{1 + (n\kappa_W)^2}$ follows from Lemma 4.2.2. □

The next lemma will be needed to prove Lemma 4.2.2 and also to analyze the LLS algorithm. Let us say that the vector $y \in \mathbb{R}^n$ *conforms to* $x \in \mathbb{R}^n$ if $x_i y_i > 0$ whenever $y_i \neq 0$.

Lemma 3.3.9. *For $i \in I \subseteq [n]$ with $e^i_I \in \pi_I(W)$, let $z = L^W_I(e^i_I)$. Then for any $j \in \operatorname{supp}(z)$ we have $\kappa^W_{ij} \ge |z_j|$.*

Proof. We consider the cone $F \subseteq W$ of vectors that conform to z. The faces of F are bounded by inequalities of the form $z_k y_k \ge 0$ or $y_k = 0$. The edges (rays) of F are of the form $\{\, \alpha g : \alpha \ge 0 \,\}$ with $\operatorname{supp}(g) \in \mathcal{C}_W$. It is easy to see from the Minkowski–Weyl theorem that z can be written as

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

where $h \le n$, $C_1, C_2, \ldots, C_h \in \mathcal{C}_W$ are circuits, and the vectors $g^1, g^2, \ldots, g^h \in W$ conform to z and $\operatorname{supp}(g^k) = C_k$ for all $k \in [h]$. Note that $i \in C_k$ for all $k \in [h]$, as otherwise, $z' = z - g^k$ would also satisfy $z'_I = e^i_I$, but $\|z'\| < \|z\|$ due to g^k being conformal to z, a contradiction to the definition of z.

At least one $k \in [h]$ contributes at least as much to $|z_j| = \frac{\sum_{k=1}^h |g^k_j|}{\sum_{k=1}^h g^k_i}$ as the average. Hence we find $\kappa^W_{ij} \ge |g^k_j / g^k_i| \ge |z_j|$. □

3.3.2 The condition number δ and bounds on diameters of polyhedra

Another related condition number is δ, defined as follows:

Definition 3.3.10. *Let $V \subseteq \mathbb{R}^n$ be a set of vectors. Then δ_V is the largest value such that for any set of linearly independent vectors $\{v_i : i \in I\} \subseteq V$ and $\lambda \in \mathbb{R}^I$,*

$$\left\| \sum_{i \in I} \lambda_i v_i \right\| \geq \delta_V \max_{i \in I} |\lambda_i| \cdot \|v_i\| .$$

For a matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$, we let $\delta_{\mathbf{M}}$ denote the value associated with the rows $\mathbf{M}^1, \mathbf{M}^2, \ldots, \mathbf{M}^m$ of $\mathbf{M}$.

This can be equivalently characterized as follows: for a subset $\{ v_i : i \in I \} \subseteq V$ and $v_j \in V$, $v_j \notin W = \operatorname{span}(\{ v_i : i \in I \})$, the sine of the angle between the vector v_j and the subspace W is at least δ_V (see e.g., for the equivalence [DVZ]).

A line of work studied this condition number in the context of the simplex algorithm and diameter bounds. The diameter of a polyhedron P is the diameter of the vertex-edge graph associated with P; Hirsch's famous conjecture from 1957 asserted that the diameter of a polytope (a bounded polyhedron) in n dimensions with m facets is at most $m - n$. This was disproved by Santos in 2012 [San12], but the polynomial Hirsch conjecture, i.e., a poly(n, m) diameter bound remains wide open.

Consider the LP in standard inequality form with n variables and m constraints as

$$\max \langle c, x \rangle \text{ s.t. } x \in P , \quad P = \{ x \in \mathbb{R}^n : \mathbf{A}x \leq b \} , \tag{3.19}$$

for $\mathbf{A} \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^m$. Using a randomized dual simplex algorithm, Dyer and Frieze [DF94] showed the polynomial Hirsch conjecture for TU matrices. Bonifas et al. [Bon+14] strengthened and extended this to the bounded subdeterminant case, showing a diameter bound of $O(n^4 \Delta_{\mathbf{A}}^2 \log(n\Delta_{\mathbf{A}}))$ for integer constraint matrices $\mathbf{A} \in \mathbb{Z}^{m \times n}$. Note that this is independent of the number of constraints m.

Brunsch and Röglin [BR13] analyzed the shadow vertex simplex algorithm in terms of the condition number $\Delta_{\mathbf{A}}$, noting that for integer matrices $\delta_{\mathbf{A}} \geq 1/(n\Delta_{\mathbf{A}}^2)$. They gave a diameter bound $O(mn^2/\delta_{\mathbf{A}}^2)$. Eisenbrand and Vempala [EV17] used a different approach to derive a bound poly$(n, 1/\delta_{\mathbf{A}})$ that is independent of m. Dadush and Hähnle [DH16] further improved these bounds to $O(n^3 \log(n/\delta_{\mathbf{A}})/\delta_{\mathbf{A}})$.

In recent work, Dadush et al. [DVZ] considered (3.19) in the oracle model, where for each point $x \in \mathbb{R}^n$, the oracle returns $x \in P$ or a violated inequality $\langle a_i, x \rangle \leq b_i$ from the system $\mathbf{A}x \leq b$. Their algorithm finds exact primal and dual solutions using $O(n^2 \log(n/\delta_M))$ oracle calls, where

$$\mathbf{M} = \begin{bmatrix} \mathbf{0} & 1 \\ \mathbf{A} & b \end{bmatrix} . \tag{3.20}$$

The running time is independent of the cost function c. They also show the following relation between κ and δ:

Lemma 3.3.11 ([DVZ])**.**

(i) Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be a matrix with full row rank and $m < n$, with $\|a_i\| = 1$ for all columns $i \in [n]$. Then, $\kappa_{\mathbf{A}} \leq 1/\delta_{\mathbf{A}^\top}$.

(ii) Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be in basis form $\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & \mathbf{A}' \end{bmatrix}$. Then, $1/\delta_{\mathbf{A}^\top} \leq m\kappa_{\mathbf{A}}^2$.

(iii) If B is the basis maximizing $|\det(\mathbf{A}_B)|$, then for $\bar{\mathbf{A}} = \mathbf{A}_B^{-1}\mathbf{A}$, it holds that $1/\delta_{\bar{\mathbf{A}}^\top} \leq m\kappa_{\mathbf{A}}$.

Proof. **Part (i):** Let $g \in \mathcal{E}(A)$ be an elementary vector. Select an arbitrary $i \in \operatorname{supp}(g)$, and let $J = \operatorname{supp}(g) \setminus \{i\}$. Then, the columns $\{ g_i : i \in J \}$ are linearly independent, and $-g_i a_i = \sum_{j \in J} g_j a_j$. Thus,

$$|g_i| \cdot \|a_i\| = \left\| \sum_{j \in J} g_j a_j \right\| \geq \delta_{\mathbf{A}^\top} \max_{j \in J} |g_j| \cdot \|a_j\| ,$$

and using that all columns have unit norm, we get $|g_j/g_i| \leq 1/\delta_{\mathbf{A}^\top}$ for all $j \in J$. This shows that $\kappa_{\mathbf{A}} \leq 1/\delta_{\mathbf{A}^\top}$.

Parts (ii) and (iii): Let $\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & \mathbf{A}' \end{bmatrix}$ in basis form, and let $\alpha = \max_{i \in [n]} \|\mathbf{A}_i\|$. Let us first show

$$\frac{1}{\delta_{\mathbf{A}^\top}} \leq \sqrt{m} \alpha \kappa_{\mathbf{A}} . \tag{3.21}$$

Take any set $\{\mathbf{A}_i : i \in I\}$ of linearly independent columns of $\mathbf{A}$, along with coefficients $\lambda \in \mathbb{R}^I$. Without loss of generality, assume $|I| = m$, i.e., I is a basis, by allowing $\lambda_i = 0$ for some coefficients. Let $z = \sum_{i \in I} \lambda_i \mathbf{A}_i$. Then, $\lambda = \mathbf{A}_I^{-1} z$. Lemma 3.2.3 implies that every column of $\mathbf{A}_I^{-1}$ has 2-norm at most $\sqrt{m}\kappa_{\mathbf{A}}$. Hence, $|\lambda_i| \leq \sqrt{m}\kappa_{\mathbf{A}} \|z\|$ holds for all $i \in I$, implying (3.21).

Then, part (ii) follows since $\|\mathbf{A}_i\| \leq \sqrt{m}\kappa_{\mathbf{A}}$ by Proposition 3.2.1. For part (iii), let B be a basis maximizing $|\det(\mathbf{A}_B)|$. Then, $\|\mathbf{A}_B^{-1}\mathbf{A}\|_\infty \leq 1$. Indeed, if there is an entry $|\mathbf{A}_{ij}| > 1$, then we can obtain a larger determinant by exchanging i for j. This implies $\alpha \leq \sqrt{m}$. □

Using this correspondence between δ and κ, we can derive the following bound on the diameter of polyhedra in standard form from [DH16]. This verifies the polynomial Hirsch-conjecture whenever $\kappa_{\mathbf{A}}$ is polynomially bounded.

Theorem 3.3.12. *Consider a polyhedron in the standard equality form*

$$P = \{ x \in \mathbb{R}^n : \mathbf{A}x = b, x \geq \mathbf{0} \} \tag{3.22}$$

for $\mathbf{A} \in \mathbb{R}^{m \times n}$ and $b \in \mathbb{R}^m$. Then, the diameter of P is at most $O((n-m)^3 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$.

Proof. Without loss of generality, we can assume that $\mathbf{A}$ has full row rank. Changing to a standard basis representation does neither change the geometry (in particular, the diameter) of P, nor the value of $\kappa_{\mathbf{A}}$. Let B be the basis maximizing $\det(\mathbf{A}_B)$, and let us replace $\mathbf{A}$ by $\mathbf{A}_B^{-1}A$; w.l.o.g. assume that B is the set of the last m columns. Hence, $\mathbf{A} = \begin{bmatrix} \mathbf{A}' & \mathbf{I}_m \end{bmatrix}$ for $\mathbf{A}' \in \mathbb{R}^{m \times (n-m)}$. According to Lemma 3.2.6, P has the same diameter as P' defined as

$$P' = \{ x' \in \mathbb{R}^{n-m} : \mathbf{A}'x' \leq b, x' \geq \mathbf{0} \} , \tag{3.23}$$

in other words, $P' = \{ x' \in \mathbb{R}^{n-m} : \mathbf{C}x' \leq d \}$, where

$$\mathbf{C} = \begin{bmatrix} -\mathbf{I}_{n-m} \\ \mathbf{A}' \end{bmatrix} \quad \text{and} \quad d = \begin{bmatrix} 0 \\ b \end{bmatrix} . \tag{3.24}$$

There is a one-to-one correspondence between the vertices and edges of P and P', and hence, the two polyhedra have the same diameter. Thus, [DH16] gives a bound $O((n-m)^3 \log(n/\delta_{\mathbf{C}})/\delta_{\mathbf{C}})$ on the diameter of P'. By the choice of B, from Lemma 3.3.11(iii), we obtain the diameter bound $O((n-m)^3 m \kappa_{\mathbf{C}^\top} \log(\kappa_{\mathbf{C}^\top} + n))$. We claim that $\kappa_{\mathbf{C}^\top} = \kappa_{\mathbf{A}}$. Indeed, the kernels of $\mathbf{A} = \begin{bmatrix} \mathbf{A}' & \mathbf{I}_m \end{bmatrix}$ and

$\mathbf{C}^\top = \begin{bmatrix} -\mathbf{I}_{n-m} & (\mathbf{A}')^\top \end{bmatrix}$ represent orthogonal complements, thus $\kappa_{\mathbf{C}^\top} = \kappa_{\mathbf{A}}$ by Proposition 3.2.14. This completes the proof. □

The diameter bound in [DH16] is proved constructively, using the shadow simplex method. However, in the proof we choose B maximizing $|\det(\mathbf{A}_B)|$, a hard computational problem to solve even approximately [DEFM14]. However, we do not actually require a (near) maximizing subdeterminant. For the argument, we only need to find a basis $B \subseteq [n]$ such that for $\bar{\mathbf{A}} = \mathbf{A}_B^{-1}\mathbf{A}$, $\|\bar{\mathbf{A}}\|_\infty \leq \mu$ for some constant $\mu > 1$. Then, (3.21), gives $1/\delta_{\bar{A}^\top} \leq m\mu\kappa_{\mathbf{A}}$.

Such a basis B corresponds to approximate local subdeterminant maximization, and can be found using the following simple algorithm proposed by Knuth [Knu85]. As long as there is an entry $|\mathbf{A}_{ij}| > \mu$, then swapping i for j increases $|\det(\mathbf{A}_B)|$ by a factor $|\mathbf{A}_{ij}| > \mu$. Using that $|\det(\mathbf{A}_B)| \leq (\dot{\kappa}_W)^m$ by Proposition 3.2.19, the algorithm terminates in $O(m \log(\dot{\kappa}_W/\mu))$ iterations.

We also note that $\delta_{\mathbf{A}}$ was also studied for lattice basis reduction by Seysen [Sey93]. A related quantity has been used to characterize Hoffman constants (introduced in Section 3.5), see [GHR95; KT95; PVZ20].

3.4 Optimizing circuit imbalances

Recall that $\mathfrak{D}_n$ is the set of $n \times n$ positive definite diagonal matrices. For every $\mathbf{D} \in \mathfrak{D}_n$, $\mathbf{AD}$ represents a column rescaling. This is a natural symmetry in linear programming, and particularly relevant in the context of interior point algorithms.

The condition number $\kappa_{\mathbf{AD}}$ may vastly differ from $\kappa_{\mathbf{A}}$. In terms of the subspace $W = \ker(\mathbf{A})$, this amounts to rescaling the subspace by $\mathbf{D}^{-1}$; we denote this by $\mathbf{D}^{-1}W$. It is natural to ask for the best possible value that can be achieved by rescaling:

$$\kappa^*{}_W := \inf\{\, \kappa_{\mathbf{D}W} : \mathbf{D} \in \mathfrak{D}_n \,\}, \quad \text{and} \quad \bar{\chi}^*{}_W := \inf\{\, \kappa_{\mathbf{D}W} : \mathbf{D} \in \mathfrak{D}_n \,\}.$$

In most algorithmic and polyhedral results in this thesis, the κ_W dependence can be replaced by κ^*_W dependence. For example, the diameter bound in Theorem 3.3.12 is true in the stronger form with κ^*_W, since the diagonal rescaling maintains the geometry of the polyhedron.

A key result in [DHNV20] shows that an approximately optimal rescaling can be found. It will be stated and proven in Theorem 3.4.7 in this section. This is in surprising contrast with the inapproximability result Theorem 3.3.2. Note that there is no contradiction since the approximation factor $(\kappa^*_A)^2$ is not bounded as $2^{\text{poly}(n)}$ in general.

The key idea of the proof of Theorem 3.4.7 is to analyze the pairwise imbalances $\kappa_{ij} = \kappa^W_{ij}$ introduced in Section 3.2.3. In the 4-dimensional example Example 3.4.3, we have $\kappa_{34} = \kappa_{43} = M$. Let $\mathbf{D} \in \mathfrak{D}$ and let $d \in \mathbb{R}^n$ denote the diagonal elements; i.e., the rescaling multiplies the i-th coordinate of every $w \in W$ by d_i. Then, we can see that $\kappa^{\mathbf{D}W}_{ij} = \kappa_{ij} d_j/d_i$. In particular, for any pair of variables i and j, $\kappa^{\mathbf{D}W}_{ij}\kappa^{\mathbf{D}W}_{ji} = \kappa_{ij}\kappa_{ji}$. Consequently, we get a lower bound $\kappa_{ij}\kappa_{ji} \leq (\kappa^*_W)^2$.

Theorem 3.4.7 is based on a combinatorial min-max characterization that extends this idea. For the rest of this section, let us assume that the matroid $\mathcal{M}(W)$ is non-separable. In case it is separable, we can obtain κ^*_W by taking a maximum over the non-separable components.

Let $G = ([n], E)$ be the complete directed graph on n vertices with edge weights κ_{ij}. Since $\mathcal{M}(W)$ is assumed to be non-separable, Proposition 3.2.21 implies that $\kappa_{ij} > 0$ for any $i, j \in [n]$. We will refer to this weighted digraph as the *circuit ratio digraph*.

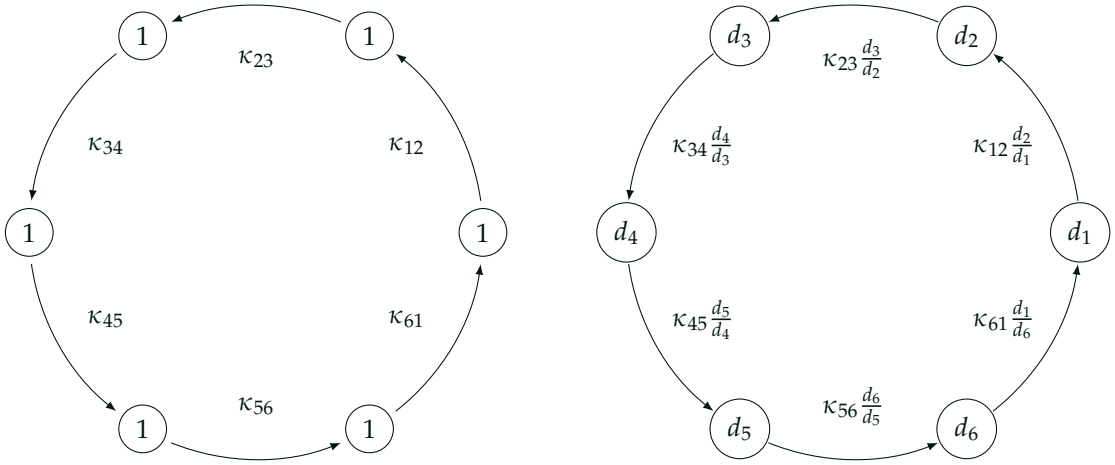


Figure 3.2: circuit imbalances κ in the circuit ratio digraph under rescaling $d > 0$ of variables

Let H be a *cycle* in G, that is, a sequence of indices $i_1, i_2, \ldots, i_k, i_{k+1} = i_1$. We use $|H| = k$ to denote the length of the cycle. (In this terminology, *cycles* refer to objects in G, whereas *circuits* to objects in C_W.)

We use the notation $\kappa(H) = \kappa_W(H) = \prod_{j=1}^{k} \kappa^W_{i_j i_{j+1}}$. The observation for length-2 cycles remains valid in general: $\kappa(H)$ is invariant under any rescaling. This leads to the lower bound $(\kappa(H))^{1/|H|} \leq \kappa^*_W$. The best of these bounds turns out to be tight:

A min-max theorem. We next provide a combinatorial min-max characterization of κ^*_W. Consider the circuit ratio digraph $G = ([n], E)$ on the node set $[n]$ where $(i, j) \in E$ if $\kappa_{ij} > 0$, that is, there exists a circuit $C \in \mathcal{C}$ with $i, j \in C$. We will refer to $\kappa_{ij} = \kappa^W_{ij}$ as the weight of the edge (i, j). (Note that $(i, j) \in E$ if and only if $(j, i) \in E$, but the weight of these two edges can be different.)

Let H be a *cycle* in G, that is, a sequence of indices $i_1, i_2, \ldots, i_k, i_{k+1} = i_1$. We use $|H| = k$ to denote the length of the cycle. (In our terminology, 'cycles' always refer to objects in G, whereas 'circuits' refer to the minimum supports in $\ker(\mathbf{A})$.)

We use the notation $\kappa(H) = \kappa_W(H) = \prod_{j=1}^{k} \kappa^W_{i_j i_{j+1}}$. For a vector $d \in \mathbb{R}^n_{++}$, we denote $\kappa^d_W(H) = \kappa_{\mathrm{diag}(d)W}(H)$. A simple but important observation is that such a rescaling does not change the value associated with the cycle, that is,

$$\kappa^d_W(H) = \kappa_W(H) \quad \forall d \in \mathbb{R}^n_{++} \quad \text{for any cycle } H \text{ in } G\,. \tag{3.25}$$

Theorem 3.4.1. *For a subspace $W \subset \mathbb{R}^n$, we have*

$$\kappa^*_W = \min_{d>0} \kappa^d_W = \max\Big\{ \kappa_W(H)^{1/|H|} : H \text{ is a cycle in } G \Big\}\,. \tag{3.26}$$

Proof. For the direction $\kappa_W(H)^{1/|H|} \leq \kappa^*_W$ we use (3.25). Let $d > 0$ be a scaling and H a cycle. We have $\kappa^d_{ij} \leq \kappa^d_W$ for every $i, j \in [n]$, and hence $\kappa_W(H) = \kappa^d_W(H) \leq (\kappa^d_W)^{|H|}$. Since this inequality holds for every $d > 0$, it follows that $\kappa_W(H) \leq (\kappa^*_W)^{|H|}$.

For the reverse direction, consider the following optimization problem.

$$\begin{aligned} &\min t \\ &\kappa_{ij} d_j / d_i \leq t \quad \forall (i,j) \in E \\ &d > 0. \end{aligned} \tag{3.27}$$

For any feasible solution (d, t) and $\lambda > 0$, we get another feasible solution $(\lambda d, t)$ with the same objective value. As such, we can strengthen the condition $d > 0$ to $d \geq 1$ without changing the objective value. This makes it clear that the optimum value is achieved by a feasible solution.

Any rescaling $d > 0$ provides a feasible solution with objective value κ^d, which means that the optimal value t^* of (3.27) is $t^* = \kappa^*$. Moreover, with the variable substitution $z_i = \log d_i$, $s = \log t$, (3.27) can be written as a linear program:

$$\begin{aligned} &\min s \\ &\log \kappa_{ij} + z_j - z_i \leq s \quad \forall (i,j) \in E \\ &z \in \mathbb{R}^n. \end{aligned} \tag{3.28}$$

This is the dual of a minimum-mean cycle problem with respect to the cost function $\log(\kappa_{ij})$. Therefore, an optimal solution corresponds to the cycle maximizing $\sum_{ij \in H} \log \kappa_{ij} / |H|$, or in other words, maximizing $\kappa(H)^{1/|H|}$. □

Whereas this formulation verifies Theorem 3.4.1, it does not give a polynomial-time algorithm to compute κ^*_W. The caveat is that the values κ^W_{ij} are typically not available; in fact, approximating them up to a factor $2^{O(m)}$ is NP-hard, as follows from the work of Tunçel [Tun99].

Nevertheless, the following corollary of Theorem 3.4.1 shows that any arbitrary circuit containing i and j yields a $(\kappa^*)^2$ approximation to κ_{ij}.

Corollary 3.4.2. *Let us be given a linear subspace $W \subseteq \mathbb{R}^n$ and $i, j \in [n]$, $i \neq j$, and a circuit $C \in \mathcal{C}_W$ with $i, j \in C$. Let $g \in W$ be the corresponding vector with $\operatorname{supp}(g) = C$. Then,*

$$\frac{\kappa^W_{ij}}{\left(\kappa^*_W\right)^2} \leq \frac{|g_j|}{|g_i|} \leq \kappa^W_{ij}.$$

Proof. The second inequality follows by definition. For the first inequality, note that the same circuit C yields $|g_i/g_j| \leq \kappa^W_{ji}(C) \leq \kappa^W_{ji}$. Therefore, $|g_j/g_i| \geq 1/\kappa^W_{ji}$.

From Theorem 3.4.1 we see that $\kappa^W_{ij}\kappa^W_{ji} \leq (\kappa^*_W)^2$, giving $1/\kappa^W_{ji} \geq \kappa^W_{ij}/(\kappa^*_W)^2$, completing the proof. □

In Section 3.4.1, we use techniques from matroid theory and linear algebra to efficiently identify a circuit for any pair of variables that are contained in the same circuit.

The following example shows that $\kappa^* \leq \bar{\chi}^*$ can be arbitrarily big.

Example 3.4.3. *Take $W = \operatorname{span}((0,1,1,M)^\top, (1,0,M,1)^\top)$, where $M > 0$. Then $\{2,3,4\}$ and $\{1,3,4\}$ are circuits with $\kappa^W_{34}(\{2,3,4\}) = M$ and $\kappa^W_{43}(\{1,3,4\}) = M$. Hence, by Theorem 3.4.1, we see that $\kappa^* \geq M$.*

3.4.1 Finding circuits: a detour in matroid theory

To prove Theorem 3.4.6, showing how to efficiently obtain a family $\hat{\mathcal{C}} \subseteq \mathcal{C}_W$ such that for any $i, j \in [n]$, $\hat{\mathcal{C}}$ includes a circuit containing both i and j, provided there exists such a circuit.

We give a different proof of (iii) ⇒ (iv) in Lemma 3.4.5 that will be convenient for our algorithmic purposes. First, we need a simple lemma that is commonly used in matroid optimization, see e.g. [Fra11, Lemma 13.1.11] or [Sch03, Theorem 39.13].

Lemma 3.4.4. *Let I be an independent set of a matroid $\mathcal{M} = ([n], \mathcal{I})$, and $U = \{u_1, u_2, \dots, u_\ell\} \subseteq I$, $V = \{v_1, v_2, \dots, v_\ell\} \subseteq [n] \setminus I$ such that $I \cup \{v_i\}$ is dependent for each $i \in [\ell]$. Further, assume that for each $t \in [\ell]$, $u_t \in C(I, v_t)$ and $u_t \notin C(I, v_h)$ for all $h < t$. Then, $(I \setminus U) \cup V \in \mathcal{I}$.*

We give a sketch of the proof. First, we note that for each $t \in [\ell]$, $u_t \in C(I, v_t)$ means that exchanging v_t for u_t maintains independence. The statement follows by induction on ℓ: we consider the independent set $I' = (I \setminus \{u_\ell\}) \cup \{v_\ell\}$. We can apply induction for I', $U' = \{u_1, u_2, \dots, u_{\ell-1}\}$, and $V' = \{v_1, v_2, \dots, v_{\ell-1}\}$, noting that the assumption guarantees that $C(I', v_t) = C(I, v_t)$ for all $t \in [\ell - 1]$. Based on this lemma, we show the following exchange property.

Lemma 3.4.5. *Let B be a basis of the matroid $\mathcal{M} = ([n], \mathcal{I})$, and let $U = \{u_1, u_2, \dots, u_\ell\} \subseteq B$, and $V = \{v_1, v_2, \dots, v_\ell, v_{\ell+1}\} \subseteq [n] \setminus B$. Assume $C(B, v_1) \cap U = \{u_1\}$, $C(B, v_{\ell+1}) \cap U = \{u_\ell\}$, and for each $2 \le t \le \ell$, $C(B, v_t) \cap U = \{u_{t-1}, u_t\}$. Then $(B \setminus U) \cup V$ contains a unique circuit C, and $V \subseteq C$.*

The situation described here corresponds to a minimal path in the hypergraph $\mathcal{C}^B$ of the fundamental circuits with respect to a basis B. The hyperedges $C(B, v_i)$ form a path from v_1 to $v_{\ell+1}$ such that no shortcut is possible (note that this is weaker than requiring a shortest path).

Proof of Lemma 3.4.5. Note that $S = (B \setminus U) \cup V \notin \mathcal{I}$ since $|S| > |B|$ and B is a basis. For any $i \in [\ell + 1]$, we can use Lemma 3.4.4 to show that $S \setminus \{v_i\} = (B \setminus U) \cup (V \setminus \{v_i\}) \in \mathcal{I}$ (and thus, is a basis). To see this, we apply Lemma 3.4.4 for the ordered sets $V' = \{v_1, \dots, v_{i-1}, v_{\ell+1}, v_\ell, \dots, v_{i+1}\}$ and $U' = \{u_1, \dots, u_{i-1}, u_\ell, u_{\ell-1}, \dots, u_i\}$.

Consequently, every circuit in S must contain the entire set V. The uniqueness of the circuit in S follows by the well-known circuit axiom asserting that if $C, C' \in \mathcal{C}$, $C \neq C'$ and $v \in C \cap C'$, then there exists a circuit $C'' \in \mathcal{C}$ such that $C'' \subseteq (C \cup C') \setminus \{v\}$, contradicting the claim that every circuit in S contains the entire set V. □

We are ready to describe the algorithm that will be used to obtain lower bounds on all κ_{ij} values.

Theorem 3.4.6. *Given $\mathbf{A} \in \mathbb{R}^{m \times n}$, there exists an $O(n^2 m^2)$ time algorithm FIND-CIRCUITS($\mathbf{A}$) that obtains a decomposition of $\mathcal{M}(\mathbf{A})$ to a direct sum of non-separable linear matroids, and returns a family $\hat{\mathcal{C}}$ of circuits such that if i and j are in the same non-separable component, then there exists a circuit in $\hat{\mathcal{C}}$ containing both i and j. Further, for each $i \neq j$ in the same component, the algorithm returns a value $\hat{\kappa}_{ij}$ as the the maximum of $|g_j/g_i|$ such that $g \in W$, $\operatorname{supp}(g) = C$ for some $C \in \hat{\mathcal{C}}$ containing i and j. For these values, $\hat{\kappa}_{ij} \le \kappa_{ij} \le (\kappa^*)^2 \hat{\kappa}_{ij}$.*

Proof. Once we have found the set of circuits $\hat{\mathcal{C}}$, and computed $\hat{\kappa}_{ij}$ as in the statement, the inequalities $\hat{\kappa}_{ij} \le \kappa_{ij} \le (\kappa^*)^2 \hat{\kappa}_{ij}$ follow easily. The first inequality is by the definition of κ_{ij}, and the second inequality is from Corollary 3.4.2.

We now turn to the computation of $\hat{\mathcal{C}}$. We first obtain a basis $B \subseteq [n]$ of $\ker(\mathbf{A})$ via Gauss-Jordan elimination in time $O(nm^2)$. Recall the assumption that A has full row-rank. Let us assume that $B = [m]$ is the set of first m indices. The elimination transforms it to the basis form $\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & \mathbf{H} \end{bmatrix}$, where $\mathbf{H} \in \mathbb{R}^{m \times (n-m)}$ corresponds to the non-basis elements. In this form, the fundamental circuit $C(B, i)$ is the support of the i-th column of $\mathbf{A}$ together with i for every $m + 1 \le i \le n$. We let $\mathcal{C}^B$ denote the set of all these fundamental circuits.

We construct an undirected graph $G = (B, E)$ as follows. For each $i \in [n] \setminus B$, we add a clique between the nodes in $C(B, i) \setminus \{i\}$. This graph can be constructed in $O(nm^2)$ time.

The connected components of G correspond to the connected components of C^B restricted to B. Thus, due to the equivalence shown in Proposition 3.2.21 we can obtain the decomposition by identifying the connected components of G. For the rest of the proof, we assume that the entire hypergraph is connected; connectivity can be checked in $O(m^2)$ time.

We initialize $\hat{C}$ as C^B. We will then check all pairs $i, j \in [n]$, $i \neq j$. If no circuit $C \in \hat{C}$ exists with $i, j \in C$, then we will add such a circuit to $\hat{C}$ as follows.

Assume first $i, j \in [n] \setminus B$. We can find a shortest path in G between the sets $C(B, i) \setminus \{i\}$ and $C(B, j) \setminus \{j\}$ in time $O(m^2)$. This can be represented by the sequences of points $V = \{v_1, v_2, \ldots, v_{\ell+1}\} \subseteq [n] \setminus B$, $v_1 = i$, $v_{\ell+1} = j$, and $U = \{u_1, u_2, \ldots, u_\ell\} \subseteq B$ as in Lemma 3.4.5. According to the lemma, $S = (B \setminus U) \cup V$ contains a unique circuit C that contains all v_t's, including i and j.

We now show how this circuit can be identified in $O(m)$ time, along with the vector g^C. Let $\mathbf{A}_S$ be the submatrix corresponding to the columns in S. Since $g = g^C$ is unique up to scaling, we can set $g_{v_1} = 1$. Note that for each $t \in [\ell]$, the row of $\mathbf{A}_S$ corresponding to u_t contains only two nonzero entries: $\mathbf{A}_{u_t v_t}$ and $\mathbf{A}_{u_t v_{t+1}}$. Thus, the value $g_{v_1} = 1$ can be propagated to assigning unique values to $g_{v_2}, g_{v_3}, \ldots, g_{v_{\ell+1}}$. Once these values are set, there is a unique extension of g to the indices $t \in B \cap S$ in the basis. Thus, we have identified g as the unique element of $\ker(\mathbf{A}_S)$ up to scaling. The circuit C is obtained as $\operatorname{supp}(g)$. The above procedure can be implemented in $O(m)$ time.

The argument easily extends to finding circuits for the case $\{i, j\} \cap B \neq \emptyset$. If $i \in B$, then for any choice of $V = \{v_1, v_2, \ldots, v_{\ell+1}\}$ and $U = \{u_1, u_2, \ldots, u_\ell\}$ as in Lemma 3.4.5 such that $i \in C(B, v_1)$ and $i \notin C(B, v_t)$ for $t > 1$, the unique circuit in $(B \setminus U) \cup V$ also contains i. This follows from Lemma 3.4.4 by taking $V' = \{v_{\ell+1}, v_\ell, \ldots, v_1\}$ and $U' = \{u_\ell, \ldots, u_1, i\}$, which proves that $S \setminus \{i\} = (B \setminus U') \cup V' \in \mathcal{I}$. Similarly, if $j \in B$ with $j \in C(B, v_{\ell+1})$ and $j \notin C(B, v_t)$ for $t < \ell + 1$, taking $V'' = V$ and $U'' = \{u_1, \ldots, u_\ell, j\}$ gives $S \setminus \{j\} \in \mathcal{I}$.

The bottleneck for the running time is finding the shortest paths for the $n(n-1)$ pairs, in time $O(m^2)$ each. □

Theorem 3.4.7. *There is an $O(n^2m^2 + n^3)$ time algorithm that for any matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$ computes an estimate ξ of $\bar{\chi}_W$ such that*

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

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

$$\bar{\chi}^*{}_W \leq \bar{\chi}_{\mathbf{D}W} \leq n(\bar{\chi}^*_W)^3 .$$

Proof. Let us run the algorithm Finding-Circuits(A) described in Theorem 3.4.6 to obtain the values $\hat{\kappa}_{ij}$ such that $\hat{\kappa}_{ij} \leq \kappa_{ij} \leq (\kappa^*_W)^2 \hat{\kappa}_{ij}$. We let $G = ([n], E)$ be the circuit ratio digraph, that is, $(i, j) \in E$ if $\kappa_{ij} > 0$.

To show the first statement on approximating $\bar{\chi}$, we simply set $\xi = \max_{(i,j) \in E} \hat{\kappa}_{ij}$. Then,

$$\xi \leq \kappa_W \leq \bar{\chi}_W \leq n\kappa_W \leq n(\kappa^*_W)^2 \xi \leq n(\bar{\chi}^*{}_W)^2 \xi$$

follows by Theorem 3.3.8.

For the second statement on finding a nearly optimal rescaling for $\bar{\chi}^*{}_W$, we consider the following optimization problem, which is an approximate version of (3.27) from Theorem 3.4.1.

$$\begin{aligned} &\min t \\ &\hat{\kappa}_{ij} d_j / d_i \leq t \quad \forall (i, j) \in E \\ &d > 0. \end{aligned} \tag{3.29}$$

Let $\hat{d}$ be an optimal solution to (3.29) with value $\hat{t}$. We will prove that $\kappa^{\hat{d}} \leq (\kappa^*_W)^3$.

First, observe that $\kappa^{\hat{d}}_{ij} = \kappa_{ij}\hat{d}_j/\hat{d}_i \leq (\kappa^*_W)^2\hat{\kappa}_{ij}\hat{d}_j/\hat{d}_i \leq (\kappa^*_W)^2\hat{t}$ for any $(i,j) \in E$. Now, let $d^* > 0$ be such that $\kappa^{d^*} = \kappa^*_W$. The vector d^* is a feasible solution to (3.29), and so $\hat{t} \leq \max_{i\neq j} \hat{\kappa}_{ij}d^*_j/d^*_i \leq \max_{i\neq j} \kappa_{ij}d^*_j/d^*_i = \kappa^{d^*}$. Hence we find that $\hat{d}$ gives a rescaling with

$$\bar{\chi}_{W\widehat{D}} \leq n\kappa^{\hat{d}} \leq n(\kappa^*_W)^3 \leq n(\bar{\chi}_W)^3\,,$$

where we again used Theorem 3.3.8.

We can obtain the optimal value $\hat{t}$ of (3.29) by solving the corresponding maximum-mean cycle problem (see Theorem 3.4.1). It is easy to develop a multiplicative version of the standard dynamic programming algorithm of the classical minimum-mean cycle problem (see e.g. [AMO93, Theorem 5.8]) that allows finding the optimum to (3.29) directly, in the same $O(n^3)$ time.

It is left to find the labels $d_i > 0$, $i \in [n]$ such that $\hat{\kappa}_{ij}d_j/d_i \leq \hat{t}$ for all $(i,j) \in E$. We define the following weighted directed graph. We associate the weight $w_{ij} = \log \hat{t} - \log \hat{\kappa}_{ij}$ with every $(i,j) \in E$, and add an extra source vertex r with edges (r,i) of weight $w_{ri} = 0$ for all $i \in [n]$.

By the choice of $\hat{t}$, this graph does not contain any negative weight directed cycles. We can compute the shortest paths from r to all nodes in $O(n^3)$ using the Bellman-Ford algorithm; let σ_i be the shortest path label for i. We then set $d_i = \exp(\sigma_i)$. One can avoid computing logarithms by using a multiplicative variant of the Bellman-Ford algorithm instead.

The running time of the whole algorithm will be bounded by $O(n^2m^2 + n^3)$. The running time is dominated by the $O(n^2m^2)$ complexity of Finding-Circuits(A) and the $O(n^3)$ complexity of solving the minimum-mean cycle problem and shortest path computation. □

3.4.2 Perfect balancing: $\kappa^*_W = 1$

Let us now show that $\kappa^*_{\mathbf{A}} = 1$ can be efficiently checked.

Theorem 3.4.8. *There exists a strongly polynomial algorithm, that given a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, returns one of the following outcomes:*

(a) A diagonal matrix $\mathbf{D} \in \mathfrak{D}_n$ such that $\kappa_{\mathbf{AD}} = 1$ showing that $\kappa^*_{\mathbf{A}} = 1$. The algorithm also returns the exact value of $\kappa_{\mathbf{A}}$. Further, if $\ker(\mathbf{A})$ is a rational linear space, then we can select $\mathbf{D}$ with integer diagonal entries that divide $\dot{\kappa}_{\mathbf{A}}$.

(b) The answer $\kappa^*_{\mathbf{A}} > 1$, along with a cycle of circuits H such that $\kappa_{\mathbf{A}}(H) > 1$.

Proof. As noted above, we can assume without loss of generality that the matroid $\mathcal{M}(W)$ is non-separable, as we can reduce the problem to solving on all connected components separately.

We obtain estimates $\hat{\kappa}_{ij}$ for every edge (i,j) of the circuit ratio graph using a circuit $C \in \mathcal{C}_W$ with $i,j \in C$. Assuming that $\kappa^*_W = 1$, Corollary 3.4.2 implies that $\hat{\kappa}_{ij} = \kappa_{ij}$ holds and the rescaling factors d_i must satisfy

$$\hat{\kappa}_{ij}d_j = d_i \quad \forall i,j \in [n]\,. \tag{3.30}$$

If this system is infeasible, then using the circuits that provided the estimates $\hat{\kappa}_{ij}$, we can obtain a cycle H such that $\kappa_{\mathbf{A}}(H) > 1$, that is, outcome (b). Let us now assume that (3.30) is feasible; then it has a unique solution d up to scalar multiplication. We define $\mathbf{D} \in \mathfrak{D}_n$ with diagonal entries $\mathbf{D}_{ii} = d_i$.

Since $\mathcal{M}(W)$ is non-separable, we can conclude that $\kappa^*_{\mathbf{A}} = 1$ if and only if $\kappa_{\mathbf{AD}} = 1$. By Theorem 3.2.4, this holds if and only if $\mathbf{A}' = \mathbf{A}_B^{-1}\mathbf{AD}$ is a TU-matrix for any basis B.

We run Seymour's algorithm [Sey80] for $\mathbf{A}'$. If it confirms that $\mathbf{A}$ is TU (certified by a construction sequence), then we return outcome (a). In this case, $|g_j^C / g_i^C|$ is the same for any circuit C with $i, j \in C$; therefore $\kappa_{ij} = \hat{\kappa}_{ij}$, and we can return $\kappa_{\mathbf{A}} = \max_{(i,j)\in E} \hat{\kappa}_{ij}$.

Otherwise, Seymour's algorithm finds a $k \times k$ submatrix $\mathbf{T}$ of $\mathbf{A}'$ with $\det(\mathbf{T}) \notin \{0, \pm 1\}$. As in the proof of Proposition 3.2.2, we can recover a circuit C in $C_{\mathbf{A}'} = C_{\mathbf{AD}}$ with two entries $i, j \in C$ such that $|\bar{g}_j| \neq |\bar{g}_i|$ for the corresponding elementary vector $\bar{g} \in \mathcal{E}(\mathbf{AD})$. Note that $\hat{\kappa}_{ij} d_j = d_i$ for the rescaled estimates. Hence, the circuit C' with $i, j \in C'$ used to obtain the estimate κ_{ij}, together with C certifies that $\kappa^*_{\mathbf{A}} > 1$ as required for outcome (b).

Finally, if $\ker(\mathbf{A})$ is a rational linear space and we concluded $\kappa^*_{\mathbf{A}} = 1$, then let us select the solution d_i to (3.30) such that $d \in \mathbb{Z}^n$ and $\gcd(d) = 1$. We claim that $d_i \mid \dot{\kappa}_W$ for all i. Indeed, let $k = \operatorname{lcm}(d)$. For each pair $i, j \in [n]$, $d_j/d_i = r/q$ for two integers $r, q \mid \dot{\kappa}_{\mathbf{A}}$. Hence, for any prime $p \in \mathbb{P}$, $\nu_p(k) \leq \nu_p(\dot{\kappa}_{\mathbf{A}})$, implying $k \mid \dot{\kappa}_{\mathbf{A}}$. □

Let $W \subseteq \mathbb{R}^n$ be a linear space such that $\mathcal{M}(W)$ is non-separable. Recall from Theorem 3.2.22 that $\mathcal{K}^W_{ij} \subseteq \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$ for all $i, j, k \in [n]$. We now characterize when equality holds for all triples.

Proposition 3.4.9. *Let $W \subseteq \mathbb{R}^n$ be a linear space such that $\mathcal{M}(W)$ is non-separable. Then, the following are equivalent:*

(i) $\kappa^*_W = 1$,

(ii) $|\mathcal{K}^W_{ij}| = 1$ *for all* $i, j \in [n]$,

(iii) $\mathcal{K}^W_{ij} = \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$ *holds for all distinct* $i, j, k \in [n]$.

Proof. (i) ⇔ (ii): Consider any rescaling $\mathbf{D} \in \mathfrak{D}_n$ with diagonal entries $d_i = \mathbf{D}_{ii}$. Then, $\mathcal{K}^{DW}_{ij} = \{1\}$, and $\mathcal{K}^{DW}_{ij} = \frac{d_j}{d_i} \mathcal{K}^W_{ij}$. Hence, if $\kappa_{DW} = 1$ for some $D \in \mathfrak{D}_n$, then $\mathcal{K}^{DW}_{ij} = \{1\}$ implying $|\mathcal{K}^W_{ij}| = 1$ for every $i, j \in [n]$. If $|\mathcal{K}^W_{ij}| > 1$ for some i, j, it follows that $\kappa_{DW} \neq 1$ for any diagonal rescaling.
(ii) ⇒ (iii): We have $\mathcal{K}^W_{ij} \subseteq \mathcal{K}^W_{ik} \cdot \mathcal{K}^W_{kj}$ by Theorem 3.2.22. If all three sets are of size one, then equality must hold.
(iii) ⇒ (i): Let $i, j \in [n]$ arbitrary but distinct and let us define

$$\Gamma_{ij} := \{\, \kappa(H) \mid H \text{ closed walk in } G, (i,j) \in E(H) \,\}.$$

Note that either $\Gamma_{ij} = \{1\}$ or Γ_{ij} is infinite as any cycle H can be traversed multiple times to form a closed walk. Note that by (iii) we have for any $i, j \in [n]$ that

$$\Gamma_{ij} \subseteq \bigcup \Big\{ \Pi_{(k,\ell)\in E(H)} \mathcal{K}_{k,\ell} \;\Big|\; H \text{ closed walk in } G, (i,j) \in E(H) \Big\} = \mathcal{K}_{ij} \cdot \mathcal{K}_{ji}. \tag{3.31}$$

The set $\mathcal{K}_{ij} \cdot \mathcal{K}_{ji}$ is finite, implying that $\Gamma_{ij} = \{1\}$. This, together with Theorem 3.4.1 gives (i). □

A surprising finding by Lee [Lee89; Lee90] is that if $\dot{\kappa}_W$ is an odd prime power, then $\kappa^*_W = 1$ holds.[1] We first present a proof sketch following the lines of the one in [Lee89; Lee90]. We also present a second, almost self-contained proof, relying only on basic results on TU matrices.

Theorem 3.4.10 (Lee [Lee89; Lee90])**.** *Each W for which $\dot{\kappa}_W = p^\alpha$ where $p \in \mathbb{P}$, $p > 2$, $\alpha \in \mathbb{N}$, then $\kappa^*_W = 1$.*

Proof. A theorem by Tutte [Tut65] asserts that W can be represented as the kernel of an unimodular matrix, i.e. $\kappa^*_W = 1$ or W has a minor W' such that $C(W') \cong C(U^4_2)$ where U^4_2 is the uniform matroid

[1] The statement in the paper is slightly more general, for k-adic subspaces with $k > 2$; the proof is essentially the same.

on four elements such that the independent sets are the sets of cardinality at most two. Here, a matroid minor corresponds to iteratively either deleting variables or projecting variables out. In the first case we are done, so let us consider the second case. Note that $W' \subset \mathbb{R}^4$ and by Lemma 3.2.16 we have that for all $i, j \in [4]$ we have that $\mathcal{K}_{ij}^{W'} \subseteq \mathcal{K}_{ij}^{W}$ and so in particular $\dot{\kappa}_{W'} = p^\beta$ for some $\beta \leq \alpha$. An easy consequence of the proof of Proposition 3.2.18 and the congruence $C(W') \cong C(U_2^4)$ is that W' can be represented by $\mathbf{A}'$, i.e., $\ker(\mathbf{A}') = W'$ such that

$$\mathbf{A}' = \begin{bmatrix} 1 & 0 & p^{\gamma_1} & p^{\gamma_2} \\ 0 & 1 & p^{\gamma_3} & p^{\gamma_4} \end{bmatrix} \tag{3.32}$$

for $\gamma_i \in \mathbb{N} \cup \{0\}$ and $i \in [4]$. Further, by $C(W') \cong C(U_2^4)$ and $\Delta_{\mathbf{A}'} \mid \dot{\kappa}_{W'}$ (Proposition 3.2.18) we have that

$$0 \neq \det \begin{bmatrix} p^{\gamma_1} & p^{\gamma_2} \\ p^{\gamma_3} & p^{\gamma_4} \end{bmatrix} = p^{\gamma_1+\gamma_4} - p^{\gamma_2+\gamma_3} \mid p^\beta. \tag{3.33}$$

It is immediate that (3.33) cannot be fulfilled for $p > 2$. □

Alternative Proof of Theorem 3.4.10. Let $\mathbf{A} \in \mathbb{R}^{m \times n}$ be such that $\mathbf{A} = \ker(W)$ satisfying the properties in Proposition 3.2.18 in basis form $\mathbf{A} = \begin{bmatrix} \mathbf{I}_m & \mathbf{A}' \end{bmatrix}$; for simplicity, assume the identity matrix is in the first m columns. Let $G = \big([n], E(G)\big)$ be a directed multigraph associated with $\mathbf{A}$ with edge set $E(G) = \bigcup_{k \in [m]} E_k(G)$ where $E_k(G) = \big\{ (i,j) : \mathbf{A}_{ki}\mathbf{A}_{kj} \neq 0 \big\}$. Further, define $\gamma \colon E(G) \to \mathbb{R}_+$ where for $e \in E_k$ we let $\gamma(e) = |\mathbf{A}_{kj}/\mathbf{A}_{ki}|$. For a directed cycle C in G we define $\gamma(C) \coloneqq \prod_{e \in E(C)} \gamma(e)$.

Claim 3.4.10.1. *All cycles C in G fulfill $\gamma(C) = 1$.*

Proof. For a contradiction, assume that there exists a cycle C such that $\gamma(C) \neq 1$ and let C be a shortest cycle with this property. Then C has no chord $f \in E(G)$, as otherwise $C \cup \{f\}$ contains two shorter cycles C_1, C_2 such that $\gamma(C_1)\gamma(C_2) = \gamma(C) \neq 1$ and so in particular $\gamma(C_1) \neq 1$ or $\gamma(C_2) \neq 1$. This also means that the support of the corresponding submatrix $\mathbf{A}_{I,J}$ of $\mathbf{A}$ where $I \coloneqq \{ i \in [m] : E_i(G) \cap E(C) \neq 0 \}$ and $J \coloneqq V(C)$ is exactly the set of non-zeros of an incidence matrix of a cycle. We have that $\det(\mathbf{A}_{I,J}) \neq 0$ as the corresponding cycle C has $\gamma(C) \neq 1$. Recall the Leibniz determinant formula. As $\mathbf{A}_{I,J}$ is supported on the incidence matrix of a cycle there exist only two bijective maps $\phi, \psi : I \to J$, $\phi \neq \psi$ such that $\prod_{i \in I} \mathbf{A}_{i,\phi(i)} \neq 0 \neq \prod_{i \in I} \mathbf{A}_{i,\psi(i)}$ is non-vanishing. One of the maps corresponds to traversing the cycle forward, the other corresponds to traversing it backwards. As all the entries of A are powers of p we therefore have that $0 \neq \det(\mathbf{A}_{I,J}) = \pm p^\alpha \pm p^\beta$ for some $\alpha, \beta \in \mathbb{N}$. This contradicts Proposition 3.2.18(iii) for $p > 2$. □

The above claim implies the existence of a rescaling of rows and columns $\tilde{\mathbf{A}} \coloneqq \mathbf{LAR}$ where $\mathbf{L} \in \mathfrak{D}_n$, $\mathbf{R} \in \mathfrak{D}_m$ such that $\tilde{\mathbf{A}} \in \{-1, 0, 1\}^{m \times n}$. If $\tilde{\mathbf{A}}$ is TU, then we are done by Proposition 3.2.2 as now $\kappa_W^* = 1$. Otherwise, we use a result by Gomory (see [Cam65] and [Sch98, Theorem 19.3]) that states that any matrix $\mathbf{B}$ with entries in $\{-1, 0, 1\}$ that is not TU has a submatrix $\mathbf{B}'$ with $|\det(\mathbf{B}')| = 2$. Let $I \subseteq [m]$ and $J \subseteq [n]$ such that $|\det(\tilde{\mathbf{A}}_{I,J})| = 2$. Note that w.l.o.g. the diagonal entries of $\mathbf{L}$ and $\mathbf{R}$ are of the form p^α for some $\alpha \in \mathbb{Z}$. Therefore, $|\det(\mathbf{A}_{I,J})| = \prod_{i \in I} \mathbf{L}_{ii} \prod_{j \in J} \mathbf{R}_{jj} |\det(\tilde{\mathbf{A}}_{I,J})| = 2p^\beta$ for some $\beta \in \mathbb{Z}$. As $|\det(\mathbf{A}_{I,J})| \in \mathbb{N}$ we must have $\beta \geq 0$ and $2 \mid |\det(\mathbf{A}_{I,J})|$. This again contradicts Proposition 3.2.18(iii) for $p > 2$. □

3.5 Proximity via Hoffman-bounds

Hoffman's seminal work [Hof52] has analyzed proximity of LP solutions. Given

$$P = \{ x \in \mathbb{R}^n : \mathbf{A}x \leq b \}, \tag{3.34}$$

$x_0 \in \mathbb{R}^n$, 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 $H_{\alpha,\beta}(\mathbf{A})\|(\mathbf{A}x_0 - b)^+\|_\beta$, where the Lipschitz-bound $H_{\alpha,\beta}(\mathbf{A})$ is a constant that only depends on $\mathbf{A}$ and the norms. Such bounds have been shown for 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 $\bar{\chi}_{\mathbf{A}}$ 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

$$\Lambda(d, c) := \operatorname{supp}(d^-) \cup \operatorname{supp}(c^+). \tag{3.35}$$

Theorem 3.5.1 (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. We have $\operatorname{supp}(u^-) \cap \operatorname{supp}(\ell^+) = \emptyset$. 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 2.0.3. 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 3.3.7. □

We can derive useful corollaries for feasibility and optimization problems.

Corollary 3.5.2. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $d \in \mathbb{R}^n$. If the primal Primal(W, d) is feasible, then the system*

$$\begin{aligned} x &\in W + d \\ \|x - d\|_\infty &\leq \kappa_W \|d^-\|_1 \\ x &\geq \mathbf{0}, \end{aligned}$$

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 3.5.1 guarantees a solution with $\|z\|_\infty \leq \kappa_W \|d^-\|_1$, as required. □

Corollary 3.5.3. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d \in \mathbb{R}^n$, and let $c \geq \mathbf{0}$. If Primal(W, d) is feasible, then there is an optimal primal (x, s) to LP(W, d, c) such that*

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

Proof. Let (x^*, s^*) be optimal to LP(W, d, c). That is, x^* minimizes $\langle c, x\rangle$ over $x \in W + d, x \geq \mathbf{0}$. Consider the feasibility system $x \in W + d, x \geq \mathbf{0}$, 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 3.5.1. □

We can immediately use this theorem to derive a conclusion on the support of the optimal dual solutions to LP(W, d, c).

Corollary 3.5.4. *Let (W, d, c) be a feasible instance of LP with $c \geq \mathbf{0}$ and let*

$$R := \left\{ i \in [n] : d_i > \kappa_W \|d_{\Lambda(d,c)}\|_1 \right\} . \tag{3.36}$$

Then for every optimal dual solution s^* to LP(W, d, c), we have $s^*_R = \mathbf{0}$.

Proof. By Corollary 3.5.3, there exists an optimal primal solution x^* such that for all $r \in R$ we have

$$x^*_r \geq d_r - \|d - x^*\|_\infty > 0 . \tag{3.37}$$

By complementary slackness we must have s^*_r for all optimal dual solutions s^* of LP(W, d, c). □

The next lemma can also be used to conclude that a primal variable $s_i^* = 0$ in every solution (x^*, s^*) to 1.2. 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 $\Delta_{\mathbf{A}}$. 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 $\bar{\chi}_{\mathbf{A}}$. 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 $\Delta_{\mathbf{A}}$. Hence, our formulation with κ_W can be seen as a natural extension.

Lemma 3.5.5. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d \in \mathbb{R}^n$. Let $(\tilde{x}, s)$ be an optimal solution to LP$(W, \tilde{d}, c)$. Then there exists an optimal solution (x^*, s^*) to LP(W, d, c) such that*

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

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

We apply Corollary 3.5.3 to (W, x, s). This guarantees the existence of an optimal (x^*, s^*) to LP(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 $x_{\operatorname{supp}(s^+)} = x_{\operatorname{supp}(s^+)} - \tilde{x}_{\operatorname{supp}(s^+)}$. These together imply that

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

□

We can immediately use Lemma 3.5.5 to derive a conclusion on the support of the optimal dual solutions to System 1.2 with data (W, d, c) using the optimal solution with data $(W, \tilde{d}, c)$.

Theorem 3.5.6. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, d \in \mathbb{R}^n$. Let $(\tilde{x}, s)$ be an optimal solution to LP$(W, \tilde{x}, c)$ and*

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

Then for every dual optimal solution s^* to System 1.2, we have $s^*_R = \mathbf{0}$.

Proof. By Lemma 3.5.5 there exists an optimal solution (x', s') to LP(W, d, c) such that $\|x' - \tilde{x}\|_\infty \leq (\kappa_W + 1)\|d - \tilde{x}\|_1$. Consequently, $x'_R > 0$, implying $s^*_R = \mathbf{0}$ for every dual optimal s^* by complementary slackness. □

Proximity for Capacitated LP. In Section 8.7, we use a dual version of this theorem, also including upper bound constraints in the primal side. We now adapt the required proximity result to the following primal and dual LPs, and formulate it in matrix language to conform to the algorithm in Section 8.7.

Note that any $y \in \mathbb{R}^m$ induces a feasible dual solution with $s_i = (c_i - \langle a_i, y\rangle)^+$ and $t_i = (\langle a_i, y\rangle - c_i)^+$ for $i \in [n]$. A primal feasible solution x and $y \in \mathbb{R}^m$ are optimal solutions if and only if $\langle a_i, y\rangle \leq c_i$ if $x_i < u_i$ and $\langle a_i, y\rangle \geq c_i$ if $x_i > 0$.

Theorem 3.5.7. *Let (x', y') be optimal primal and dual solutions to System 1.3 for input (b, u, c'), and (x'', y'') for input (b, u, c''). Let*

$$R_0 := \left\{ i \in [n] : \langle a_i, y'\rangle < c'_i - (\kappa_W + 1)\|c' - c''\|_1 \right\},$$
$$R_u := \left\{ i \in [n] : \langle a_i, y'\rangle > c'_i + (\kappa_W + 1)\|c' - c''\|_1 \right\}.$$

Then $x''_i = 0$ for every $i \in R_0$ and $x''_i = u_i$ for every $i \in R_u$.

Proof. Let

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{I}_n & \mathbf{I}_n \end{bmatrix}. \tag{3.38}$$

It is easy to see that $\kappa_{\bar{A}} = \kappa_{\mathbf{A}}$. Let $\bar{d} \in \mathbb{R}^{2n}$ such that $\bar{\mathbf{A}}\bar{d} = \begin{bmatrix} b \\ u \end{bmatrix}$. With $\bar{c} = (c, \mathbf{0}_n)$, the primal system can be equivalently written as $\min\langle \bar{c}, \bar{x}\rangle$, $\bar{x} \in \ker(\bar{\mathbf{A}}) + \bar{d}$, $\bar{x} \geq \mathbf{0}$. The statement follows by Theorem 3.5.6 applied for $W = (\ker(\bar{\mathbf{A}}))^\perp = \mathrm{im}(\bar{\mathbf{A}}^\top)$. □

Subspace and scaling proximity bounds

The reader will have noted that we could have chosen d and c arbitraily in $W + d$ resp. $W^\perp + c$ as such shift in subspace does not change the optimality of solutions. Scaling is another operation under which optimal solutions are invariant. In particular, the set of primal optimal solutions $\mathcal{X}^*(W, d, c)$ to LP(W, d, c) is the same as $\mathcal{X}^*(W, d, \tilde{c})$ for all $\tilde{c} \in \{ \alpha z : z \in W^\perp + c, \alpha > 0 \}$. To deduce a further proximity result which projects out this scaling by α we introduce following notation.

$$\sphericalangle : \mathbb{R}^n \times \mathbb{R}^n \to [0, 2\pi),\ (v, w) \mapsto \arccos\left(\frac{\langle v, w\rangle}{\|v\|\|w\|}\right). \tag{3.39}$$

In abuse of notation, we also writ efor subsets $U, V \subseteq \mathbb{R}^n$

$$\sphericalangle(U, V) := \max\left\{ \sphericalangle(u, v) : u \in U \setminus \{0\}, v \in V \setminus \{0\} \right\}.$$

Lemma 3.5.8. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, \tilde{c}, d \in \mathbb{R}^n$ such that $\tilde{c} \notin W^\perp$ and $\|c\| = \|\tilde{c}\| = 1$. Let $(x, \tilde{s})$ be an optimal solution to LP$(W, d, \tilde{c})$. Then there exists an optimal solution (x^*, s^*) to LP(W, d, c) such that*

$$\sin\left(\sphericalangle(s^*, \tilde{s})\right) \leq \frac{n(\kappa_W + 1)\sphericalangle(c, \tilde{c})}{\cos(\sphericalangle(\tilde{c}, W))}. \tag{3.40}$$

Proof. By law of cosine

$$\|c - \tilde{c}\| = \sqrt{\|c\|^2 + \|\tilde{c}\|^2 - 2\|c\|\|\tilde{c}\|\cos\left(\sphericalangle(c, \tilde{c})\right)} = 2\sin\left(\frac{\sphericalangle(c, \tilde{c})}{2}\right) \leq \sphericalangle(c, \tilde{c}).$$

Also $\tilde{s} \in W^\perp + \tilde{c}$ and so $\|\tilde{s}\| \geq \|\Pi_W(\tilde{c})\| = \cos\left(\sphericalangle(\tilde{c}, W)\right)\|\tilde{c}\|$. Finally, for any two vectors $v, w \in \mathbb{R}^n$, we have

$$\|v - w\| \geq \|v\| \sin\left(\sphericalangle(v, w)\right),$$

and so with Lemma 3.5.5

$$\sin \sphericalangle(s^*, \tilde{s}) \leq \frac{\|s^* - \tilde{s}\|}{\|\tilde{s}\|} \leq \frac{\sqrt{n}(\kappa_W + 1)\|c - \tilde{c}\|_1}{\cos\left(\sphericalangle(\tilde{c}, W)\right)} \leq \frac{n(\kappa_W + 1)\sphericalangle(c, \tilde{c})}{\cos\left(\sphericalangle(\tilde{c}, W)\right)}.$$ □

The following theorem is an angle analogue of Theorem 3.5.6.

Theorem 3.5.9. *Let $W \subseteq \mathbb{R}^n$ be a subspace and $c, \tilde{c}, d \in \mathbb{R}^n$, $\|c\| = \|\tilde{c}\| = 1$. Let $(\tilde{x}, \tilde{s})$ be an optimal solution to LP$(W, x, \tilde{c})$ and*

$$R := \left\{ i \in [n] : \frac{\tilde{s}_i}{\|\tilde{s}\|} > \frac{n(\kappa_W + 1)\sphericalangle(c, \tilde{c})}{\cos(\sphericalangle(\tilde{c}, W))} \right\}.$$

Then for every primal optimal solution x^* to LP(W, d, c), we have $x^*_R = \mathbf{0}$. Note that $R \neq \emptyset$ whenever $\frac{n(\kappa_W+1)\sphericalangle(c,\tilde{c})}{\cos(\sphericalangle(\tilde{c},W))} < \frac{1}{\sqrt{n}}$.

Proof. Any vector $s' \in \mathbb{R}^n$ with $s'_i = 0$ for some coordinate $i \in [n]$ fulfills $\sphericalangle(s', \tilde{s}) \geq \sphericalangle(\hat{s}, \tilde{s})$, where $\hat{s} \in \mathbb{R}^n$ is defined to be $\hat{s}_j = \tilde{s}_j$ for all $j \neq i$ and $\hat{s}_j = 0$. But then,

$$\cos(\sphericalangle(s', \tilde{s})) \leq \cos(\sphericalangle(\hat{s}, \tilde{s})) = \frac{\langle \tilde{s}, \hat{s} \rangle}{\|\tilde{s}\|\|\hat{s}\|} = \frac{\|\tilde{s}\|^2 - \tilde{s}_i^2}{\|\tilde{s}\|\sqrt{\|\tilde{s}\|^2 - \tilde{s}_i^2}} = \sqrt{1 - \frac{\tilde{s}_i^2}{\|\tilde{s}\|^2}}. \tag{3.41}$$

In particular,

$$\sin(\sphericalangle(s', \tilde{s})) = \sqrt{1 - \cos^2(\sphericalangle(s', \tilde{s}))} \geq \frac{\tilde{s}_i}{\|\tilde{s}_i\|}. \tag{3.42}$$

On the other hand, by Lemma 3.5.8 there exists an optimal solution s^* that fulfills

$$\sin\left(\sphericalangle(s^*, \tilde{s})\right) \leq \frac{n(\kappa_W + 1)\sphericalangle(c, \tilde{c})}{\cos(\sphericalangle(\tilde{c}, W))} \tag{3.43}$$

Hence $s' \neq s^*$ whenever $s'_r = 0$ for some $r \in R$. Consequently, $s^*_R > 0$, implying $x^*_R = \mathbf{0}$ for every primal optimal x^* by complementary slackness. □

3.6 Circuits, integer proximity, and Graver bases

We now briefly discuss implications of circuit imbalances to the integer program (IP) of the form

$$\begin{aligned} \min\ & \langle c, x \rangle \\ & \mathbf{A}x = b \\ & x \geq \mathbf{0}, \\ & x \in \mathbb{Z}^n. \end{aligned} \tag{IP}$$

Many algorithms for (IP) solve the LP-relaxation first and deduce from the optimal solution of the relaxation information about the IP itself. The following proximity lemma shows that in case that (IP) is feasible, the distance of an optimal integral solution to the optimal solution of the relaxation can be bounded in terms of max-circuit imbalance $\bar{\kappa}_A$. So, a local search within a radius of this guaranteed proximity will provide the optimal solution for the IP; see [Lee89, Proposition 4.1].

Lemma 3.6.1. *Let x^* be an optimal solution to 1.1. Then there exists an optimal solution $\hat{x}$ to (IP) such that $\|\hat{x} - x\|_1 \leq n\bar{\kappa}_W$.*

Proof. Let $\hat{x}$ be an optimal solution to (IP) that minimizes $\|\hat{x} - x^*\|_1$ and consider $w = x^* - \hat{x} \in W$ and a conformal circuit decomposition $w = \sum_{i=1}^{k} \lambda_i g^{C_i}$ for some $k \leq n$ and circuits $C_1, \ldots, C_k$ and $\lambda_1, \ldots, \lambda_k \geq 0$. Then, $\langle c, g^{C_i} \rangle \leq 0$ for all $i \in [k]$ as otherwise $x^* - \lambda_i g^{C_i}$ would be a feasible solution to the primal of 1.1 with strictly better objective than x^*. Further, note that $\lambda_i \leq 1$ for all $i \in [k]$ as otherwise $\hat{x} - g^{C_i}$ is a feasible solution to (IP) that has an objective value as least as good as $\hat{x}$ and would in ℓ_1 norm be strictly closer to x^* than $\hat{x}$. Therefore,

$$\|\hat{x} - x^*\|_\infty \leq \sum_{i=1}^{k} \|g^{C_i}\|_\infty \leq n\bar{\kappa}_W. \tag{3.44}$$

□

Another popular and well-studied quantity in integer programming is the *Graver basis*, defined as follows.

Definition 3.6.2 (Graver basis)**.** *The Graver basis of a matrix $\mathbf{A}$, denoted by $\mathcal{G}(\mathbf{A})$, consists of all $g \in \ker(\mathbf{A}) \cap \mathbb{Z}^n$ such that there exists no $h \in (\ker(\mathbf{A}) \cap \mathbb{Z}^n) \setminus \{g\}$ such that g and h are conformal and $|h_i| \leq |g_i|$ for all $i \in [n]$. We can further define*

$$g_1(\mathbf{A}) := \max_{v \in \mathcal{G}(\mathbf{A})} \|v\|_1, \qquad g_\infty(\mathbf{A}) := \max_{v \in \mathcal{G}(\mathbf{A})} \|v\|_\infty. \tag{3.45}$$

See [LHK12] for extensive treatment of the Graver basis and [Eis+19] for more recent developments. Elementary vectors, scaled such that its entries have Greatest Common Divisor (gcd) equal to one belong to the Graver basis:

$$\left\{ g \in \mathcal{E}(W) \cap \mathbb{Z}^n : \gcd(g) = 1 \right\} \subseteq \mathcal{G}(\mathbf{A}). \tag{3.46}$$

We will furthermore see how the max-circuit imbalance measure and Graver basis are related.

Lemma 3.6.3. $\bar{\kappa}_{\mathbf{A}} \leq g_\infty(\mathbf{A}) \leq n\bar{\kappa}_{\mathbf{A}}$.

Proof. The first inequality follows from the paragraph above, noting that

$$\left\{ g^C : C \in \mathcal{C}(W) \right\} \subseteq \mathcal{G}(\mathbf{A}),$$

for the normalized elementary vectors g^C with $\operatorname{lcm}(g^C) = 1$. For the second inequality, let $g \in \mathcal{G}(\mathbf{A})$ and $g = \sum_{i=1}^{k} \lambda_i g^{C_i}$ be a conformal circuit decomposition where $k \leq n$. Note that $\lambda_i < 1$ for all $i \in [k]$ as otherwise h^i would contradict that $g \in \mathcal{G}(\mathbf{A})$. Therefore,

$$\|g\|_\infty \leq \sum_{i=1}^{k} \lambda_i \|g^{C_i}\|_\infty \leq n\bar{\kappa}_{\mathbf{A}}. \qquad \square$$

Using the Steinitz lemma, Eisenbrand, Hunkenschröder and Klein [EHK18, Lemma 2] gave a bound on $\mathfrak{g}_1(\mathbf{A})$ that only depends on m but is independent of n:

Theorem 3.6.4. *Let $\mathbf{A} \in \mathbb{Z}^{m\times n}$. Then $\mathfrak{g}_1(\mathbf{A}) \leq (2m\|\mathbf{A}\|_{\max} + 1)^m$.*

3.7 A decomposition conjecture

Let $W \subseteq \mathbb{R}^n$ be a linear space. As the analogue of maximal augmentations, we say that a conformal circuit decomposition of $z \in W$ is *maximal* if it can be obtained as follows. If $z \in W$ is an elementary vector, return the decomposition containing the single vector z. Otherwise, select an arbitrary $g \in \mathcal{E}(W)$ that is conformal with z (in particular, $\operatorname{supp}(g) \subsetneq \operatorname{supp}(z)$), and set $g^1 = \alpha g$ for the largest value $\alpha > 0$ such that $z - g^1$ is conformal with z. Then, recursively apply this procedure to $z - g^1$ to obtain the other elementary vectors $g^2, \dots, g^h$. We have $h \leq n$, since the support decreases by at least one due to the maximal choice of α. If $\kappa_W = \dot{\kappa}_W = 1$, then it is easy to verify the following.

Proposition 3.7.1. *Let $W \subseteq \mathbb{R}^n$ be a linear space with $\kappa_W = 1$, and let $z \in W \cap \mathbb{Z}^n$. Then, for every maximal conformal circuit decomposition $z = \sum_{k=1}^{h} g^k$, we have $g^k \in \mathcal{E}(W) \cap \mathbb{Z}^n$.*

We formulate a conjecture asserting that this property generalizes for arbitrary $\dot{\kappa}_W$ values. Note that in the conjecture, we only require the existence of some (not necessarily maximal) conformal circuit decomposition.

Conjecture 3.7.1.1. *Let $W \subseteq \mathbb{R}^n$ be a rational linear subspace. Then, for every $z \in W \cap \mathbb{Z}^n$, there exists a conformal circuit decomposition $z = \sum_{k=1}^{h} g^k$, $h \leq n$ such that each g^k is a $1/\dot{\kappa}_W$-integral vector in $\mathcal{E}(W)$.*

Note that it is equivalent to require the same property for elements of the Graver basis $z \in \mathcal{G}(\mathbf{A})$. Hence, the conjecture asserts that every vector in the Graver basis is a *nice* combination of elementary vectors.

We present some preliminary evidence towards this conjecture:

Proposition 3.7.2. *Let $W \subseteq \mathbb{R}^n$ be a rational linear subspace with $\kappa^*_W = 1$. Then, for every $z \in W \cap \mathbb{Z}^n$, and every maximal conformal circuit decomposition $z = \sum_{k=1}^{h} g^k$, we have that g^k is a $1/\dot{\kappa}$-integral vector in $\mathcal{E}(W)$.*

Proof. Assume $\kappa_{\mathbf{D}W} = 1$ for some $\mathbf{D} \in \mathfrak{D}_n$. By Theorem 3.4.8, we can select $\mathbf{D}$ such that all diagonal entries $d_i = \mathbf{D}_{ii} \in \mathbb{Z}$ and $d_i | \dot{\kappa}_W$. Let $z = \sum_{k=1}^{h} g^k$ be any maximal conformal circuit decomposition of $z \in W \cap \mathbb{Z}^n$. Now, $\mathbf{D}z = \sum_{k=1}^{h} \mathbf{D}g^k$ is also a maximal conformal circuit decomposition of $\mathbf{D}z \in \mathbf{D}W \cap \mathbb{Z}^n$. By Proposition 3.7.1, $\mathbf{D}g^k \in \mathcal{E}(\mathbf{D}W) \cap \mathbb{Z}^n$. Since $d_i \mid \dot{\kappa}_W$, this implies that g^k is $1/\dot{\kappa}_W$-integral. $\square$

By Theorem 3.4.10, this implies the conjecture whenever $\dot{\kappa}_W = p^\alpha$ for $p \in \mathbb{P}$, $p > 2$, $\alpha \in \mathbb{N}$. Let us now consider the case when $\dot{\kappa}_W$ is a power of 2. We verify the conjecture when the decomposition contains at most three terms.

Proposition 3.7.3. *Let $W \subseteq \mathbb{R}^n$ be a rational linear subspace with $\varkappa_W = 2^\alpha$ for $\alpha \in \mathbb{Z}^n$. If $z \in W \cap \mathbb{Z}^n$ has a maximal conformal circuit decomposition $z = \sum_{k=1}^h g^k$ with $h \leq 3$, then each g^k is a $1/\dot{\varkappa}$-integral vector in $\mathcal{E}(W)$.*

Proof. Let us write the maximal conformal circuit decomposition in the form $z = \sum_{k=1}^h \lambda_k g^k$ such that $\operatorname{lcm}(g^k) = 1$, and all entries $g_i^k \in \{\pm 1, \pm 2, \pm 4, \ldots, \pm 2^\alpha\}$ for $k \in [h]$, $i \in [n]$. There is nothing to prove for $h = 1$. If $h = 2$, then by the maximality of the decomposition, $\lambda_1 = \min_j\{z_j/g_j^1\}$. Hence, λ_1 is $1/2^\alpha$-integral. Consequently, both $\lambda_1 g^1$ and $\lambda_2 g^2 = z - \lambda_1 g^1$ are $1/2^\alpha$-integral.

If $h = 3$, then $\lambda_1 g^1$ is $1/2^\alpha$-integral as above. It also follows that $\lambda_2 g^2$ and $\lambda_3 g^3$ are $1/2^\beta$-integral for some $\beta \geq \alpha$. Let us choose the smallest such β; we are done if $\beta = \alpha$.

Assume for a contradiction $\beta > \alpha$. Let $\mu_k = 2^\beta \lambda_k$ for $k = 1, 2, 3$. Thus, $\mu_k \in \mathbb{Z}$, μ_1 is even, and at least one of μ_2 and μ_3 is odd. We show that both μ_2 and μ_3 must be odd. Let us first assume that μ_3 is odd. There exists an $i \in [n]$ such that $|g_i^3| = 1$. Then, $2^\beta z_i = \mu_1 g_i^1 + \mu_2 g_i^2 + \mu_3 g_i^3$ implies that μ_2 must also be odd. Similarly, if μ_2 is odd then μ_3 must also be odd.

Let us take any $j \in [n]$ such that $g_j^1 = 0$. Then, $2^\beta z_i = \mu_2 g_j^2 + \mu_3 g_j^3$. Noting that $|g_j^2|$ and $|g_j^3|$ are powers of 2, both at most 2^α, it follows that $|g_j^2| = |g_j^3|$; by conformity, we have $g_j^2 = g_j^3$.

Consequently, $\operatorname{supp}(g^2 - g^3) \subseteq \operatorname{supp}(g^1)$. We have $g^2 - g^3 \in W \setminus \{\mathbf{0}\}$, and the containment is strict by the maximality of the decomposition: there exists an index $i \in \operatorname{supp}(z)$ such that $z_j = \lambda_1 g_j^1$. This contradicts the fact that $g^1 \in \mathcal{E}(W)$. □

3.A On Example 3.2.20

Example 3.2.20. *Consider the matrix*

$$\mathbf{A} = \begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix}. \tag{3.11}$$

For this matrix $\dot{\varkappa}_{\mathbf{A}} = 5850 = 2 \times 3^2 \times 5^2 \times 13$ holds, and there exists no $\tilde{\mathbf{A}} \in \mathbb{Z}^{2\times 4}$ such that $\ker(\tilde{\mathbf{A}}) = \ker(\mathbf{A})$ and the inverse of every nonsingular 2×2 submatrix of $\tilde{\mathbf{A}}$ is $1/5850$-integral.

Proof. We know all other representations of the space like $\tilde{A}$ such that $\ker(\tilde{A}) = \ker(A)$ are of the form $\tilde{A} = BA$ where B is a 2×2 invertible matrix. Since $A_{11} = 1$ then to get an integral $\tilde{A}$ we need to have integer B_{11} and B_{21}. Furthermore since the g.c.d. of the numbers in the second column is equal to 1, then B_{12} and B_{22} should be integers as well.

It can be verified by computer that the only 2×1 matrices like v such that all entries of $v^T A$ are divisors of 5850 are

$$\pm\begin{bmatrix} 9 \\ -4 \end{bmatrix}, \pm\begin{bmatrix} 10 \\ -3 \end{bmatrix}, \pm\begin{bmatrix} 13 \\ -3 \end{bmatrix}, \pm\begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

Checking all different 2×2 matrices we can get these matrices:

$$\begin{bmatrix} 9 & -4 \\ 10 & -3 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} \mathbf{9} & \mathbf{-25} & 0 & -13 \\ \mathbf{10} & \mathbf{-9} & 13 & 0 \end{bmatrix}$$
$$\begin{bmatrix} 13 & -3 \\ 10 & -3 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} \mathbf{13} & 0 & \mathbf{25} & 9 \\ \mathbf{10} & -9 & \mathbf{13} & 0 \end{bmatrix}$$
$$\begin{bmatrix} 9 & -4 \\ 13 & -3 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} \mathbf{9} & -25 & 0 & \mathbf{-13} \\ \mathbf{13} & 0 & 25 & \mathbf{9} \end{bmatrix}$$
$$\begin{bmatrix} 0 & 1 \\ 9 & -4 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{13} & 9 & \mathbf{10} \\ 9 & \mathbf{-25} & 0 & \mathbf{-13} \end{bmatrix}$$
$$\begin{bmatrix} 0 & 1 \\ 10 & -3 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{13} & \mathbf{9} & 10 \\ 10 & \mathbf{-9} & \mathbf{13} & 0 \end{bmatrix}$$
$$\begin{bmatrix} 0 & 1 \\ 13 & -3 \end{bmatrix}\begin{bmatrix} 1 & 3 & 4 & 3 \\ 0 & 13 & 9 & 10 \end{bmatrix} = \begin{bmatrix} 0 & 13 & \mathbf{9} & \mathbf{10} \\ 13 & 0 & \mathbf{25} & \mathbf{9} \end{bmatrix}$$

All of these matrices contain a 2×2 submatrix such that its inverse is not $\frac{1}{5850}$-integral. □

4 A scaling-invariant algorithm for linear programming whose running time depends only on the constraint matrix

Following the breakthrough work of Tardos [Tar85] in the bit-complexity model, Vavasis and Ye [VY96] gave the first exact algorithm for LP in the real model of computation with running time depending only on the constraint matrix. For solving LP as in the form System 1.1, i.e., $\max \langle c, x \rangle$, s.t. $\mathbf{A}x = b, x \geq \mathbf{0}, \mathbf{A} \in \mathbb{R}^{m \times n}$, Vavasis and Ye developed a primal-dual IPM using a LLS step, and showed that $O(n^{3.5} \log(\bar{\chi}_{\mathbf{A}} + n))$ iterations suffice to solve LP exactly, where $\bar{\chi}_{\mathbf{A}}$ is a condition measure controlling the size of solutions to linear systems related to $\mathbf{A}$.

Monteiro and Tsuchiya [MT03], noting that the central path is invariant under rescalings of the columns of $\mathbf{A}$ and c, asked whether there exists an LP algorithm depending instead on the measure $\bar{\chi}^*_{\mathbf{A}}$, defined as the minimum $\bar{\chi}_{\mathbf{AD}}$ value achievable by a column rescaling $\mathbf{AD}$ of $\mathbf{A}$, and gave strong evidence that this should be the case. We resolved this open question affirmatively in Chapter 3 by providing a near-optimal rescaling of the constraint matrix.

While this resolved Monteiro and Tsuchiya's question by appropriate preprocessing, it falls short of providing either a truly scaling invariant algorithm or an improvement upon the base LLS analysis. In this vein, as our main result in this chapter we develop a *scaling invariant* LLS algorithm, which uses and dynamically maintains improving estimates of the circuit ratio digraph, together with a refined potential function based analysis for LLS algorithms in general. With this analysis, we derive an improved $O(n^{2.5} \log(n) \log(\bar{\chi}^*_{\mathbf{A}} + n))$ iteration bound for optimally solving LP using our algorithm. The same argument also yields a factor $n/\log n$ improvement on the iteration complexity bound of the original Vavasis-Ye algorithm.

This chapter is based on joint work with Daniel Dadush, Sophie Huiberts, and László A. Végh [DHNV20].

Contents

4.1 Introduction

In a seminal work, Vavasis and Ye [VY96] introduced a new type of interior-point method that optimally solves System 1.1 within $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}}+n))$ iterations, where the condition number $\bar{\chi}_{\mathbf{A}}$ controls the size of solutions to certain linear systems related to the kernel of $\mathbf{A}$ (see Section 4.2 for the formal definition).

Before detailing the Vavasis–Ye (henceforth VY) algorithm, we recall the basics of path following interior-point methods. If both the primal and dual problems in System 1.1 are strictly feasible, the central path for System 1.1 is the curve $\{\, x(\mu), y(\mu), s(\mu)) : \mu > 0) \,\}$ defined by

$$\begin{aligned} x(\mu)_i s(\mu)_i &= \mu, \quad \forall i \in [n] \\ \mathbf{A}x(\mu) &= b,\ x(\mu) > \mathbf{0}, \\ \mathbf{A}^\top y(\mu) + s(\mu) &= c,\ s(\mu) > \mathbf{0}, \end{aligned} \tag{CP}$$

which converges to complementary optimal primal and dual solutions (x^*, y^*, s^*) as $\mu \to 0$, recalling that the duality gap at time μ is exactly $x(\mu)^\top s(\mu) = n\mu$. We thus refer to μ as the normalized duality gap. Methods that "follow the path" generate iterates that stay in a certain neighborhood around it while trying to achieve rapid multiplicative progress w.r.t. to μ, where given (x, y, s) 'close' to the path, we define the normalized duality gap as $\mu(x, y, s) = \sum_{i=1}^n x_i s_i / n$. Given a target parameter μ' and starting point close to the path at parameter μ, standard path following methods [Gon92] can compute a point at parameter lower than μ' in at most $O(\sqrt{n}\log(\mu/\mu'))$ iterations, and hence the quantity $\log(\mu/\mu')$ can be usefully interpreted as the length of the corresponding segment of the central path.

Crossover events and layered least squares steps. At a very high level, Vavasis and Ye show that the central path can be decomposed into at most $\binom{n}{2}$ short but curved segments, possibly joined by long (a priori unbounded) but very straight segments. At the end of each curved segment, they show that a new ordering relation $x_i(\mu) > x_j(\mu)$—called a crossover event—is implicitly learned. This inequality did not hold at the start of the segment, but is guaranteed to hold at every point from the end of the segment onwards. These $\binom{n}{2}$ relations give a combinatorial way to measure progress along the central path. In contrast to Tardos's algorithm, where the main progress is setting variables to zero explicitly, the variables participating in crossover events cannot be identified; the analysis only shows their existence.

At a technical level, the VY-algorithm is a variant of the Mizuno–Todd–Ye [MTY93] predictor-corrector method (MTY P-C). In predictor-corrector methods, corrector steps bring an iterate closer to the path, i.e., improve centrality, and predictor steps "shoot down" the path, i.e., reduce μ without losing too much centrality. Vavasis and Ye's main algorithmic innovation was the introduction of a new predictor step, called the *'layered least squares'* (LLS) step, which crucially allowed them to cross each aforementioned "straight" segment of the central path in *a single step*, recalling that these straight segments may be arbitrarily long. To traverse the short and curved segments of the path, the standard predictor step, known as *affine scaling* (AS), in fact suffices.

To compute the LLS direction, the variables are decomposed into 'layers' $J_1 \cup J_2 \cup \ldots \cup J_p = [n]$. The goal of such a decomposition is to eventually learn a refinement of the optimal partition of the variables $B^* \cup N^* = [n]$, where $B^* := \{i \in [n] : x_i^* > 0\}$ and $N^* := \left\{ i \in [n] : s_i^* > 0 \right\}$ for the limit optimal solution (x^*, y^*, s^*).

The primal affine scaling direction can be equivalently described by solving a weighted least squares problem in $\ker(\mathbf{A})$, with respect to a weighting defined according to the current iterate. The primal LLS direction is obtained by solving a series of weighted least squares problems, starting with focusing only on the final layer J_p. This solution is gradually extended to the higher layers (i.e., layers with lower indices). The dual directions have analogous interpretations, with the solutions on the layers obtained in the opposite direction, starting with J_1. If we use the two-level layering $J_1 = B^*$, $J_2 = N^*$, and are sufficiently close to the limit (x^*, y^*, s^*) of the central path, then the LLS step reaches an exact optimal solution in a single step. We note that standard Affine Scaling (AS) steps generically never find an exact optimal solution, and thus some form of "LLS rounding" in the final iteration is always necessary to achieve finite termination with an exact optimal solution.

Of course, guessing B^* and N^* correctly is just as hard as solving System 1.1. Still, if we work with a "good" layerings, these will reveal new information about the "optimal order" of the variables, where B^* is placed on higher layers than N^*. The crossover events correspond to swapping two wrongly ordered variables into the correct ordering. Namely, a variable $i \in B^*$ and $j \in N^*$ are currently ordered on the same layer, or j is in a higher layer than i. After the crossover event, i will always be placed on a higher layer than j.

Computing good layerings and the $\bar{\chi}_{\mathbf{A}}$ condition measure. Given the above discussion, the obvious question is how to come up with "good" layerings? The philosophy behind LLS can be stated as saying that if modifying a set of variables x_I barely affects the variables in $x_{[n]\setminus I}$ (recalling that movement is constrained to $\Delta x \in \ker(\mathbf{A})$), then one should optimize over x_I without regard to the effect on $x_{[n]\setminus I}$; hence x_I should be placed on lower layers.

VY's strategy for computing such layerings was to directly use the size of the coordinates of the current iterate x (where (x, y, s) is a point near the central path). In particular, assuming $x_1 \geq x_2 \geq \ldots \geq x_n$, the layering $J_1 \cup J_2 \cup \ldots \cup J_p = [n]$ corresponds to consecutive intervals constructed in decreasing order of x_i values. The break between J_i and J_{i+1} occurs if the gap $x_r/x_{r+1} > g$, where r is the rightmost element of J_i and $g > 0$ is a threshold parameter. Thus, the expectation is that if $x_i > g x_j$, then a small multiplicative change to x_j, subject to moving in $\ker(\mathbf{A})$, should induce a small multiplicative change to x_i. By proximity to the central path, the dual ordering is reversed as mentioned above.

The threshold g for which this was justified in the VY-algorithm is a function of the $\bar{\chi}_{\mathbf{A}}$ condition measure. We now provide a convenient definition that immediately yields this justification (see Proposition 3.3.6). Letting $W = \ker(\mathbf{A})$ and $\pi_I(W) = \{x_I : x \in W\}$, we define $\bar{\chi}_{\mathbf{A}} := \bar{\chi}_W$ 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 $\|y\| \leq M\|z\|$. Thus, a change of norm ϵ in the variables in I can be lifted to a change

of norm at most $\bar{\chi}_{\mathbf{A}}\epsilon$ in the variables in $[n] \setminus I$. Crucially, $\bar{\chi}$ is a "self-dual" quantity. That is, $\bar{\chi}_W = \bar{\chi}_{W^\perp}$, where $W^\perp = \text{range}(\mathbf{A}^\top)$ is the movement subspace for the dual problem, justifying the reversed layering for the dual (see Sections 4.2 for more details).

The question of scale invariance and $\bar{\chi}^*_{\mathbf{A}}$. While the VY layering procedure is powerful, its properties are somewhat mismatched with those of the central path. In particular, variable ordering information has *no intrinsic meaning* on the central path, as the path itself is *scaling invariant*. Namely, the central path point $(x(\mu), y(\mu), s(\mu))$ w.r.t. the problem instance $(\mathbf{A}, b, c)$ is in bijective correspondence with the central path point $(\mathbf{D}^{-1}x(\mu), \mathbf{D}y(\mu), \mathbf{D}s(\mu)))$ w.r.t. the problem instance $(\mathbf{AD}, \mathbf{D}c, b)$ for any positive diagonal matrix $\mathbf{D}$. The standard path following algorithms are also scaling invariant in this sense.

This lead Monteiro and Tsuchiya [MT03] to ask whether a scaling invariant LLS algorithm exists. They noted that any such algorithm would then depend on the potentially much smaller parameter

$$\bar{\chi}^*_{\mathbf{A}} := \inf_{\mathbf{D}\in\mathfrak{D}_n} \bar{\chi}_{\mathbf{AD}}\,, \tag{4.1}$$

where the infimum is taken over the set of $n \times n$ positive diagonal matrices. Thus, Monteiro and Tsuchiya's question can be rephrased to whether there exists an exact LP algorithm with running time $\text{poly}(n, m, \log \bar{\chi}^*_{\mathbf{A}})$.

Substantial progress on this question was made in the followup works [LMT09; MT05]. The paper [MT05] showed that the number of iterations of the MTY predictor-corrector algorithm [MTY93] can get from $\mu_0 > 0$ to $\eta > 0$ on the central path in

$$O\Big(n^{3.5}\log \bar{\chi}^*_{\mathbf{A}} + \min\{n^2 \log\log(\mu^0/\eta), \log(\mu^0/\eta)\}\Big)$$

iterations. This is attained by showing that the standard AS steps are reasonably close to the LLS steps. This proximity can be used to show that the AS steps can traverse the "curved" parts of the central path in the same iteration complexity bound as the VY algorithm. Moreover, on the "straight" parts of the path, the rate of progress amplifies geometrically, thus attaining a log log convergence on these parts. Subsequently, [LMT09] developed an affine invariant *trust region* step, which traverses the full path in $O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}} + n))$ iterations. However, the running time of each iteration is weakly polynomial in b and c. The question of developing an LP algorithm with complexity bound $\text{poly}(n, m, \log \bar{\chi}^*_{\mathbf{A}})$ thus remained open.

A related open problem to the above is whether it is possible to compute a near-optimal rescaling $\mathbf{D}$ to the constraint matrix $\mathbf{A}$? This would give an alternate pathway to the desired LP algorithm by simply preprocessing the matrix $\mathbf{A}$. The related question of approximating $\bar{\chi}_{\mathbf{A}}$ was already studied by Tunçel [Tun99], who showed NP-hardness for approximating $\bar{\chi}_{\mathbf{A}}$ to within a $2^{\text{poly}(\text{rk}(\mathbf{A}))}$ factor. Taken at face value, this may seem to suggest that approximating the rescaling $\mathbf{D}$ should be hard.

A further open question is whether Vavasis and Ye's cross-over analysis can be improved. Ye showed in [Ye06] that the iteration complexity can be reduced to $O(n^{2.5}\log(\bar{\chi}_{\mathbf{A}} + n))$ for feasibility problems and further to $O(n^{1.5}\log(\bar{\chi}_{\mathbf{A}} + n))$ for homogeneous systems, though the $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}} + n))$ bound for optimization has not been improved since [VY96].

4.1.1 The contributions in [DHNV20]

In [DHNV20], we resolved all the above questions in the affirmative. We detail our contributions below.

1. Finding an approximately optimal rescaling. As our first contribution, we give an $O(m^2n^2 + n^3)$ time algorithm that works on the linear matroid of $\mathbf{A}$ to compute a diagonal rescaling matrix $\mathbf{D}$ which achieves $\bar{\chi}_{\mathbf{AD}} \leq n(\bar{\chi}^*_{\mathbf{A}})^3$, given any $m \times n$ matrix $\mathbf{A}$. Furthermore, this same algorithm allows us to approximate $\bar{\chi}_{\mathbf{A}}$ to within a factor $n(\bar{\chi}^*_{\mathbf{A}})^2$. The algorithm bypasses Tunçel's hardness result by allowing the approximation factor to depend on $\mathbf{A}$ itself, namely on $\bar{\chi}^*_{\mathbf{A}}$. This gives a simple first answer to Monteiro and Tsuchiya's question: by applying the Vavasis-Ye algorithm directly on the preprocessed $\mathbf{A}$ matrix, we may solve any LP with constraint matrix $\mathbf{A}$ using $O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}} + n))$ iterations. Note that the approximation factor $n(\bar{\chi}^*_{\mathbf{A}})^2$ increases the runtime only by a constant factor.

A proof of this result is given in Theorem 3.4.7.

2. Scaling invariant LLS algorithm. While the above yields an LP algorithm with $\text{poly}(n, m, \log \bar{\chi}^*_{\mathbf{A}})$ running time, it does not satisfactorily address Monteiro and Tsuchiya's question on a scaling invariant algorithm. As our second contribution, we use the circuit ratio digraph directly to give a natural scaling invariant LLS layering algorithm together with a scaling invariant crossover analysis.

At a conceptual level, we show that the circuit ratios give a scale invariant way to measure whether '$x_i > x_j$' and enable a natural layering algorithm. Assume for now that the value of the circuit imbalance κ_{ij} is known for every pair (i, j). Given the circuit ratio graph induced by the κ_{ij}'s and given a primal point x near the path, our layering algorithm can be described as follows. We first rescale the variables so that x becomes the all ones vector, which rescales κ_{ij} to $\kappa_{ij}x_i/x_j$. We then restrict the graph to its edges of length $\kappa_{ij}x_i/x_j \geq 1/\text{poly}(n)$—the *long edges* of the (rescaled) circuit ratio graph—and let the layering $J_1 \cup J_2 \cup \ldots \cup J_p$ be a topological ordering of its Strongly Connected Component (SCC) with edges going from left to right. Intuitively, variables that "affect each other" should be in the same layer, which motivates the SCC definition.

We note that our layering algorithm does not have access to the true circuit ratios κ_{ij}; these are in fact NP-hard to compute. Getting a good enough initial estimate for our purposes however is easy: we let $\hat{\kappa}_{ij}$ be the ratio corresponding to an *arbitrary* circuit containing i and j. This already turns out to be within a factor $(\bar{\chi}^*_{\mathbf{A}})^2$ from the true value κ_{ij}—recall this is the maximum over all such circuits. Our layering algorithm learns better circuit ratio estimates if the lifting costs of our SCC layering, i.e., how much it costs to lift changes from lower layer variables to higher layers (as in the definition of $\bar{\chi}_{\mathbf{A}}$), are larger than we expected them to be based on the previous estimates.

We develop a scaling-invariant analogue of cross-over events as follows. Before the crossover event, $\text{poly}(n)(\bar{\chi}^*_{\mathbf{A}})^n > \kappa_{ij}x_i/x_j$, and after the crossover event, $\text{poly}(n)(\bar{\chi}^*_{\mathbf{A}})^n < \kappa_{ij}x_i/x_j$ for all further central path points. Our analysis relies on $\bar{\chi}^*_{\mathbf{A}}$ in only a minimalistic way, and does not require an estimate on the value of $\bar{\chi}^*_{\mathbf{A}}$. Namely, it is only used to show that if $i, j \in J_q$, for a layer $q \in [p]$, then the rescaled circuit ratio $\kappa_{ij}x_i/x_j$ is in the range $(\text{poly}(n)\bar{\chi}^*_{\mathbf{A}})^{\pm O(|J_q|)}$. The argument to show this crucially utilizes the maximum geometric mean cycle characterization. Furthermore, unlike prior analyses [MT03; VY96], our definition of a "good" layering (i.e., 'balanced' layerings, see Section 4.3.5), is completely independent of $\bar{\chi}^*_{\mathbf{A}}$.

3. Improved potential analysis. As our third contribution, we improve the Vavasis–Ye crossover analysis using a new and simple potential function based approach. When applied to our new LLS algorithm, we derive an $O(n^{2.5}\log n \log(\bar{\chi}^*_{\mathbf{A}} + n))$ iteration bound for path following, improving the polynomial term by an $\Omega(n/\log n)$ factor compared to the VY analysis.

Our potential function can be seen as a fine-grained version of the crossover events as described above. In case of such a crossover event, it is guaranteed that in every subsequent iteration, i is in a layer before j. We analyze less radical changes instead: an "event" parametrized by τ means that i

and j are currently together on a layer of size $\leq \tau$, and after the event, i is on a layer before j, or if they are together on the same layer, then this layer must have size $\geq 2\tau$. For every LLS step, we can find a parameter τ such that an event of this type happens concurrently for at least $\tau - 1$ pairs within the next $O(\sqrt{n}\tau \log(\bar{\chi}^*_{\mathbf{A}} + n))$ iterations,

Our improved analysis is also applicable to the original VY-algorithm. Let us now comment on the relation between the VY-algorithm and our new algorithm. The VY-algorithm starts a new layer once $x_{\pi(i)} > g x_{\pi(i+1)}$ between two consecutive variables where the permutation π is a non-increasing order of the x_i variables, and $g = \text{poly}(n)\bar{\chi}_{\mathbf{A}}$. Setting the initial 'estimates' $\hat{\kappa}_{ij} = \bar{\chi}_{\mathbf{A}}$ for a suitable polynomial, our algorithm runs the same way as the VY algorithm. Using these estimates, the layering procedure becomes much simpler: there is no need to verify 'balancedness' as in our algorithm.

However, using estimates $\hat{\kappa}_{ij} = \bar{\chi}_{\mathbf{A}}$ has drawbacks. Most importantly, it does not give a lower bound on the true circuit ratio κ_{ij}—to the contrary, g will be an upper bound. In effect, this causes VY's layers to be "much larger" than ours, and for this reason, the connection to $\bar{\chi}^*_{\mathbf{A}}$ is lost. Nevertheless, our potential function analysis can still be adapted to the VY-algorithm to obtain the same $\Omega(n/\log n)$ improvement on the iteration complexity bound; see Section 4.4.1 for more details.

4.1.2 Related work

Since the seminal works of Karmarkar [Kar84] and Renegar [Ren88], there has been a tremendous amount of work on speeding up and improving interior-point methods. In contrast to the present work, the focus of these works has mostly been to improve complexity of *approximately solving* LPs. Progress has taken many forms, such as the development of novel barrier methods, such as Vaidya's volumetric barrier [Vai89] and the recent entropic barrier of Bubeck and Eldan [BE15] and the weighted log-barrier of Lee and Sidford [LS14; LS19], together with new path following techniques, such as the predictor-corrector framework [Meh92; MTY93], as well as advances in fast linear system solving [LS15; ST04a]. For this last line, there has been substantial progress in improving IPM by amortizing the cost of the iterative updates, and working with approximate computations, see e.g., [Ren88; Vai89] for classical results. Recently, Cohen, Lee, and Song [CLS19] developed a new inverse maintenance scheme to get a randomized $\widetilde{O}(n^\omega \log(1/\varepsilon))$-time algorithm for ε-approximate LP, which was derandomized by van den Brand [Bra20]; here $\omega \approx 2.37$ is the matrix multiplication exponent. A recent result by van den Brand et al. [BTSS20] obtained a randomized $\tilde{O}(nm + m^3)$ algorithm. For special classes of LP such as network flow and matching problems, even faster algorithms have been obtained using, among other techniques, fast Laplacian solvers, see e.g. [Bra+20; Bra+21; DS08; Mad13]. In Chapter 7 we show how Tardos's framework can be extended to the real model of computation [DNV20], showing that $\text{poly}(n, m, \log \bar{\chi}_{\mathbf{A}})$ running time can be achieved using approximate solvers in a black box manner. Combined with [Bra20], one obtains a deterministic $\widetilde{O}(mn^\omega \log(\bar{\chi}_{\mathbf{A}}))$ LP algorithm; using the initial rescaling subroutine from Chapter 3, the dependence can be improved to $\bar{\chi}^*_{\mathbf{A}}$ resulting in a running time of $\widetilde{O}(mn^{\omega+1} \log(\bar{\chi}^*_{\mathbf{A}}))$. A weaker extension of Tardos's framework to the real model of computation was previously given by Ho and Tunçel [HT02].

With regard to LLS algorithms, the original VY-algorithm required explicit knowledge of $\bar{\chi}_{\mathbf{A}}$ to implement their layering algorithm. The paper [MMT98] showed that this could be avoided by computing all LLS steps associated with n candidate partitions and picking the best one. In particular, they showed that all such LLS steps can be computed in $O(m^2 n)$ time. In [MT03], an alternate approach was presented to compute an LLS partition directly from the coefficients of the

AS step. We note that these methods crucially rely on the variable ordering, and hence are not scaling invariant. Kitahara and Tsuchiya [KT13], gave a 2-layer LLS step which achieves a running time depending only on $\bar{\chi}^*_{\mathbf{A}}$ and right-hand side b, but with no dependence on the objective, assuming the primal feasible region is bounded.

A series of papers have studied the central path from a differential geometry perspective. Monteiro and Tsuchiya [MT08] showed that a curvature integral of the central path, first introduced by Sonnevend, Stoer, and Zhao [SSZ91], is in fact upper bounded by $O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}}+n))$. This has been extended to SDP and symmetric cone programming [KOT14], and also studied in the context of information geometry [KOT13].

Circuits have appeared in several papers on linear and integer optimization (see [LKS19] and references within). The idea of using circuits within the context of LP algorithms also appears in [DHL15]. They develop a circuit augmentation framework for LP (as well ILP) and show that simplex-like algorithms that take steps according to the "best circuit" direction achieves linear convergence, though these steps are hard to compute. Very recently, [DKNV22] used circuit imbalance measures to obtain a circuit augmentation algorithm for LP with $\mathrm{poly}(n,\log(\bar{\chi}_{\mathbf{A}}))$ iterations. We refer to [ENV22] for an overview on circuit imbalances and their applications.

Our algorithm makes progress towards strongly polynomial solvability of LP, by improving the dependence $\mathrm{poly}(n,m,\log\bar{\chi}_{\mathbf{A}})$ to $\mathrm{poly}(n,m,\log\bar{\chi}^*_{\mathbf{A}})$. However, in a remarkable recent paper, Allamigeon et al. [ABGJ18] have shown, using tools from tropical geometry, that path-following methods for the standard logarithmic barrier *cannot* be strongly polynomial. In particular, they give a parametrized family of instances, where, for sufficiently large parameter values, any sequence of iterations following the central path must be of exponential length—thus, $\bar{\chi}^*_{\mathbf{A}}$ will be doubly exponential.

4.1.3 Organization

The rest of the chapter is organized as follows. We conclude this section by introducing some notation. Section 4.2 discusses our results on the circuit imbalance measure.

In Section 4.3, we develop our scaling invariant interior-point method. Interior-point preliminaries are given in Section 4.3.1, Section 4.3.2 introduces the affine scaling and layered-least-squares directions, and proves some basic properties, Section 4.3.3 provides a detailed overview of the high level ideas and a roadmap to the analysis and Section 4.3.4 further develops the theory of LLS directions and introduces partition lifting scores. Section 4.3.5 gives our scaling invariant layering procedure, and our overall algorithm can be found in Section 4.3.6.

In Section 4.4, we give the potential function proof for the improved iteration bound, relying on technical lemmas. The full proof of these lemmas is deferred to Section 4.6; however, Section 4.4 provides the high-level ideas to each proof. Section 4.4.1 shows that our argument also leads to a factor $\Omega(n/\log n)$ improvement in the iteration complexity bound of the VY-algorithm.

In Section 4.5, we prove the technical properties of our LLS step, including its proximity to AS and step length estimates. Finally, in Section 4.7, we discuss the initialization of the interior-point method.

4.2 Finding an approximately optimal rescaling

Recall that a matroid is non-separable if the circuit hypergraph is connected; precise definitions and background were described in Section 3.4.1. Further, recall the definitions of κ_W, $\bar{\chi}$ and L from Definition 3.1.1, Equation (3.13).

Symbol	Description	Defined in
$w = (x, y, s)$	tuple of feasible solutions	Section 4.3.1
$\mu(w)$	normalized duality gap	Section 4.3.1
L_I^W	lifting map $\pi_I(W) \to W$	Definition 3.3.4
$\ell^W(\mathcal{J})$	lifting score	Definition 4.2.1
$\mathcal{N}(\beta)$	ℓ_2-neighborhood of the central path	Section 4.3.1
$\Delta w^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta y^{\mathrm{a}}, \Delta s^{\mathrm{a}})$	affine scaling direction	Section 4.3.2
$\Delta w^{\mathrm{c}} = (\Delta x^{\mathrm{c}}, \Delta y^{\mathrm{c}}, \Delta s^{\mathrm{c}})$	centrality direction	Section 4.3.2
$\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$	layered least-squares scaling direction	Section 4.3.2
$\epsilon_I^{\mathrm{a}}(w)$	norm of AS residuals on $I \subseteq [n]$	Equation (4.13)
$w^+ = w + \alpha\Delta w$	iterate after predictor step	Section 4.3.2
$\delta = \delta(w)$	approximate rescaled dual $\delta = s^{1/2}x^{-1/2}$	Equation (4.7)
$Rx^{\mathrm{a}}, Rs^{\mathrm{a}}, Rx^{\mathrm{ll}}, Rs^{\mathrm{ll}}$	residuals	Equation (4.11)
$G_{w,\sigma} = ([n], E_{w,\sigma})$	long edge graph	Page 68
$\mathcal{J} = (J_1, J_2, \ldots, J_p)$	ordered partition	Page 68
$W_{\mathcal{J},k}$	subspace	Section 4.3.4
γ	parameter	Equation (4.23)
$\hat{G}_{\delta,\sigma} = ([n], \hat{E}_{\delta,\sigma})$	auxiliary graph	Definition 4.3.11
$\varrho^\mu(i,j), \Psi^\mu(i,j), \Psi(\mu)$	potential function	Page 78
$\xi_I^{\mathrm{ll}}(w)$	norm of LLS residuals on $I \subseteq [n]$	Equation (4.24)
$\mathfrak{B}, \mathfrak{N}$	Partition of variables based on $Rx^{\mathrm{ll}}, Rs^{\mathrm{ll}}$	Equation (4.41)

Table 4.1: Recurring symbols that will be defined throughout the chapter.

The following notation will be convenient for our algorithm.

Definition 4.2.1. *For a subspace $W \subseteq \mathbb{R}^n$ and an index set $I \subseteq [n]$, if $\pi_I(W) \neq \{0\}$ then we define the* lifting score

$$\ell^W(I) := \sqrt{\|L_I^W\|^2 - 1}\,. \tag{4.2}$$

Otherwise, we define $\ell^W(I) = 0$. This means that for any $z \in \pi_I(W)$ and $x = L_I^W(z)$, $\|x_{[n]\setminus I}\| \leq \ell^W(I)\|z\|$.

The next lemma provides a subroutine that efficiently yields upper bounds on $\ell^W(I)$ or lower bounds on some circuit imbalance values. Recall the definition of the lifting score $\ell^W(I)$ from (4.2.1).

Lemma 4.2.2. *There exists a subroutine VERIFY-LIFT(W, I, θ) that, given a linear subspace $W \subseteq \mathbb{R}^n$, an index set $I \subseteq [n]$, and a threshold $\theta \in \mathbb{R}_{++}$, either returns the answer 'pass', verifying $\ell^W(I) \leq \theta$, or returns the answer 'fail', and a pair $i \in I, j \in [n] \setminus I$ such that $\theta/n \leq \kappa_{ij}^W$. The running time can be bounded as $O(n(n-m)^2)$.*

Proof. Take any minimal $I' \subset I$ such that $\dim(\pi_{I'}(W)) = \dim(\pi_I(W))$. Then we know that $\pi_{I'}(W) = \mathbb{R}^{I'}$ and for $p \in \pi_I(W)$ we can compute $L_I^W(p) = L_{I'}^W(p_{I'})$. Let $\mathbf{B} \in \mathbb{R}^{([n]\setminus I)\times I'}$ be the matrix sending any $q \in \pi_{I'}(W)$m to the corresponding vector $(L_{I'}^W(q))_{[n]\setminus I}$. The column $\mathbf{B}_i$ can be computed as $(L_{I'}^W(e_{I'}^i))_{[n]\setminus I}$ for $e_{I'}^i \in \mathbb{R}^{I'}$. We have $\|L_I^W(p)\|^2 = \|p\|^2 + \|(L_{I'}^W(p_{I'}))_{[n]\setminus I}\|^2 \leq \|p\|^2 + \|\mathbf{B}\|^2\|p_{I'}\|^2$ for any $p \in \pi_I(W)$, and so $\ell^W(I) = \sqrt{\|L_I^W\|^2 - 1} \leq \|B\|$. We upper bound the operator norm by the Frobenius norm as $\|\mathbf{B}\| \leq \|\mathbf{B}\|_F = \sqrt{\sum_{ji} \mathbf{B}_{ji}^2} \leq n \max_{ji} |\mathbf{B}_{ji}|$. By Lemma 3.3.9 it follows that $|\mathbf{B}_{ji}| = |(L_{I'}^W(e^i))_j| \leq \kappa_{ij}^W$. The algorithm returns the answer 'pass' if $n \max_{ji} |\mathbf{B}_{ji}| \leq \theta$ and 'fail' otherwise.

To implement the algorithm, we first need to select a minimal $I' \subset I$ such that $\dim(\pi_{I'}(W)) = \dim(\pi_I(W))$. This can be found by computing a matrix $\mathbf{M} \in \mathbb{R}^{n\times(n-m)}$ such that $\mathrm{im}(\mathbf{M}) = W$, and selecting a maximal number of linearly independent columns of $\mathbf{M}_{I,\cdot}$. Then, we compute the matrix

$B \in \mathbb{R}^{([n]\setminus I)\times I'}$ that implements the transformation $[L_{I'}^W]_{[n]\setminus I} : \pi_{I'}(W) \to \pi_{[n]\setminus I}(W)$. The algorithm returns the pair (i, j) corresponding to the entry maximizing $|B_{ji}|$. The running time analysis will be given in the proof of Lemma 4.3.15, together with an amortized analysis of a sequence of calls to the subroutine. □

Remark 4.2.3. We note that the algorithm Verify-Lift does not need to compute the circuit as in Lemma 3.3.9. The following observation will be important in the analysis: the algorithm returns the answer 'fail' even if $\ell^W(I) \le \theta < n|B_{ji}|$.

4.3 A scaling-invariant layered least squares interior-point algorithm

4.3.1 Preliminaries on interior-point methods

In this section, we introduce the standard definitions, concepts and results from the interior-point literature that will be required for our algorithm. We consider an LP problem in the form System 1.1, or equivalently in the subspace form System 1.2 for $W = \ker(\mathbf{A})$. We let

$$\mathcal{P}_{++} := \{ x \in \mathbb{R}^n : \mathbf{A}x = b, x > \mathbf{0} \}, \quad \mathcal{D}_{++} := \left\{ (y, s) \in \mathbb{R}^{m+n} : \mathbf{A}^\top y + s = c, s > \mathbf{0} \right\}. \tag{4.3}$$

Recall the central path defined in (CP), with $w(\mu) = (x(\mu), y(\mu), s(\mu))$ denoting the central path point corresponding to $\mu > 0$. We let $w^* = (x^*, y^*, s^*)$ denote the primal and dual optimal solutions to System 1.1 that correspond to the limit of the central path for $\mu \to 0$.

For a point $w = (x, y, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$, the *normalized duality gap* is $\mu(w) = x^\top s/n$.

The *ℓ_2-neighborhood of the central path with opening $\beta > 0$* is the set

$$\mathcal{N}(\beta) := \left\{ w \in \mathcal{P}_{++} \times \mathcal{D}_{++} : \left\| \frac{xs}{\mu(w)} - \mathbf{1} \right\| \le \beta \right\}$$

Throughout the chapter, we will assume β is chosen from $(0, 1/4]$; in Algorithm 4.2 we use the value $\beta = 1/8$. The following proposition gives a bound on the distance between w and $w(\mu)$ if $w \in \mathcal{N}(\beta)$. See e.g. [Gon92, Lemma 5.4], [MT03, Proposition 2.1].

Proposition 4.3.1. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$ and $\mu = \mu(w)$, and consider the central path point $w(\mu) = (x(\mu), y(\mu), s(\mu))$. For each $i \in [n]$,*

$$\frac{x_i}{1+2\beta} \le \frac{1-2\beta}{1-\beta} \cdot x_i \le x_i(\mu) \le \frac{x_i}{1-\beta}, \quad \text{and}$$
$$\frac{s_i}{1+2\beta} \le \frac{1-2\beta}{1-\beta} \cdot s_i \le s_i(\mu) \le \frac{s_i}{1-\beta}.$$

We will often use the following proposition which is immediate from definiton of β.

Proposition 4.3.2. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, and $\mu = \mu(w)$. Then for each $i \in [n]$*

$$(1-\beta)\sqrt{\mu} \le \sqrt{s_i x_i} \le (1+\beta)\sqrt{\mu}.$$

Proof. By definition of $\mathcal{N}(\beta)$ we have for all $i \in [n]$ that $|\frac{x_i s_i}{\mu} - 1| \le \|\frac{xs}{\mu} - \mathbf{1}\| \le \beta$ and so $(1-\beta)\mu \le x_i s_i \le (1+\beta)\mu$. Taking roots gives the results. □

A key property of the central path is *"near monotonicity"*, formulated in the following lemma, see [VY96, Lemma 16].

Lemma 4.3.3. *Let $w = (x, y, s)$ be a central path point for μ and $w' = (x', y', s')$ be a central path point for $\mu' \le \mu$. Then $\|x'/x + s'/s\|_\infty \le n$. Further, for the optimal solution $w^* = (x^*, y^*, s^*)$ corresponding to the central path limit $\mu \to 0$, we have $\|x^*/x\|_1 + \|s^*/s\|_1 = n$.*

Proof. We show that $\|x'/x\|_1 + \|s'/s\|_1 \le 2n$ for any feasible primal x' and dual (y', s') such that $(x')^\top s' \le x^\top s = n\mu$; this implies the first statement with the weaker bound $2n$. For the stronger bound $\|x'/x + s'/s\|_\infty \le n$, see the proof of [VY96, Lemma 16]. Since $x - x' \in W$ and $s - s' \in W^\perp$, we have $(x - x')^\top (s - s') = 0$. This can be rewritten as $x^\top s' + (x')^\top s = x^\top s + (x')^\top s'$. By our assumption on x' and s', the right-hand side is bounded by $2n\mu$. Dividing by μ, and noting that $x_i s_i = \mu$ for all $i \in [n]$, we obtain

$$\left\| \frac{x'}{x} \right\|_1 + \left\| \frac{s'}{s} \right\|_1 = \sum_{i=1}^{n} \frac{x'_i}{x_i} + \frac{s'_i}{s_i} \le 2n \,.$$

The second statement follows by using this to central path points (x', y', s') with parameter μ', and taking the limit $\mu' \to 0$. □

4.3.2 The affine scaling and layered-least-squares steps

Given $w = (x, y, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$, the search directions commonly used in interior-point methods are obtained as the solution $(\Delta x, \Delta y, \Delta s)$ to the following linear system for some $\sigma \in [0, 1]$.

$$\mathbf{A}\Delta x = \mathbf{0} \,, \tag{4.4}$$

$$\mathbf{A}^\top \Delta y + \Delta s = \mathbf{0} \,, \tag{4.5}$$

$$s\Delta x + x\Delta s = \sigma\mu e - xs \,. \tag{4.6}$$

Predictor-corrector methods, such as the Mizuno-Todd-Ye Predictor-Corrector (MTY P-C) algorithm [MTY93], alternate between two types of steps. In *predictor steps*, we use $\sigma = 0$. This direction is also called the *affine scaling direction*, and will be denoted as $\Delta w^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta y^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ throughout. In *corrector steps*, we use $\sigma = 1$. This gives the *centrality direction*, denoted as $\Delta w^{\mathrm{c}} = (\Delta x^{\mathrm{c}}, \Delta y^{\mathrm{c}}, \Delta s^{\mathrm{c}})$.

In the predictor steps, we make progress along the central path. Given the search direction on the current iterate $w = (x, y, s) \in \mathcal{N}(\beta)$, the step-length is chosen such that the line segment between the current and next steps remain in $\mathcal{N}(2\beta)$, i.e.,

$$\alpha^{\mathrm{a}} \le \sup\{\, \alpha \in [0, 1] : \forall \alpha' \in [0, \alpha] : w + \alpha' \Delta w^{\mathrm{a}} \in \mathcal{N}(2\beta) \,\}.$$

Thus, we obtain a point $w^+ = w + \alpha^{\mathrm{a}} \Delta w^{\mathrm{a}} \in \mathcal{N}(2\beta)$. The corrector step finds a next iterate $w^c = w^+ + \Delta w^{\mathrm{c}}$, where Δw^{c} is the centrality direction computed at w^+. The next proposition summarizes well-known properties, see e.g. [Ye97, Section 4.5.1].

Proposition 4.3.4. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$.*

(i) For the affine scaling step, we have $\mu(w^+) = (1 - \alpha)\mu(w)$.

(ii) The affine scaling step-length can be chosen as

$$\alpha^{\mathrm{a}} \ge \max\left\{ \frac{\beta}{\sqrt{n}}, 1 - \frac{\|\Delta x^{\mathrm{a}} \Delta s^{\mathrm{a}}\|}{\beta\mu(w)} \right\}.$$

(iii) *For $w^+ \in \mathcal{N}(2\beta)$, and $w^c = w^+ + \Delta w^c$, we have $\mu(w^c) = \mu(w^+)$ and $w^c \in \mathcal{N}(\beta)$.*

(iv) *After a sequence of $O(\sqrt{n}t)$ predictor and corrector steps, we obtain an iterate $w' = (x', y', s') \in \mathcal{N}(\beta)$ such that $\mu(w') \le \mu(w)/2^t$.*

Minimum norm viewpoint and residuals. For any point $w = (x, y, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$ we define

$$\delta := \delta(w) = s^{1/2} x^{-1/2} \in \mathbb{R}^n. \tag{4.7}$$

With this notation, we can write (4.6) for $\sigma = 0$ in the form

$$\delta \Delta x + \delta^{-1} \Delta s = -s^{1/2} x^{1/2}. \tag{4.8}$$

Note that for a point $w(\mu) = (x(\mu), y(\mu), s(\mu))$ on the central path, we have $\delta_i(w(\mu)) = s_i(\mu)/\sqrt{\mu} = \sqrt{\mu}/x_i(\mu)$ for all $i \in [n]$. From Proposition 4.3.1, we see that if $w \in \mathcal{N}(\beta)$, and $\mu = \mu(w)$, then for each $i \in [n]$,

$$\sqrt{1-2\beta} \cdot \delta_i(w(\mu)) \le \delta_i(w) \le \frac{1}{\sqrt{1-2\beta}} \cdot \delta_i(w(\mu)). \tag{4.9}$$

The matrix $\mathrm{diag}(\delta(w))$ will often be used for rescaling in the algorithm. That is, for the current iterate $w = (x, y, s)$ in the interior-point method, we will perform projections in the space $\mathrm{diag}(\delta(w))W$. To simplify notation, for $\delta = \delta(w)$, we use L_I^δ and κ_{ij}^δ as short-hands for $L_I^{\mathrm{diag}(\delta)W}$ and $\kappa_{ij}^{\mathrm{diag}(\delta)W}$. The subspace $W = \ker(\mathbf{A})$ will be fixed throughout.

It is easy to see from the optimality conditions that the components of the affine scaling direction $\Delta w^a = (\Delta x^a, \Delta y^a, \Delta s^a)$ are the optimal solutions of the following minimum-norm problems.

$$\begin{aligned} \Delta x^a &= \operatorname*{arg\,min}_{\Delta x \in \mathbb{R}^n} \left\{ \|\delta(x + \Delta x)\|^2 : \mathbf{A}\Delta x = \mathbf{0} \right\} \\ (\Delta y^a, \Delta s^a) &= \operatorname*{arg\,min}_{(\Delta y, \Delta s) \in \mathbb{R}^m \times \mathbb{R}^n} \left\{ \|\delta^{-1}(s + \Delta s)\|^2 : \mathbf{A}^\top \Delta y + \Delta s = \mathbf{0} \right\} \end{aligned} \tag{4.10}$$

Following [MT05], for a search direction $\Delta w = (\Delta x, \Delta y, \Delta s)$, we define the *residuals* as

$$Rx := \frac{\delta(x + \Delta x)}{\sqrt{\mu}}, \qquad Rs := \frac{\delta^{-1}(s + \Delta s)}{\sqrt{\mu}}. \tag{4.11}$$

We let Rx^a and Rs^a denote the residuals for the affine scaling direction Δw^a. Hence, the primal affine scaling direction Δx^a is the one that minimizes the ℓ_2-norm of the primal residual Rx^a, and the dual affine scaling direction $(\Delta y^a, \Delta s^a)$ minimizes the ℓ_2-norm of the dual residual Rs^a. The next lemma summarizes simple properties of the residuals, see [MT05].

Lemma 4.3.5. *For $\beta \in (0, 1/4]$ such that $w = (x, y, s) \in \mathcal{N}(\beta)$ and the affine scaling direction $\Delta w = (\Delta x^a, \Delta y^a, \Delta s^a)$, we have*

(i)

$$Rx^a Rs^a = \frac{\Delta x^a \Delta s^a}{\mu}, \quad Rx^a + Rs^a = \frac{x^{1/2} s^{1/2}}{\sqrt{\mu}}, \tag{4.12}$$

(ii)

$$\|Rx^a\|^2 + \|Rs^a\|^2 = n,$$

(iii) *We have $\|Rx^a\|, \|Rs^a\| \le \sqrt{n}$, and for each $i \in [n]$, $\max\{Rx_i^a, Rs_i^a\} \ge \frac{1}{2}(1 - \beta)$.*

(iv)

$$Rx^{\mathrm{a}} = -\frac{1}{\sqrt{\mu}}\delta^{-1}\Delta s^{\mathrm{a}}, \quad Rs^{\mathrm{a}} = -\frac{1}{\sqrt{\mu}}\delta\Delta x^{\mathrm{a}}\,.$$

Proof. Parts (i) and (iv) are immediate from the definitions and from (4.4)-(4.6) and (4.8). In part (ii), we use part (i) and $(Rx^{\mathrm{a}})^\top Rs^{\mathrm{a}} = 0$. In part, (iii), the first statement follows by part (ii), and the second statement follows from (i) and Proposition 4.3.2. □

For a subset $I \subset [n]$, we define

$$\epsilon_I^{\mathrm{a}}(w) := \max_{i\in I} \min\{|Rx_i^{\mathrm{a}}|, |Rs_i^{\mathrm{a}}|\}\,, \quad \text{and} \quad \epsilon^{\mathrm{a}}(w) := \epsilon_{[n]}^{\mathrm{a}}(w)\,. \tag{4.13}$$

The next claim shows that for the affine scaling direction, a small $\epsilon(w)$ yields a long step; see [MT05, Lemma 2.5].

Lemma 4.3.6. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$. Then the affine scaling step can be chosen such that*

$$\frac{\mu(w + \alpha^{\mathrm{a}}\Delta w^{\mathrm{a}})}{\mu(w)} \le \min\left\{1 - \frac{\beta}{\sqrt{n}}, \frac{2\sqrt{n}\epsilon^{\mathrm{a}}(w)}{\beta}\right\}.$$

Proof. Let $\epsilon := \epsilon^{\mathrm{a}}(w)$. From Lemma 4.3.5(i), we get $\|\Delta x^{\mathrm{a}}\Delta s^{\mathrm{a}}\|/\mu = \|Rx^{\mathrm{a}}Rs^{\mathrm{a}}\|$. We can bound $\|Rx^{\mathrm{a}}Rs^{\mathrm{a}}\| \le \epsilon(\|Rx^{\mathrm{a}}\| + \|Rs^{\mathrm{a}}\|) \le 2\epsilon\sqrt{n}$, where the latter inequality follows by Lemma 4.3.5(iii). From Proposition 4.3.4(ii), we get $\alpha^{\mathrm{a}} \ge \max\{\beta/\sqrt{n}, 1 - 2\sqrt{n}\epsilon/\beta\}$. The claim follows by part (i) of the same proposition. □

The layered-least-squares direction

Let $\mathcal{J} = (J_1, J_2, \ldots, J_p)$ be an *ordered partition* of $[n]$.[1] For $k \in [p]$, we use the notations $J_{<k} := J_1 \cup \ldots \cup J_{k-1}$, $J_{>k} := J_{k+1} \cup \ldots \cup J_p$, and similarly $J_{\le k}$ and $J_{\ge k}$. We will also refer to the sets J_k as *layers*, and $\mathcal{J}$ as a *layering*. Layers with lower indices will be referred to as 'higher' layers.

Given $w = (x, y, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$, and the layering $\mathcal{J}$, the LLS is defined as follows. For the primal direction, we proceed backwards, with $k = p, p-1, \ldots, 1$. Assume the components on the lower layers $\Delta x^{\mathrm{ll}}_{J_{>k}}$ have already been determined. We define the components in J_k as the coordinate projection $\Delta x^{\mathrm{ll}}_{J_k} = \pi_{J_k}(X_k)$, where the affine subspace X_k is defined as the set of minimizers

$$X_k := \operatorname*{arg\,min}_{\Delta x\in\mathbb{R}^n}\left\{ \left\|\delta_{J_k}(x_{J_k} + \Delta x_{J_k})\right\|^2 : \mathbf{A}\Delta x = \mathbf{0}, \Delta x_{J_{>k}} = \Delta x^{\mathrm{ll}}_{J_{>k}} \right\}. \tag{4.14}$$

The dual direction Δs^{ll} is determined in the forward order of the layers $k = 1, 2, \ldots, p$. Assume we already fixed the components $\Delta s^{\mathrm{ll}}_{J_{<k}}$ on the higher layers. Then, $\Delta s^{\mathrm{ll}}_{J_k} = \pi_{J_k}(S_k)$ for

$$S_k = \operatorname*{arg\,min}_{\Delta s\in\mathbb{R}^n}\left\{ \left\|\delta_{J_k}^{-1}(s_{J_k} + \Delta s_{J_k})\right\|^2 : \exists y \in \mathbb{R}^m, \mathbf{A}^\top\Delta y + \Delta s = \mathbf{0}, \Delta s_{J_{<k}} = \Delta s^{\mathrm{ll}}_{J_{<k}} \right\}. \tag{4.15}$$

The component Δy^{ll} is obtained as the optimal Δy for the final layer $k = p$. We use the notation Rx^{ll} and $\varepsilon^{\mathrm{ll}}(w)$ analogously to the affine scaling direction. This search direction was first introduced in [VY96].

The affine scaling direction is a special case for the single element partition. In this case, the definitions (4.14) and (4.15) coincide with those in (4.10).

[1] In contrast to how ordered partitions were defined in [MT05], we use the term *ordered* only to the p-tuple $(J_1, \ldots, J_p)$, which is to be viewed independently of δ.

4.3.3 Overview of ideas and techniques

A key technique in the analysis of layered least-squares algorithms [LMT09; MT03; VY96] is to argue about variables that have 'converged'. According to Proposition 4.3.1 and Lemma 4.3.3, for any iterate $w = (x, y, s) \in \mathcal{N}(\beta)$ and the limit optimal solution $w^* = (x^*, y^*, s^*)$, the bounds $x_i^* \le O(n)x_i$ and $s_i^* \le O(n)s_i$ hold. We informally say that x_i (or s_i) has *converged* if $x_i \le O(n)x_i^*$ ($s_i \le O(n)s_i^*$) hold for the current iterate. Thus, the value of x_i (or s_i) remains within a multiplicative factor $O(n^2)$ for the rest of the algorithm. Note that if $\mu > \mu'$ and x_i has converged at μ, then $\frac{s_i(\mu')/s_i(\mu)}{\mu'/\mu} \in \left[\frac{1}{O(n^2)}, O(n^2)\right]$; thus, s_i keeps "shooting down" with the central path parameter.

Converged variables in the affine scaling algorithm. Let us start by showing that at any point of the algorithm, at least one primal or dual variable has converged.

Suppose for simplicity that our current iterate is exactly on the central path, i.e., that $xs = \mu e$. This assumption will be maintained throughout this overview. In this case, the residuals can be simply written as $Rx^a = (x + \Delta x^a)/x$, $Rs^a = (s + \Delta s^a)/s$. Recall from (4.10) that the affine scaling direction corresponds to minimizing the residuals Rx^a and Rs^a. From this choice, we see that

$$\left\|\frac{x^*}{x}\right\| \ge \left\|\frac{x + \Delta x^a}{x}\right\|, \quad \left\|\frac{s^*}{s}\right\| \ge \left\|\frac{s + \Delta s^a}{s}\right\|. \tag{4.16}$$

We have $\|Rx^a\|^2 + \|Rs^a\|^2 = n$ by Lemma 4.3.5(ii). Let us assume $\|Rx^a\|^2 \ge n/2$; thus, there exists a $i \in [n]$ such that $x_i^* \ge x_i/\sqrt{2}$. In other words, just by looking at the residuals, we get the guarantee that a primal or a dual variable has already converged. Based on the value of the residuals, we can guarantee this to be a primal or a dual variable, but cannot identify which particular x_i or s_i this might be.

For $\|Rx^a\|^2 \ge n/2$, a primal variable has already converged *before* performing the predictor and corrector steps. We now show that even if $\|Rx^a\|$ is small, a primal variable will have converged *after* a single iteration. From (4.16), we see that there is an index i with $x_i^*/x_i \ge \|Rx^a\|/\sqrt{n}$.

Furthermore, Proposition 4.3.4(ii) and Lemma 4.3.5 imply that $1 - \alpha \le \|Rx^a\| \cdot \|Rs^a\|/\beta \le \sqrt{n}\|Rx^a\|/\beta$, since $\|Rs^a\| \le \sqrt{n}$. The predictor step moves to $x^+ := x + \alpha\Delta x^a = (1-\alpha)x + \alpha(x + \Delta x^a)$. Hence, $x^+ \le \left(\frac{\sqrt{n}\|Rx^a\|}{\beta} + \|Rx^a\|\right)x$. Putting the two inequalities together, we learn that $x_i^+ \le O(n)x_i^*$ for some $i \in [n]$. Since $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$, Proposition 4.3.1 implies that x_i will have converged after this iteration. An analogous argument proves that some s_j will also have converged after the iteration. We again emphasize that the argument only shows the existence of converged variables but we cannot identify them in general.

Measuring combinatorial progress. Tying the above together, we find that after a single affine scaling step, at least one primal variable x_i and at least one dual variable s_j has converged. This means that for any $\mu' < \mu$, $\frac{x_i(\mu')/x_j(\mu')}{x_i(\mu)/x_j(\mu)} \in \left[\frac{\mu}{O(n^4)\mu'}, \frac{O(n^4)\mu}{\mu'}\right]$; thus, the ratio of these variables keeps asymptotically increasing. The x_i/x_j ratios serve as the main progress measure in the Vavasis–Ye algorithm. If x_i/x_j is between $1/(\text{poly}(n)\bar{\chi})$ and $\text{poly}(n)\bar{\chi}$ before the affine scaling step for the pair of converged variables x_i and s_j, then after $\text{poly}(n)\log\bar{\chi}$ iterations, the x_i/x_j ratio must leave this interval and never return. Thus, we obtain a 'crossover-event' that cannot again occur for the same pair of variables. In the affine scaling algorithm, there is no guarantee that x_i/x_j falls in such a bounded interval for the converging variables x_i and s_j; in particular, we may obtain the same pairs of converged variables after each step.

The main purpose of layered-least-squares methods is to proactively force that in every certain number of iterations, some 'bounded' x_i/x_j ratios become 'large' and remain so for the rest of the algorithm.

In our approach, the first main insight is to focus on the scaling invariant quantities $\kappa^W_{ij} x_i/x_j$ instead. For simplicity's sake, we first present the algorithm with the assumption that all values κ^W_{ij} are known. We will then explain how this assumption can be removed by using gradually improving estimates on the values.

The combinatorial progress will be observed in the long edge graph. For a primal-dual feasible point $w = (x, y, s)$ and $\sigma = 1/O(n^6)$, this is defined as $G_{w,\sigma} = ([n], E_{w,\sigma})$ with edges (i, j) such that $\kappa^W_{ij} x_i/x_j \geq \sigma$. Observe that for any $i, j \in [n]$, at least one of (i, j) and (j, i) are long edges: this follows since for any circuit C with $i, j \in C$, we get lower bounds $|g^C_j/g^C_i| \leq \kappa^W_{ij}$ and $|g^C_i/g^C_j| \leq \kappa^W_{ji}$.

Intuitively, our algorithm will enforce the following two types of events. The analysis in Section 4.4 is based on a potential function analysis capturing roughly the same progress.

- For an iterate w and a value $\mu > 0$, we have $i, j \in [n]$ in a strongly connected component in $G_{w,\sigma}$ of size $\leq \tau$, and for any iterate w' with $\mu(w') > \mu$, if i, j are in a strongly connected component of $G_{w',\sigma}$ then this component has size $\geq 2\tau$.
- For an iterate w and a value $\mu > 0$, we have $(i, j) \notin E_{w,\sigma}$, and for any iterate w' with $\mu(w') > \mu$ we have $(i, j) \in E_{w',\sigma}$.

At most $O(n^2 \log n)$ such events can happen overall, so if we can prove that on average an event will happen every $O(\sqrt{n} \log(\bar{\chi}^*_A + n))$ iterations or the algorithm terminates, then we have the desired convergence bound of $O(n^{2.5} \log(n) \log(\bar{\chi}^*_A + n))$ iterations.

Converged variables cause combinatorial progress. We now show that combinatorial progress as above must happen in the affine scaling step in the case when the graph $G_{w,\sigma}$ is strongly connected. As noted above, for the pair of converged variables x_i and s_j after the affine scaling step, x_i/x_j, and thus $\kappa^W_{ij} x_i/x_j$, will asymptotically increase by a factor 2 in every $O(\sqrt{n})$ iterations.

By the strong connectivity assumption, there is a directed path in the long edge graph from i to j of length at most $n - 1$. Each edge has length at least σ, and by the cycle characterization (Theorem 3.4.1) we know that $(\kappa^W_{ji} x_j/x_i) \cdot \sigma^{n-1} \leq (\kappa^*_W)^n$. As such, $\kappa^W_{ji} x_j/x_i \leq (\kappa^*_W)^n/\sigma^{n-1}$. Since $\kappa^W_{ij} \kappa^W_{ji} \geq 1$, we obtain the lower bound $\kappa^W_{ij} x_i/x_j \geq \sigma^{n-1}/(\kappa^*_W)^n$.

This means that after $O(\sqrt{n} \log((\kappa^*_W/\sigma)^n)) = O(n^{1.5} \log(\kappa^*_W + n))$ affine scaling steps, the weight of the edge (i, j) will be more than $(\kappa^*_W/\sigma)^{4n}$. There can never again be a length n or shorter path from j to i in the long edge graph, for otherwise the resulting cycle would violate Theorem 3.4.1. Moreover, by the triangle inequality (Theorem 3.2.22), any other $k \neq i, j$ will have either (i, k) or (k, j) of length at least $(\kappa^*_W/\sigma)^{2n}$, similarly causing a pair of variables to never again be in the same connected component. As such, we took $O(n^{1.5} \log(\kappa^*_W + n))$ affine scaling steps and in that time at least $n - 1$ combinatorial progress events have occured.

The layered least squares step. Similarly to the Vavasis–Ye algorithm [VY96] and subsequent literature, our algorithm is a predictor-corrector method using LLS steps as in Section 4.3.2 for certain predictor iterations. Our algorithm (Algorithm 4.2) uses LLS steps only sometimes, and most steps are the simpler affine scaling steps; but for simplicity of this overview, we can assume every predictor iteration uses an LLS step.

We define the ordered partition $\mathcal{J} = (J_1, J_2, \ldots, J_p)$ corresponding to the strongly connected components in topological ordering. Recalling that either (i, j) or (j, i) is a long edge for every pair

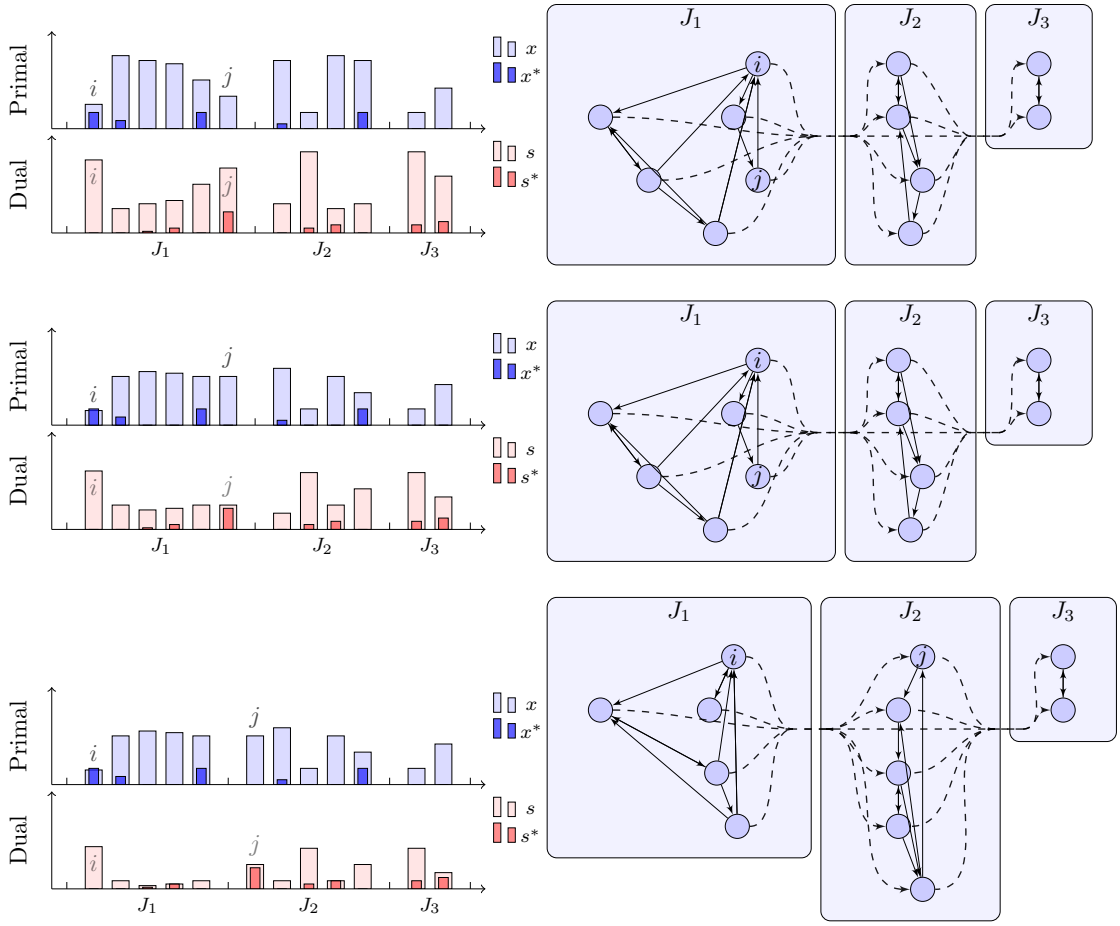


Figure 4.1: Top-down we have a chart of primal/dual variables and the *estimated* subgraph of the circuit ratio digraph (Definition 4.3.11) for three different iterations: *(i)* All variables except x_i are far away from their optimal values. *(ii)* On J_1 there is a primal variable (i) and dual variable (j) that have converged, i.e. x_i is close to x_i^* and s_i is close to s_i^*. *(iii)* j moves to layer J_2 due to a change in the underlying subgraph of the circuit ratio digraph.

$i, j \in [n]$, this order is unique and such that there is a complete directed graph of long edges from every J_k to $J_{k'}$ for $1 \le k < k' \le p$.

The first important property of the LLS step is that it is very close to the affine scaling step. In Section 4.3.4, we introduce the partition lifting cost $\ell^W(\mathcal{J}) = \max_{2\le k\le p} \ell^W(J_{\ge k})$ as the cost of lifting from lower to higher layers; we let $\ell^{1/x}(\mathcal{J})$ be a shorthand for $\ell^{\mathrm{diag}(1/x)W}(\mathcal{J})$. Note that this same rescaling is used for the affine scaling step in (4.10), since $\delta = \sqrt{\mu}/x$ if w is on the central path. In Lemma 4.3.10(ii), we show that for a small partition lifting cost, the LLS residuals will remain near the affine scaling residuals. Namely,

$$\|Rx^{\mathrm{ll}} - Rx^{\mathrm{a}}\|, \|Rs^{\mathrm{ll}} - Rs^{\mathrm{a}}\| \le 6n^{3/2}\ell^{1/x}(\mathcal{J})\,.$$

Recall that the LLS residuals can be written as $Rx^{\mathrm{ll}} = (x + \Delta x^{\mathrm{ll}})/x$, $Rs^{\mathrm{ll}} = (s+\Delta s^{\mathrm{ll}})/s$ for a point on the central path. For $\mathcal{J}$ defined as above, Lemma 4.2.2 yields $\ell^{1/x}(\mathcal{J}) \le n \max_{i\in J_{>k}, j\in J_{\le k}, k\in[p]} \kappa_{ij}^W x_i/x_j$. This will be sufficiently small as this maximum is taken over 'short' edges (not in $E_{w,\sigma}$).

A second, crucial property of the LLS step is that it "splits" our LP into p separate LPs that have "negligible" interaction. Namely, the direction $(\Delta x^{\mathrm{ll}}_{J_k}, \Delta s^{\mathrm{ll}}_{J_k})$ will be very close to the affine scaling step obtained in the problem restricted to the subspace $W_{\mathcal{J},k} = \left\{ x_{J_k} : x \in W, x_{J_{>k}} = 0 \right\}$ (see Lemma 4.3.10(i)).

Since each component J_k is strongly connected in the long edge graph $G_{w,\sigma}$, if there is at least one primal x_i and dual s_j in J_k that have converged after the LLS step, we can use the above argument to show combinatorial progress regarding the $\kappa_{ij}^W x_i/x_j$ value (Lemma 4.4.3).

Exploiting the proximity between the LLS and affine scaling steps, Lemma 4.3.10(iv) gives a lower bound on the step size $\alpha \ge 1 - \frac{3\sqrt{n}}{\beta} \max_{i\in[n]} \min\{|Rx_i^{\mathrm{ll}}|, |Rs_i^{\mathrm{ll}}|\}$. Let J_k be the component where $\min\{\|Rx^{\mathrm{ll}}_{J_k}\|, \|Rs^{\mathrm{ll}}_{J_k}\|\}$ is the largest. Hence, the step size α can be lower bounded in terms of $\min\{\|Rx^{\mathrm{ll}}_{J_k}\|, \|Rs^{\mathrm{ll}}_{J_k}\|\}$.

The analysis now distinguishes two cases. Let $w^+ = w + \alpha\Delta s^{\mathrm{ll}}$ be the point obtained by the predictor LLS step. If the corresponding partition lifting cost $\ell^{1/x^+}(\mathcal{J})$ is still small, then a similar argument that has shown the convergence of primal and dual variables in the affine scaling step will imply that after the LLS step, at least one x_i and one s_j will have converged for $i, j \in J_k$. Thus, in this case we obtain the combinatorial progress (Lemma 4.4.4).

The remaining case is when $\ell^{1/x^+}(\mathcal{J})$ becomes large. In Lemma 4.4.5, we show that in this case a new edge will enter the long edge graph, corresponding to the second combinatorial event listed previously. Intuitively, in this case one layer "crashes" into another.

Refined estimates on circuit imbalances. In the above overview, we assumed the circuit imbalance values κ_{ij}^W are given, and thus the graph $G_{w,\sigma}$ is available. Whereas these quantities are difficult to compute, we can naturally work with lower estimates. For each $i, j \in [n]$ that are contained in a circuit together, we start with the lower bound $\hat{\kappa}_{ij}^W = |g_j^C/g_i^C|$ obtained for an arbitrary circuit C with $i, j \in C$. We use the graph $\hat{G}_{w,\sigma} = ([n], \hat{E}_{w,\sigma})$ corresponding to these estimates. We have that $\hat{E}_{w,\sigma} \subseteq E_{w,\sigma}$, but some long edges may be missing. We determine the partition $\mathcal{J}$ of the strongly connected components of $\hat{G}_{w,\sigma}$ and estimate the partition lifting cost $\ell^{1/x}(\mathcal{J})$. If this is below the desired bound, the argument works correctly. Otherwise, we can identify a pair i, j responsible for this failure. Namely, we find a circuit C with $i, j \in C$ such that $\hat{\kappa}_{ij}^W < |g_j^C/g_i^C|$. In this case, we update our estimate, and recompute the partition; this is described in Algorithm 4.1. At each LLS step, the number of updates is bounded by n, since every update leads to a decrease in the number of partition classes. This finishes the overview of the algorithm.

4.3.4 A linear system viewpoint of layered least squares

We now continue with the detailed exposition of our algorithm. We present an equivalent definition of the LLS step introduced in Section 4.3.2, generalizing the linear system (4.5)–(4.6). We use the subspace notation. With this notation, (4.5)–(4.6) for the affine scaling direction can be written as

$$s\Delta x^{\mathrm{a}} + x\Delta s^{\mathrm{a}} = -xs\,, \quad \Delta x^{\mathrm{a}} \in W\,, \quad \text{and} \quad \Delta s^{\mathrm{a}} \in W^{\perp}\,, \tag{4.17}$$

which is further equivalent to $\delta\Delta x^{\mathrm{a}} + \delta^{-1}\Delta s^{\mathrm{a}} = -x^{1/2}s^{1/2}$.

Given the layering $\mathcal{J}$ and $w = (x, y, s)$, for each $k \in [p]$ we define the subspaces

$$W_{\mathcal{J},k} \coloneqq \left\{ x_{J_k} : x \in W, x_{J_{>k}} = \mathbf{0} \right\} \quad \text{and} \quad W^{\perp}_{\mathcal{J},k} \coloneqq \left\{ x_{J_k} : x \in W^{\perp}, x_{J_{<k}} = \mathbf{0} \right\}.$$

It is easy to see that these two subspaces are orthogonal complements. Our next goal is to show that, analogously to (4.17), the primal LLS step Δx^{ll} is obtained as the unique solution to the linear system

$$\delta\Delta x^{\mathrm{ll}} + \delta^{-1}\Delta s = -x^{1/2}s^{1/2}\,, \quad \Delta x^{\mathrm{ll}} \in W\,, \quad \text{and} \quad \Delta s \in W^{\perp}_{\mathcal{J},1} \times \cdots \times W^{\perp}_{\mathcal{J},p}\,, \tag{4.18}$$

and the dual LLS step Δs^{ll} is the unique solution to

$$\delta\Delta x + \delta^{-1}\Delta s^{\mathrm{ll}} = -x^{1/2}s^{1/2}\,, \quad \Delta x \in W_{\mathcal{J},1} \times \cdots \times W_{\mathcal{J},p}\,, \quad \text{and} \quad \Delta s^{\mathrm{ll}} \in W^{\perp}\,. \tag{4.19}$$

It is important to note that Δs in (4.18) may be different from Δs^{ll}, and Δx in (4.19) may be different from Δx^{ll}. In fact, $\Delta s^{\mathrm{ll}} = \Delta s$ and $\Delta x^{\mathrm{ll}} = \Delta x$ can only be the case for the affine scaling step.

The following lemma proves that the above linear systems are indeed uniquely solved by the LLS step.

Lemma 4.3.7. *For $t \in \mathbb{R}^n$, $W \subseteq \mathbb{R}^n$, $\delta \in \mathbb{R}^n_{++}$, and $\mathcal{J} = (J_1, J_2, \ldots, J_p)$, let $w = \mathrm{LLS}^{W,\delta}_{\mathcal{J}}(t)$ be defined by*

$$\delta w + \delta^{-1} v = \delta t, \qquad w \in W, \qquad v \in W^{\perp}_{\mathcal{J},1} \times \cdots \times W^{\perp}_{\mathcal{J},p}.$$

Then $\mathrm{LLS}^{W,\delta}_{\mathcal{J}}(t)$ is well-defined and

$$\left\|\delta_{J_k}(t_{J_k} - w_{J_k})\right\| = \min\left\{ \left\|\delta_{J_k}(t_{J_k} - z_{J_k})\right\| : z \in W, z_{J_{>k}} = w_{J_{>k}} \right\}$$

for every $k \in [p]$.

In the notation of the above lemma we have, for ordered partitions $\mathcal{J} = (J_1, J_2, \ldots, J_p)$, $\bar{\mathcal{J}} = (J_p, J_{p-1}, \ldots, J_1)$, and $(x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$ with $\delta = s^{1/2}x^{-1/2}$, that $\Delta x^{\mathrm{ll}} = \mathrm{LLS}^{W,\delta}_{\mathcal{J}}(-x)$ and $\Delta s^{\mathrm{ll}} = \mathrm{LLS}^{W^{\perp},\delta^{-1}}_{\bar{\mathcal{J}}}(-s)$.

Proof of Lemma 4.3.7. We first prove the equality $W \cap (W^{\perp}_{\mathcal{J},1} \times \cdots \times W^{\perp}_{\mathcal{J},p}) = \{\mathbf{0}\}$, and by a similar argument we have $W^{\perp} \cap (W_{\mathcal{J},1} \times \cdots \times W_{\mathcal{J},p}) = \{0\}$. By duality, this last equality tells us that

$$(W^{\perp} \cap (W_{\mathcal{J},1} \times \cdots \times W_{\mathcal{J},p}))^{\perp} = W + (W^{\perp}_{\mathcal{J},1} \times \cdots \times W^{\perp}_{\mathcal{J},p}) = \mathbb{R}^n.$$

Thus, the linear decomposition defining $\mathrm{LLS}^{W,\delta}_{\mathcal{J}}(t)$ has a solution and its solution is unique.

Suppose $y \in W \cap (W^{\perp}_{\mathcal{J},1} \times \cdots \times W^{\perp}_{\mathcal{J},p})$. We prove $y_{J_k} = \mathbf{0}$ by induction on k, starting at $k = p$. The induction hypothesis is that $y_{J_{>k}} = 0$, which is an empty requirement when $k = p$. The hypothesis $y_{J_{>k}} = 0$ together with the assumption $y \in W$ is equivalent to $y \in W \cap \mathbb{R}^n_{J_{\leq k}}$, and implies $y_{J_k} \in \pi_{J_k}(W \cap \mathbb{R}^n_{J_{\leq k}}) \coloneqq W_{\mathcal{J},k}$. Since we also have $y_{J_k} \in W^{\perp}_{\mathcal{J},k}$ by assumption, which is the orthogonal

complement of $W_{\mathcal{J},k}$, we must have $y_{J_k} = 0$. Hence, by induction $y = 0$. This finishes the proof that $\mathrm{LLS}^{W,\delta}_{\mathcal{J}}(t)$ is well-defined.

Next we prove that w is a minimizer of $\min\{\|\delta_{J_k}(t_{J_k} - z_{J_k})\| : z \in W, z_{J_{>k}} = w_{J_{>k}}\}$. The optimality condition is for $\delta_{J_k}(t_{J_k} - z_{J_k})$ to be orthogonal to $\delta_{J_k}u$ for any $u \in W_{\mathcal{J},k}$. By the LLS equation, we have $\delta_{J_k}(t_{J_k} - w_{J_k}) = \delta_{J_k}^{-1}v_{J_k}$, where $v_{J_k} \in W^{\perp}_{\mathcal{J},k}$. Noting then that $\langle \delta_{J_k}u, \delta_{J_k}^{-1}v\rangle = \langle u_{J_k}, v_{J_k}\rangle = 0$ for $u \in W_{\mathcal{J},k}$, the optimality condition follows immediately. □

With these tools, we can prove that the lifting costs are self-dual. This explains the reverse order in the dual vs primal LLS step and justifies our attention on the lifting cost in a self-dual algorithm. The next proposition generalizes the result of [GL97].

Proposition 4.3.8 (Proof on p. 83)**.** *For a linear subspace $W \subseteq \mathbb{R}^n$ and index set $I \subseteq [n]$ with $J = [n] \setminus I$,*

$$\|L^W_I\| \le \max\{1, \|L^{W^\perp}_J\|\}.$$

In particular, $\ell^W(I) = \ell^{W^\perp}(J)$.

We defer the proof to Section 4.5. Note that this proposition also implies Proposition 3.3.3(ii).

Partition lifting scores

A key insight is that if the layering $\mathcal{J}$ is "well-separated", then we indeed have $x\Delta s^{\mathrm{ll}} + s\Delta x^{\mathrm{ll}} \approx -xs$, that is, the LLS direction is close to the affine scaling direction. This will be shown in Lemma 4.3.10. The notion of "well-separatedness" can be formalized as follows. Recall the definition of the lifting score (4.2.1). The lifting score of the layering $\mathcal{J} = (J_1, J_2, \dots, J_p)$ of $[n]$ with respect to W is defined as

$$\ell^W(\mathcal{J}) := \max_{2\le k\le p} \ell^W(J_{\ge k}).$$

For $\delta \in \mathbb{R}^n_{++}$, we use $\ell^{W,\delta}(I) := \ell^{\mathrm{diag}(\delta)W}(I)$ and $\ell^{W,\delta}(\mathcal{J}) := \ell^{\mathrm{diag}(\delta)W}(\mathcal{J})$. When the context is clear, we omit W and write $\ell^{\delta}(I) := \ell^{W,\delta}(I)$ and $\ell^{\delta}(\mathcal{J}) := \ell^{W,\delta}(\mathcal{J})$.

The following important duality claim asserts that the lifting score of a layering equals the lifting score of the reverse layering in the orthogonal complement subspace. It is an immediate consequence of Proposition 4.3.8.

Lemma 4.3.9. *Let $W \subseteq \mathbb{R}^n$ be a linear subspace, $\delta \in \mathbb{R}^n_{++}$. For an ordered partition $\mathcal{J} = (J_1, J_2, \dots, J_p)$, let $\bar{\mathcal{J}} = (J_p, J_{p-1}, \dots, J_1)$ denote the reverse ordered partition. Then, we have*

$$\ell^{W,\delta}(\mathcal{J}) = \ell^{W^\perp,\delta^{-1}}(\bar{\mathcal{J}}).$$

Proof. Let $U = \mathrm{diag}(\delta)W$. Note that $U^\perp = \mathrm{diag}(\delta^{-1})W^\perp$. Then by Proposition 4.3.8, for $2 \le k \le p$, we have that

$$\ell^{W,\delta}(J_{\ge k}) = \ell^{U}(J_{\ge k}) = \ell^{U^\perp}(J_{\le k-1}) = \ell^{U^\perp}(\bar{J}_{\ge p-k+2}) = \ell^{W^\perp,\delta^{-1}}(\bar{J}_{\ge p-k+2}).$$

In particular, $\ell^{W,\delta}(\mathcal{J}) = \ell^{W^\perp,\delta^{-1}}(\bar{\mathcal{J}})$, as needed. □

The next lemma summarizes key properties of the LLS steps, assuming the partition has a small lifting score. We show that if $\ell^{\delta}(\mathcal{J})$ is sufficiently small, then on the one hand, the LLS step will be very close to the affine scaling step, and on the other hand, on each layer $k \in [p]$, it will be very close to the affine scaling step restricted to this layer for the subspace $W_{\mathcal{J},k}$. The proof is deferred to Section 4.5.

Lemma 4.3.10 (Proof on p. 87). *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, let $\mu = \mu(w)$ and $\delta = \delta(w)$. Let $\mathcal{J} = (J_1, \dots, J_p)$ be a layering with $\ell^\delta(\mathcal{J}) \le \beta/(32n^2)$, and let $\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$ denote the LLS direction for the layering $\mathcal{J}$. Let furthermore $\epsilon^{\mathrm{ll}}(w) = \max_{i\in[n]} \min\{|Rx_i^{\mathrm{ll}}|, |Rs_i^{\mathrm{ll}}|\}$, and define the maximal step length as*

$$\alpha^* := \sup\left\{ \alpha' \in [0,1] : \forall \bar{\alpha} \in [0, \alpha'] : w + \bar{\alpha}\Delta w^{\mathrm{ll}} \in \mathcal{N}(2\beta) \right\}.$$

Then the following properties hold.

(i) We have

$$\|\delta_{J_k}\Delta x^{\mathrm{ll}}_{J_k} + \delta^{-1}_{J_k}\Delta s^{\mathrm{ll}}_{J_k} + x^{1/2}_{J_k}s^{1/2}_{J_k}\| \le 6n\ell^\delta(\mathcal{J})\sqrt{\mu}\,, \quad \forall k \in [p], \text{ and} \tag{4.20}$$

$$\|\delta\Delta x^{\mathrm{ll}} + \delta^{-1}\Delta s^{\mathrm{ll}} + x^{1/2}s^{1/2}\| \le 6n^{3/2}\ell^\delta(\mathcal{J})\sqrt{\mu}\,. \tag{4.21}$$

(ii) For the affine scaling direction $\Delta w^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta y^{\mathrm{a}}, \Delta s^{\mathrm{a}})$,

$$\|Rx^{\mathrm{ll}} - Rx^{\mathrm{a}}\|, \|Rs^{\mathrm{ll}} - Rs^{\mathrm{a}}\| \le 6n^{3/2}\ell^\delta(\mathcal{J})\,.$$

(iii) For the residuals of the LLS steps we have $\|Rx^{\mathrm{ll}}\|, \|Rs^{\mathrm{ll}}\| \le \sqrt{2n}$. For each $i \in [n]$, $\max\{|Rx_i^{\mathrm{ll}}|, |Rs_i^{\mathrm{ll}}|\} \ge \frac{1}{2} - \frac{3}{4}\beta$.

(iv) We have

$$\alpha^* \ge 1 - \frac{3\sqrt{n}\epsilon^{\mathrm{ll}}(w)}{\beta}\,, \tag{4.22}$$

and for any $\alpha \in [0,1]$

$$\mu(w + \alpha\Delta w^{\mathrm{ll}}) = (1-\alpha)\mu\,,$$

(v) We have $\epsilon^{\mathrm{ll}}(w) = 0$ if and only if $\alpha^* = 1$. These are further equivalent to $w + \Delta w^{\mathrm{ll}} = (x + \Delta x^{\mathrm{ll}}, y + \Delta y^{\mathrm{ll}}, s + \Delta s^{\mathrm{ll}})$ being an optimal solution to (LP).

4.3.5 The layering procedure

Our algorithm performs LLS steps on a layering with a low lifting score. A further requirement is that within each layer, the circuit imbalances κ^δ_{ij} defined in Section 3.4 are suitably bounded. The rescaling here is with respect to $\delta = \delta(w)$ for the current iterate $w = (x, y, s)$. To define the precise requirement on the layering, we first introduce an auxiliary graph. Throughout we use the parameter

$$\gamma := \frac{\beta}{2^{10}n^5}\,. \tag{4.23}$$

The auxiliary graph. For a vector $\delta \in \mathbb{R}^n_{++}$ and $\sigma > 0$, we define the directed graph $G_{\delta,\sigma} = ([n], E_{\delta,\sigma})$ such that $(i,j) \in E_{\delta,\sigma}$ if $\kappa^\delta_{ij} \ge \sigma$. This is a subgraph of the *circuit ratio digraph* studied in Section 4.2, including only the edges where the circuit ratio is at least the threshold σ. Note that we do not have direct access to this graph, as we cannot efficiently compute the values κ^δ_{ij}.

At the beginning of the entire algorithm, we run the subroutine Find-Circuits(A) as in Theorem 3.4.6, where $W = \ker(\mathbf{A})$. We assume the matroid $\mathcal{M}(\mathbf{A})$ is non-separable. For a separable matroid, we can solve the subproblems of our LP on the components separately. Thus, for each $i \ne j$, $i, j \in [n]$, we obtain an estimate $\hat{\kappa}_{ij} \le \kappa_{ij}$. These estimates will be gradually improved throughout the algorithm.

Note that $\kappa^\delta_{ij} = \kappa_{ij}\delta_j/\delta_i$ and $\hat{\kappa}^\delta_{ij} = \hat{\kappa}_{ij}\delta_j/\delta_i$. If $\hat{\kappa}^\delta_{ij} \ge \sigma$, then we are guaranteed $(i,j) \in E_{\delta,\sigma}$.

Definition 4.3.11. *Define $\hat{G}_{\delta,\sigma} = ([n], \hat{E}_{\delta,\sigma})$ to be the directed graph with edges (i,j) such that $\hat{\kappa}^{\delta}_{ij} \geq \sigma$; $\hat{G}_{\delta,\sigma}$ is a subgraph of $G_{\delta,\sigma}$.*

Lemma 4.3.12. *Let $\delta \in \mathbb{R}^n_{++}$. For every $i \neq j$, $i,j \in [n]$, $\hat{\kappa}^{\delta}_{ij} \cdot \hat{\kappa}^{\delta}_{ji} \geq 1$. Consequently, for any $0 < \sigma \leq 1$, at least one of $(i,j) \in \hat{E}_{\delta,\sigma}$ or $(j,i) \in \hat{E}_{\delta,\sigma}$.*

Proof. We show that this property holds at the initialization. Since the estimates can only increase, it remains true throughout the algorithm. Recall the definition of $\hat{\kappa}_{ij}$ from Theorem 3.4.6. This is defined as the maximum of $|g_j/g_i|$ such that $g \in W$, $\mathrm{supp}(g) = C$ for some $C \in \hat{C}$ containing i and j. For the same vector g, we get $\hat{\kappa}_{ji} \geq |g_i/g_j|$. Consequently, $\hat{\kappa}_{ij} \cdot \hat{\kappa}_{ji} \geq 1$, and also $\hat{\kappa}^{\delta}_{ij} \cdot \hat{\kappa}^{\delta}_{ji} \geq 1$. The second claim follows by the assumption $\sigma \leq 1$. □

Balanced layerings. We are ready to define the requirements on the layering in the algorithm. In the algorithm, $\delta = \delta(w)$ will correspond to the scaling of the current iterate $w = (x,y,s)$.

Definition 4.3.13. *Let $\delta \in \mathbb{R}^n_{++}$. The layering $\mathcal{J} = (J_1, J_2, \ldots, J_p)$ of $[n]$ is δ-*balanced *if*

(i) $\ell^{\delta}(\mathcal{J}) \leq \gamma$, and

(ii) J_k is strongly connected in $G_{\delta,\gamma/n}$ for all $k \in [p]$.

The following lemma shows that within each layer, the κ^{δ}_{ij} values are within a bounded range. This will play an important role in our potential analysis.

Lemma 4.3.14. *Let $0 < \sigma < 1$ and $t > 0$, and $i,j \in [n]$, $i \neq j$.*

(i) If the graph $G_{\delta,\sigma}$ contains a directed path of at most $t-1$ edges from j to i, then

$$\kappa^{\delta}_{ij} < \left(\frac{\kappa^*}{\sigma}\right)^t .$$

(ii) If $G_{\delta,\sigma}$ contains a directed path of at most $t-1$ edges from i to j, then

$$\kappa^{\delta}_{ij} > \left(\frac{\sigma}{\kappa^*}\right)^t .$$

Proof. For part (i), let $j = i_1, i_2, \ldots, i_h = i$ be a path in $G_{\delta,\sigma}$ in J from j to i with $h \leq t$. That is, $\kappa^{\delta}_{i_\ell i_{\ell+1}} \geq \sigma$ for each $\ell \in [h]$. Theorem 3.4.1 yields

$$(\bar{\kappa}^*)^t \geq \kappa^{\delta}_{ij} \cdot \sigma^{h-1} > \kappa^{\delta}_{ij} \cdot \sigma^t ,$$

since $h \leq t$ and $\sigma < 1$. Part (ii) follows using part (i) for j and i, and that $\kappa^{\delta}_{ij} \cdot \kappa^{\delta}_{ji} \geq 1$ according to Lemma 4.3.12. □

Description of the layering subroutine. Consider an iterate $w = (x,y,s) \in \mathcal{N}(\beta)$ of the algorithm with $\delta = \delta(w)$, The subroutine Layering$(\delta, \hat{\kappa})$, described in Algorithm 4.1, constructs a δ-balanced layering. We recall that the approximated auxilliary graph $\hat{G}_{\delta,\gamma/n}$ with respect to $\hat{\kappa}$ is as in Definition 4.3.11

We now give an overview of the subroutine Layering$(\delta, \hat{\kappa})$. We start by computing the strongly connected components (SCCs) of the directed graph $\hat{G}_{\delta,\gamma/n}$. The edges of this graph are obtained using the current estimates $\hat{\kappa}^{\delta}_{ij}$. According to Lemma 4.3.12, we have $(i,j) \in \hat{E}_{\delta,\gamma/n}$ or $(j,i) \in \hat{E}_{\delta,\gamma/n}$ for every $i,j \in [n]$, $i \neq j$. Hence, there is a linear ordering of the components $C_1, C_2, \ldots, C_\ell$ such that $(u,v) \in \hat{E}_{\delta,\gamma/n}$ whenever $u \in C_i$, $v \in C_j$, and $i < j$. We call this the ordering imposed by $\hat{G}_{\delta,\gamma/n}$.

Algorithm 4.1: Layering($\delta, \hat{\kappa}$)

Input : $\delta \in \mathbb{R}^n_{++}$ and $\hat{\kappa} \in \mathbb{R}^E_{++}$.
Output : δ-balanced layering $\mathcal{J} = (J_1, \dots, J_p)$ and updated values $\hat{\kappa} \in \mathbb{R}^E_{++}$.

Compute the strongly connected components $C_1, C_2, \dots, C_\ell$ of $\hat{G}_{\delta,\gamma/n}$, listed in the ordering imposed by $\hat{G}_{\delta,\gamma/n}$
$\bar{E} \leftarrow \hat{E}_{\delta,\gamma/n}$
for $k = 2, \dots, \ell$ **do**
  Call Verify-Lift($\mathrm{diag}(\delta)W, C_{\geq k}, \gamma$) that answers *'pass'* or *'fail'*
  if *the answer is* 'fail' **then**
    Let $i \in C_{\geq k}$, $j \in C_{<k}$, and t be the output of Verify-Lift such that $\gamma/n \leq t \leq \kappa^\delta_{ij}$
    $\hat{\kappa}_{ij} \leftarrow t\delta_i/\delta_j$
    $\bar{E} \leftarrow \bar{E} \cup \{(i,j)\}$
Compute strongly connected components $J_1, J_2, \dots, J_p$ of $([n], \bar{E})$, listed in the ordering imposed by $\hat{G}_{\delta,\gamma/n}$
return $\mathcal{J} = (J_1, J_2, \dots, J_p)$, $\hat{\kappa}$.

Next, for each $k = 2, \dots, \ell$, we use the subroutine Verify-Lift($\mathrm{diag}(\delta)W, C_{\geq k}, \gamma$) described in Lemma 4.2.2. If the subroutine returns 'pass', then we conclude $\ell^\delta(C_{\geq k}) \leq \gamma$, and proceed to the next layer. If the answer is 'fail', then the subroutine returns as certificates $i \in C_{\geq k}$, $j \in C_{<k}$, and t such that $\gamma/n \leq t \leq \kappa^\delta_{ij}$. In this case, we update $\hat{\kappa}^\delta_{ij}$ to the higher value t. We add (i,j) to an edge set $\bar{E}$; this edge set was initialized to contain $\hat{E}_{\delta,\gamma/n}$. After adding (i,j), all components C_ℓ between those containing i and j will be merged into a single strongly connected component. To see this, recall that if $i' \in C_\ell$ and $j' \in C_{\ell'}$ for $\ell < \ell'$, then $(i', j') \in \hat{E}_{\delta,\gamma/n}$ according to Lemma 4.3.12.

Finally, we compute the strongly connected components of $([n], \bar{E})$. We let $J_1, J_2, \dots, J_p$ denote their unique acyclic order, and return these layers.

Lemma 4.3.15. *The subroutine* Layering($\delta, \hat{\kappa}$) *returns a δ-balanced layering in $O(nm^2 + n^2)$ time.*

The difficult part of the proof is showing the running time bound. We note that the weaker bound $O(n^2m^2)$ can be obtained by a simpler argument.

Proof. We first verify that the output layering is indeed δ-balanced. For property (i) of Definition 4.3.13, note that each J_q component is the union of some of the C_k's. In particular, for every $q \in [p]$, the set $J_{\geq q} = C_{\geq k}$ for some $k \in [\ell]$. Assume now $\ell^\delta(C_{\geq k}) > \gamma$. At step k of the main cycle, the subroutine Verify-Lift returned the answer 'fail', and a new edge $(i,j) \in E$ was added with $i \in C_{\geq k}$, $j \in C_{<k}$. Note that we already had $(j,i) \in \hat{E}_{\delta,\gamma/n}$, since $j \in C_r$ for some $r < k$, and $i \in C_{r'}$ for $r' \geq k$. This contradicts the choice of $J_{\geq q}$ as a maximal strongly connected component in $([n], E)$.

Property (ii) follows since all new edges added to E have $\kappa_{ij} \geq \gamma/n$. Therefore, $([n], E)$ is a subgraph of $G_{\delta,\gamma/n}$.

Let us now turn to the computational cost. The initial strongly-connected components can be obtained in time $O(n^2)$, and the same bound holds for the computation of the final components. (The latter can be also done in linear time, exploiting the special structure that the components C_i have a complete linear ordering.)

The second computational bottleneck is the subroutine Verify-Lift. We assume a matrix $\mathbf{M} \in \mathbb{R}^{n\times(n-m)}$ is computed at the very beginning such that $\mathrm{im}(\mathbf{M}) = W$. We first explain how to implement one call to Verify-Lift in $O(n(n-m)^2)$ time. We then sketch how to amortize the work across the different calls to Verify-Lift, using the nested structure of the layering, to implement the whole procedure in $O(n(n-m)^2)$ time. To turn this into $O(nm^2)$, we recall that the layering

procedure is the same for W and $W^\perp$ due to duality (Proposition 4.3.8). Since $\dim(W^\perp) = m$, applying this subroutine on $W^\perp$ instead of W achieves the same result but in time $O(nm^2)$.

We now explain the implementation of Verify-Lift, where we are given as input $C \subseteq [n]$ and the basis matrix $\mathbf{M} \in \mathbb{R}^{n\times(n-m)}$ as above with $\mathrm{im}(\mathbf{M}) = W$. The running time is dominated by the computation of the set $I \subseteq C$ and the matrix $\mathbf{B} \in \mathbb{R}^{([n]\setminus C)\times|I|}$ satisfying $L_C^W(x)_{[n]\setminus C} = \mathbf{B}x_I$, for $x \in \pi_C(W)$. We explain how to compute I and $\mathbf{B}$ from $\mathbf{M}$ using column operations (note that these preserve the range). The valid choices for $I \subseteq C$ are in correspondence with maximal sets of linear independent rows of $\mathbf{M}_{C,\bullet}$, noting then that $|I| = r$ where $r := \mathrm{rk}(M_{C,\bullet})$. Let $D_1 = [n-m-r]$ and $D_2 = [n-m] \setminus [n-m-r]$. By applying columns operations to $\mathbf{M}$, we can compute $I \subseteq C$ such that $\mathbf{M}_{I,D_2} = \mathbf{I}_r$ ($r \times r$ identity) and $\mathbf{M}_{C,D_1} = \mathbf{0}$. This requires $O(n(n-m)|C|)$ time using Gaussian elimination. At this point, note that $\pi_C(W) = \mathrm{im}(\mathbf{M}_{C,D_2})$, $\pi_I(W) = \mathbb{R}^I$ and $\mathrm{im}(\mathbf{M}_{\bullet,D_1}) = W \cap \mathbb{R}^n_{[n]\setminus C}$. To compute $\mathbf{B}$, we must transform the columns of $\mathbf{M}_{\bullet,D_2}$ into minimum norm lifts of $e^i \in \pi_I(W)$ into W, for all $i \in I$. For this purpose, it suffices to make the columns of $\mathbf{M}_{[n]\setminus C,D_2}$ orthogonal to the range of $\mathbf{M}_{[n]\setminus C,D_1}$. Applying Gram-Schmidt orthogonalization, this requires $O((n-|C|)(n-m)(n-m-r))$ time. From here, the desired matrix $\mathbf{B} = \mathbf{M}_{[n]\setminus C,D_2}$. Thus, the total running time of Verify-Lift is $O(n(n-m)|C| + (n-|C|)(n-m)(n-m-r)) = O(n(n-m)^2)$.

We now sketch how to amortize the work of all the calls of Verify-Lift during the layering algorithm, to achieve a total $O(n(n-m)^2)$ running time. Let $C_1, \ldots, C_\ell$ denote the candidate SCC layering. Our task is to compute the matrices $\mathbf{B}_k$, $2 \le k \le \ell$, needed in the calls to Verify-Lift on $W, C_{\ge k}$, $2 \le k \le \ell$, in total $O(n(n-m)^2)$ time. We achieve this in three steps working with the basis matrix $\mathbf{M}$ as above. Firstly, by applying column operations to $\mathbf{M}$, we compute sets $I_k \subseteq C_k$ and $D_k = [|I_{\le k}|] \setminus [|I_{<k}|]$, $k \in [\ell]$, such that $\mathbf{M}_{I_k,D_k} = \mathbf{I}_{r_k}$, where $r_k = |I_k|$, and $\mathbf{M}_{C_{\ge k},D_{<k}} = \mathbf{0}$, $2 \le k \le \ell$. Note that this enforces $\sum_{k=1}^{\ell} r_k = (n-m)$. This computation requires $O(n(n-m)^2)$ time using Gaussian elimination. This computation achieves $\mathrm{im}(\mathbf{M}_{C_k,D_k}) = \pi_{C_k}(W \cap \mathbb{R}^n_{C_{\le k}})$, $\mathrm{im}(\mathbf{M}_{C_{\ge k},D_{\ge k}}) = \pi_{C_{\ge k}}(W)$ and $\mathrm{im}(\mathbf{M}_{\bullet,D_{\le k}}) = W \cap \mathbb{R}^n_{C_{\le k}}$, for all $k \in [\ell]$.

From here, we block orthogonalize $\mathbf{M}$, such that the columns of $\mathbf{M}_{\bullet,D_k}$ are orthogonal to the range of $\mathbf{M}_{\bullet,D_{<k}}$, $2 \le k \le \ell$. Applying an appropriately adapted Gram-Schmidt orthogonalization, this requires $O(n(n-m)^2)$ time. Note that this operation maintains $\mathbf{M}_{I_k,D_k} = \mathbf{I}_{r_k}$, $k \in [\ell]$, since $\mathbf{M}_{C_{\ge k},D_{<k}} = 0$. At this point, for $k \in [\ell]$ the columns of $\mathbf{M}_{\bullet,D_k}$ are in correspondence with minimum norm lifts of $e^i \in \pi_{D_{\ge k}}(W)$ into W, for all $i \in I_k$. Note that to compute the matrix B_k we need the lifts of $e^i \in \pi_{D_{\ge k}}(W)$, for all $i \in I_{\ge k}$ instead of just $i \in I_k$.

We now compute the matrices $\mathbf{B}_\ell, \ldots, \mathbf{B}_2$ in this order via the following iterative procedure. Let k denote the iteration counter, which decrements from ℓ to 2. For $k = \ell$ (first iteration), we let $\mathbf{B}_\ell = \mathbf{M}_{C_{<\ell},D_\ell}$ and decrement k. For $k < \ell$, we eliminate the entries of $\mathbf{M}_{I_k,D_{>k}}$ by using the columns of $\mathbf{M}_{\bullet,D_k}$. We then let $\mathbf{B}_k = \mathbf{M}_{C_{<k},D_{\ge k}}$ and decrement k. To justify correctness, one only has to notice that at the end of iteration k, we maintain the orthogonality of $\mathbf{M}_{\bullet,D_{\ge k}}$ to the range of $\mathbf{M}_{\bullet,D_{<k}}$ and that $\mathbf{M}_{I_{\ge k},D_{\ge k}} = \mathbf{I}_{|I_{\ge k}|}$ is the appropriate identity. The cost of this procedure is the same as a full run of Gaussian elimination and thus is bounded by $O(n(n-m)^2)$. The calls to Verify-Lift during the layering procedure can thus be executed in $O(n(n-m)^2))$ amortized time as claimed. □

4.3.6 The overall algorithm

Algorithm 4.2 presents the overall algorithm LP-Solve($\mathbf{A}, b, c, w^0$). We assume that an initial feasible solution $w^0 = (x^0, y^0, s^0) \in \mathcal{N}(\beta)$ is given. We address this in Section 4.7, by adapting the extended system used in [VY96]. We note that this subroutine requires an upper bound on $\bar{\chi}^*$. Since computing $\bar{\chi}^*$ is hard, we can implement it by a doubling search on $\log \bar{\chi}^*$, as explained in Section 4.7. Other than for initialization, the algorithm does not require an estimate on $\bar{\chi}^*$.

Algorithm 4.2: LP-Solve(A, b, c, w^0)

Input : $\mathbf{A} \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and an initial feasible solution $w^0 = (x^0, y^0, s^0) \in \mathcal{N}(1/8)$ to System 1.1.

Output : Optimal solution $w^* = (x^*, y^*, s^*)$ to (LP).

Call Find-Circuits($\mathbf{A}$) to obtain the lower bounds $\hat{\kappa}_{ij}$ for each $i, j \in [n]$, $i \neq j$
$k \leftarrow 0, \alpha \leftarrow 0$
repeat
  /* Predictor step */
  Compute affine scaling direction $\Delta w^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta y^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ for w
  if $\epsilon^{\mathrm{a}}(w) < 10n^{3/2}\gamma$ **then** // Recall $\epsilon^{\mathrm{a}}(w)$ defined in (4.13)
    $\delta \leftarrow (s^k)^{1/2}(x^k)^{-1/2}$
    $(\mathcal{J}, \hat{\kappa}) \leftarrow$Layering$(\delta, \hat{\kappa})$
    Compute Layered Least Squares direction $\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$ for the layering $\mathcal{J}$ and w
    $\Delta w \leftarrow \Delta w^{\mathrm{ll}}$
    $\alpha \leftarrow 1 - 24\sqrt{n}\epsilon^{\mathrm{ll}}(w)$ // As in Lemma 4.3.10(ii)
  else
    $\Delta w \leftarrow \Delta w^{\mathrm{a}}$
    $\alpha \leftarrow \min\{1/(8\sqrt{n}), 1 - 8\|\Delta x^{\mathrm{a}}\Delta s^{\mathrm{a}}\|/\mu(w)\}$ // As in Proposition 4.3.4(ii)
  $w' \leftarrow w^k + \alpha\Delta w$
  /* Corrector step */
  Compute centrality direction $\Delta w^{\mathrm{c}} = (\Delta x^{\mathrm{c}}, \Delta y^{\mathrm{c}}, \Delta s^{\mathrm{c}})$ for w'
  $w^{k+1} \leftarrow w' + \Delta w^{\mathrm{c}}$
  $k \leftarrow k + 1$
until $\mu(w^k) = 0$
return $w^k = (x^k, y^k, s^k)$.

The algorithm starts with the subroutine Find-Circuits($\mathbf{A}$) as in Theorem 3.4.6. The iterations are similar to the MTY Predictor-Corrector algorithm [MTY93]. The main difference is that certain affine scaling steps are replaced by LLS steps. In every predictor step, we compute the affine scaling direction, and consider the quantity $\epsilon^{\mathrm{a}}(w) = \max_{i\in[n]} \min\{|Rx_i^{\mathrm{a}}|, |Rs_i^{\mathrm{a}}|\}$. If this is above the threshold $10n^{3/2}\gamma$, then we perform the affine scaling step. However, in case $\epsilon^{\mathrm{a}}(w) < 10n^{3/2}\gamma$, we use the LLS direction instead. In each such iteration, we call the subroutine Layering$(\delta, \hat{\kappa})$ (Algorithm 4.1) to compute the layers, and we compute the LLS step for this layering.

Another important difference is that the algorithm does not require a final rounding step. It terminates with the exact optimal solution w^* once a predictor step is able to perform a full step with $\alpha = 1$.

Theorem 4.3.16. *For given $\mathbf{A} \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$, and an initial feasible solution $w^0 = (x^0, y^0, s^0) \in \mathcal{N}(1/8)$, Algorithm 4.2 finds an optimal solution to System 1.1 in $O(n^{2.5} \log n \log(\bar{\chi}^*_{\mathbf{A}} + n))$ iterations.*

Remark 4.3.17. Whereas using LLS steps enables us to give a strong bound on the total number of iterations, finding LLS directions has a significant computational overhead as compared to finding affine scaling directions. The layering $\mathcal{J}$ can be computed in time $O(nm^2)$ (Lemma 4.3.15), and the LLS steps also require $O(nm^2)$ time, see [MMT98; VY96]. This is in contrast to the computational cost $O(n^\omega)$ of an affine scaling direction. Here $\omega < 2.373$ is the matrix multiplication constant [Vas12].

We now sketch a possible approach to amortize the computational cost of the LLS steps over the sequence of affine scaling steps. It was shown in [MT05] that for the MTY P-C algorithm, the "bad" scenario between two crossover events amounts to a series of affine scaling steps where the progress in μ increases exponentially from every iteration to the next. This corresponds to the

term $O(\min\{n^2 \log\log(\mu_0/\eta), \log(\mu_0/\eta)\})$ in their running time analysis. Roughly speaking, such a sequence of affine scaling steps indicates that an LLS step is necessary.

Hence, we could observe these accelerating sequences of affine scaling steps, and perform an LLS step after we see a sequence of length $O(\log n)$. The progress made by these affine scaling steps offsets the cost of computing the LLS direction.

4.4 The potential function and the overall analysis

Let $\mu > 0$ and $\delta(\mu) = s(\mu)^{1/2}x(\mu)^{-1/2} = \sqrt{\mu}/x(\mu) = s(\mu)/\sqrt{\mu}$ correspond to the point on the central path and recall the definition of γ in (4.23). For $i, j \in [n]$, $i \neq j$, we define

$$\varrho^{\mu}(i,j) := \frac{\log \kappa_{ij}^{\delta(\mu)}}{\log(4n\kappa^*{}_W/\gamma)},$$

and the main potentials in the algorithm as

$$\Psi^{\mu}(i,j) := \max\left\{1, \min\left\{2n, \inf_{0<\mu'<\mu} \varrho^{\mu'}(i,j)\right\}\right\} \quad \text{and} \quad \Psi(\mu) := \sum_{i,j\in[n],i\neq j} \log \Psi^{\mu}(i,j).$$

The motivation for $\varrho^{\mu}(i,j)$ and $\Psi^{\mu}(i,j)$ comes from Lemma 4.3.14, using $\sigma = \gamma/(4n)$. Thus, $\log \kappa_{ij}^{\delta(\mu)}/\log(4n\kappa^*{}_W/\gamma)$ can be seen as a lower bound on the length of the shortest j–i path. Recall that the layers are defined as strongly connected components of $\hat{G}_{\delta,\gamma/n}$, which is a subgraph of $G_{\delta(\mu),\gamma/(4n)}$ (using the bound (4.9)). Consequently, whenever $\varrho^{\mu}(i,j) \geq n$, the nodes i and j cannot be in the same strongly connected component for the normalized duality gap μ. Thus, our potentials $\Psi^{\mu}(i,j)$ can be seen as fine-grained analogues of the crossover events analyzed in [MT03; MT05; VY96]: the definition of $\Psi^{\mu}(i,j)$ contains a minimization over $0 < \mu' < \mu$; therefore, $\Psi^{\mu}(i,j) > n$ implies that i and j may never appear on the same layer for any $\mu' \leq \mu$. On the other hand, these potentials are more fine-grained: even for $t < n$, if $\Psi^{\mu}(i,j) \geq t$ then whenever a layer contains both i and j for $\mu' \leq \mu$, this layer must have size $\geq t$.

By definition, for all pairs $(i,j) \in [n] \times [n]$ we have $\Psi^{\mu'}(i,j) \geq \Psi^{\mu}(i,j)$ for $0 < \mu' \leq \mu$; and we enforce $\Psi^{\mu}(i,j) \in [1, 2n]$. The upper bound can be imposed since values $\Psi^{\mu'}(i,j) \geq n$ do not yield any new information on the layering. Hence, the overall potential $\Psi(\mu)$ is between 0 and $O(n^2 \log n)$. The overall analysis in the proof of Theorem 4.3.16 divides the iterations into phases. In each phase, we can identify a set $J \subseteq [n]$, $|J| > 1$ arising as a layer or as the union of two layers in the LLS step at the beginning of the phase. We show that $\Psi^{\mu}(i,j)$ doubles for at least $|J| - 1$ pairs $(i,j) \in J \times J$ during the subsequent $O(\sqrt{n}|J| \log(\bar{\chi}^* + n))$ iterations; consequently, $\Psi(\mu)$ increases by at least $|J| - 1$ during these iterations. This leads to the overall iteration bound $O(n^{2.5} \log(n) \log(\bar{\chi}^* + n))$. In comparison, the crossover analysis would correspond to showing that within $O(n^{1.5} \log(\bar{\chi}^* + n))$ iterations, one of the $\Psi^{\mu}(i,j)$ values previously $< n$ becomes larger than n. The following statement formalizes the above mentioned properties of $\Psi^{\mu}(i,j)$.

Lemma 4.4.1. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$. Let $i, j \in [n]$, $i \neq j$, and let $\mu = \mu(w)$.*

1. *If $\hat{G}_{\delta,\gamma/n}$ contains a path from j to i of at most $t - 1$ edges, then $\varrho^{\mu}(i,j) < t$.*
2. *If $\hat{G}_{\delta,\gamma/n}$ contains a path from i to j of at most $t - 1$ edges, then $\varrho^{\mu}(i,j) > -t$.*
3. *If $\Psi^{\mu}(i,j) \geq t$, then in any $\delta(w')$-balanced layering, where $w' = (x', y', s') \in \mathcal{N}(\beta)$ with $\mu(w') \leq \mu$,*
   - *i and j cannot be together on a layer of size at most t, and*

- *j cannot be on a layer preceding the layer containing i.*

Proof. From (4.9), we see that for any i, j,

$$\hat{\kappa}_{ij}^{\delta} \le \kappa_{ij}^{\delta} \le (1-2\beta)^{-1}\kappa_{ij}^{\delta(\mu)} \le 4\kappa_{ij}^{\delta(\mu)} .$$

Consequently, $\hat{G}_{\delta,\gamma/n}$ is a subgraph of $G_{\delta(\mu),\gamma/(4n)}$. The statement now follows from Lemma 4.3.14 with $\sigma = \gamma/(4n)$. □

In what follows, we formulate four important lemmas crucial for the proof of Theorem 4.3.16. For the lemmas, we only highlight some key ideas here, and defer the full proofs to Section 4.6.

For a triple $w \in \mathcal{N}(\beta)$, Δw^{ll} refers to the LLS direction found in the algorithm, and Rx^{ll} and Rs^{ll} denote the residuals as in (4.11). For a subset $I \subset [n]$ recall the definition

$$\epsilon_I^{\mathrm{ll}}(w) := \max_{i\in I} \min\{|Rx_i^{\mathrm{ll}}|, |Rs_i^{\mathrm{ll}}|\} .$$

We introduce another important quantity ξ for the analysis:

$$\xi_I^{\mathrm{ll}}(w) := \min\{\|Rx_I^{\mathrm{ll}}\|, \|Rs_I^{\mathrm{ll}}\|\} \tag{4.24}$$

for a subset $I \subset [n]$. For a layering $\mathcal{J} = (J_1, J_2, \dots, J_p)$, we let

$$\xi_{\mathcal{J}}^{\mathrm{ll}}(w) = \max_{k\in[p]} \xi_{J_k}^{\mathrm{ll}}(w) .$$

The key idea of the analysis is to extract information about the optimal solution $w^* = (x^*, y^*, s^*)$ from the LLS direction. The first main lemma shows that if $\|Rx_{J_q}^{\mathrm{ll}}\|$ is large on some layer J_q, then for at least one index $i \in J_q$, $x_i^*/x_i \ge 1/\mathrm{poly}(n)$, i.e., the variable x_i has "converged". The analogous statement holds on the dual side for $\|Rs_{J_q}^{\mathrm{ll}}\|$ and an index $j \in J_q$.

Lemma 4.4.2 (Proof on p. 89)**.** *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$ and let $w^* = (x^*, y^*, s^*)$ be the optimal solution corresponding to $\mu^* = 0$ on the central path. Let further $\mathcal{J} = (J_1, \dots, J_p)$ be a $\delta(w)$-balanced layering (Definition 4.3.13), and let $\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$ be the corresponding LLS direction. Then the following statement holds for every $q \in [p]$:*

(i) There exists $i \in J_q$ such that

$$x_i^* \ge \frac{2x_i}{3\sqrt{n}} \cdot (\|Rx_{J_q}^{\mathrm{ll}}\| - 2\gamma n) . \tag{4.25}$$

(ii) There exists $j \in J_q$ such that

$$s_j^* \ge \frac{2s_j}{3\sqrt{n}} \cdot (\|Rs_{J_q}^{\mathrm{ll}}\| - 2\gamma n) . \tag{4.26}$$

We outline the main idea of the proof of part (i); part (ii) follows analogously using the duality of the lifting scores (Lemma 4.3.9). On layer q, the LLS step minimizes $\|\delta_{J_q}(x_{J_q} + \Delta x_{J_q})\|$, subject to $\Delta x_{J_{>q}} = \Delta x_{J_{>q}}^{\mathrm{ll}}$ and subject to existence of $\Delta x_{J_{<q}}$ such that $\Delta x \in W$. By making use of $\ell^{\delta(w)}(J_{>q}) \le \gamma$ due to $\delta(w)$-balancedness, we can show the existence of a point $z \in W + x^*$ such that $\|\delta_{J_q}(z_{J_q} - x_{J_q}^*)\|$ is small, and $z_{J_{>q}} = x_{J_{>q}} + \Delta x_{J_{>q}}^{\mathrm{ll}}$. By the choice of $\Delta x_{J_q}^{\mathrm{ll}}$, we have $\|\delta_{J_q} z_{J_q}\| \ge \|\delta_{J_q}(x_{J_q} + \Delta x_{J_q}^{\mathrm{ll}})\| = \sqrt{\mu}\|Rx_{J_q}^{\mathrm{ll}}\|$. Therefore, $\|\delta_{J_q} x_{J_q}^*/\sqrt{\mu}\|$ cannot be much smaller than $\|Rx_{J_q}^{\mathrm{ll}}\|$. Noting that $\delta_{J_q} x_{J_q}^*/\sqrt{\mu} \approx x_{J_q}^*/x_{J_q}$, we obtain a lower bound on x_i^*/x_i for some $i \in J_q$.

We emphasize that the lemma only shows the existence of such indices i and j, but does not provide an efficient algorithm to identify them. It is also useful to note that for any $i \in [n]$, $\max\{|Rx_i^{\mathrm{ll}}|, |Rs_i^{\mathrm{ll}}|\} \geq \frac{1}{2} - \frac{3}{4}\beta$ according to Lemma 4.3.10(iii). Thus, for each $q \in [p]$, we obtain a strong and positive lower bound in either case (i) on x_i/x_i^* or case (ii) on s_i/s_i^* for some $i \in J_q$.

The next lemma allows us to argue that the potential function $\Psi^{\cdot}(\cdot,\cdot)$ increases for multiple pairs of variables, if we have strong lower bounds on both x_i^* and s_j^* for some $i, j \in [n]$, along with a lower and upper bound on $\varrho^{\mu}(i,j)$.

Lemma 4.4.3 (Proof on p. 90). *Let $w = (x,y,s) \in \mathcal{N}(2\beta)$ for $\beta \in (0,1/8]$, let $\mu = \mu(w)$ and $\delta = \delta(w)$. Let $i, j \in [n]$ and $2 \leq \tau \leq n$ such that for the optimal solution $w^* = (x^*, y^*, s^*)$, we have $x_i^* \geq \beta x_i/(2^{10}n^{5.5})$ and $s_j^* \geq \beta s_j/(2^{10}n^{5.5})$, and assume $\varrho^{\mu}(i,j) \geq -\tau$. After $O(\beta^{-1}\sqrt{n}\tau\log(\bar{\chi}^* + n))$ further iterations the duality gap μ' fulfills $\Psi^{\mu'}(i,j) \geq 2\tau$, and for every $\ell \in [n] \setminus \{i,j\}$, either $\Psi^{\mu'}(i,\ell) \geq 2\tau$, or $\Psi^{\mu'}(\ell,j) \geq 2\tau$.*

We note that i and j as in the lemma are necessarily different, since $i = j$ would imply $0 = x_i^* s_i^* \geq \beta^2\mu/(2^{20}n^{11}) > 0$.

Let us illustrate the idea of the proof of $\Psi^{\mu'}(i,j) \geq 2\tau$. For i and j as in the lemma, and for a central path element $w' = w(\mu')$ for $\mu' < \mu$, we have $x_i' \geq x_i^*/n \geq \beta x_i/(2^{10}n^{6.5})$ and $s_j' \geq s_j^*/n \geq \beta s_j/(2^{10}n^{6.5})$ by the near-monotonicity of the central path (Lemma 4.3.3). Note that

$$\kappa_{ij}^{\delta'} = \kappa_{ij} \cdot \frac{\delta_j'}{\delta_i'} = \kappa_{ij} \cdot \frac{x_i' s_j'}{\mu'} \geq \kappa_{ij} \cdot \frac{\beta^2 x_i s_j}{2^{20}n^{13}\mu'} \geq \frac{\beta^2(1-\beta)^2}{2^{20}n^{13}} \cdot \kappa_{ij}^{\delta} \cdot \frac{\mu}{\mu'},$$

where the last inequality uses Proposition 4.3.2. Consequently, as μ' sufficiently decreases, $\kappa_{ij}^{\delta'}$ will become much larger than κ_{ij}^{δ}. The claim on $\ell \in [n] \setminus \{i,j\}$ can be shown by using the triangle inequality $\kappa_{ik} \cdot \kappa_{kj} \geq \kappa_{ij}$ shown in Theorem 3.2.22.

Assume now $\xi_{J_q}^{\mathrm{ll}}(w) \geq 4\gamma n$ for some $q \in [p]$ in the LLS step. Then, Lemma 4.4.2 guarantees the existence of $i, j \in J_q$ such that $x_i^*/x_i, s_j^*/s_j \geq \frac{4}{3\sqrt{n}}\gamma n > \beta/(2^{10}n^{5.5})$. Further, Lemma 4.4.1 gives $\varrho^{\mu}(i,j) \geq -|J_q|$. Hence, Lemma 4.4.3 is applicable for i and j with $\tau = |J_q|$.

The overall potential argument in the proof of Theorem 4.3.16 uses Lemma 4.4.3 in three cases: $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) \geq 4\gamma n$ (Lemma 4.4.2 is applicable as above); $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) < 4\gamma n$ and $\ell^{\delta^+}(\mathcal{J}) \leq 4\gamma n$ (Lemma 4.4.4); and $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) < 4\gamma n$ and $\ell^{\delta^+}(\mathcal{J}) > 4\gamma n$ (Lemma 4.4.5). Here, δ^+ refers to the value of δ after the LLS step. Note that $\delta^+ > 0$ is well-defined, unless the algorithm terminated with an optimal solution.

To prove these lemmas, we need to study how the layers "move" during the LLS step. We let $\mathfrak{B} = \{t \in [n] : |Rs_t^{\mathrm{ll}}| < 4\gamma n\}$ and $\mathfrak{R} = \{t \in [n] : |Rx_t^{\mathrm{ll}}| < 4\gamma n\}$. The assumption $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) < 4\gamma n$ means that for each layer J_k, either $J_k \subseteq \mathfrak{B}$ or $J_k \subseteq \mathfrak{R}$; we accordingly refer to $\mathfrak{B}$-layers and $\mathfrak{R}$-layers.

Lemma 4.4.4 (Proof on p. 92). *Let $w = (x,y,s) \in \mathcal{N}(\beta)$ for $\beta \in (0,1/8]$, and let $\mathcal{J} = (J_1, \ldots, J_p)$ be a $\delta(w)$-balanced partition. Assume that $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) < 4\gamma n$, and let $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$ be the next iterate obtained by the LLS step with $\mu^+ = \mu(w^+)$ and assume $\mu^+ > 0$. Let $q \in [p]$ such that $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) = \xi_{J_q}^{\mathrm{ll}}(w)$. If $\ell^{\delta^+}(\mathcal{J}) \leq 4\gamma n$, then there exist $i, j \in J_q$ such that $x_i^* \geq \beta x_i^+/(16n^{3/2})$ and $s_j^* \geq \beta s_j^+/(16n^{3/2})$. Further, for any $\ell, \ell' \in J_q$, we have $\varrho^{\mu^+}(\ell,\ell') \geq -|J_q|$.*

For the proof sketch, without loss of generality, let $\xi_{\mathcal{J}}^{\mathrm{ll}} = \xi_{J_q}^{\mathrm{ll}} = \|Rx_{J_q}^{\mathrm{ll}}\|$, that is, J_q is an $\mathfrak{R}$-layer. The case $\xi_{J_q}^{\mathrm{ll}} = \|Rs_{J_q}^{\mathrm{ll}}\|$ can be treated analogously. Since the residuals $\|Rx_{J_q}^{\mathrm{ll}}\|$ and $\|Rs_{J_q}^{\mathrm{ll}}\|$ cannot be both small, Lemma 4.4.2 readily provides a $j \in J_q$ such that $s_j^*/s_j \geq 1/(6\sqrt{n})$. Using Lemma 5.2.3 and Proposition 4.3.1, $s_j^*/s_j^+ = s_j^*/s_j \cdot s_j/s_j^+ > (1-\beta)/(6(1+4\beta)n^{3/2}) > \beta/(16n^{3/2})$.

The key ideas of showing the existence of an $i \in J_q$ such that $x_i^* \geq x_i^+/(16n^{3/2})$ are the following. With $\approx$, $\lessapprox$ and $\gtrapprox$, we write equalities and inequalities that hold up to small polynomial factors. First, we show that *(i)* $\|\delta_{J_q} x^+_{J_q}\| \lessapprox \mu^+/\sqrt{\mu}$, and then, that *(ii)* $\|\delta_{J_q} x^*_{J_q}\| \gtrapprox \mu^+/\sqrt{\mu}$.

If we can show *(i)* and *(ii)* as above, we obtain that $\|\delta_{J_q} x^*_{J_q}\| \gtrapprox \|\delta_{J_q} x^+_{J_q}\|$, and thus, $x_i^* \gtrapprox x_i^+$ for some $i \in J_q$.

Let us now sketch the first step. By the assumption $J_q \subset \mathfrak{N}$, one can show $x^+_{J_q}/x_{J_q} \approx \mu^+/\mu$, and therefore

$$\|\delta_{J_q} x^+_{J_q}\| \approx \frac{\mu^+}{\mu}\|\delta_{J_q} x_{J_q}\| \approx \frac{\mu^+}{\mu}\sqrt{\mu} = \frac{\mu^+}{\sqrt{\mu}}\,.$$

The second part of the proof, namely, lower bounding $\|\delta_{J_q} x^*_{J_q}\|$, is more difficult. Here, we only sketch it for the special case when $J_q = [n]$. That is, we have a single layer only; in particular, the LLS step is the same as the affine scaling step $\Delta x^{\mathrm{ll}} = \Delta x^{\mathrm{a}}$. The general case of multiple layers follows by making use of Lemma 4.3.10, i.e. exploting that for a sufficiently small $\ell^\delta(\mathcal{J})$, the LLS step is close to the affine scaling step.

Hence, assume that $\Delta x^{\mathrm{ll}} = \Delta x^{\mathrm{a}}$. Using the equivalent definition of the affine scaling step (4.10) as a minimum-norm point, we have $\|\delta x^*\| \geq \|\delta(x + \Delta x^{\mathrm{ll}})\| = \sqrt{\mu}\|Rx^{\mathrm{ll}}\| = \sqrt{\mu}\xi^{\mathrm{ll}}_{\mathcal{J}}$. From Lemma 4.3.6, $\mu^+/\mu \leq 2\sqrt{n}\epsilon^{\mathrm{a}}(w)/\beta \leq 2\sqrt{n}\xi^{\mathrm{ll}}_{\mathcal{J}}/\beta$. Thus, we see that $\|\delta x^*\| \geq \beta\mu^+/(2\sqrt{n\mu})$.

The final statement on lower bounding $\varrho^{\mu^+}(\ell, \ell') \geq -|J_q|$ for any $\ell, \ell' \in J_q$ follows by showing that $\delta^+_\ell/\delta^+_{\ell'}$ remains close to $\delta_\ell/\delta_{\ell'}$, and hence the values of $\kappa^{\mu^+}(\ell, \ell')$ and $\kappa^{\mu}(\ell, \ell')$ are sufficiently close for indices on the same layer (Lemma 4.6.1).

Lemma 4.4.5 (Proof on p. 95)**.** *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$, and let $\mathcal{J} = (J_1, \ldots, J_p)$ be a $\delta(w)$-balanced partition. Assume that $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$, and let $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$ be the next iterate obtained by the LLS step with $\mu^+ = \mu(w^+)$ and assume $\mu^+ > 0$. If $\ell^{\delta^+}(\mathcal{J}) > 4\gamma n$, then there exist two layers J_q and J_r and $i \in J_q$ and $j \in J_r$ such that $x_i^* \geq x_i^+/(8n^{3/2})$, and $s_j^* \geq s_j^+/(8n^{3/2})$. Further, $\varrho^{\mu^+}(i, j) \geq -|J_q \cup J_r|$, and for all $\ell, \ell' \in J_q \cup J_r$, $\ell \neq \ell'$ we have $\Psi^{\mu}(\ell, \ell') \leq |J_q \cup J_r|$.*

Consider now any $\ell \in J_k \subseteq \mathfrak{B}$. Then, since Rx^{ll}_ℓ is multiplicatively close to 1, $x^+_\ell \approx x_\ell$; on the other hand s^+_ℓ will "shoot down" close to the small value $Rs^{\mathrm{ll}}_\ell \cdot s_\ell$. Conversely, for $\ell \in J_k \subseteq \mathfrak{N}$, $s^+_\ell \approx s_\ell$, and x^+_ℓ will "shoot down" to a small value.

The key step of the analysis is showing that the increase in $\ell^{\delta^+}(\mathcal{J})$ can be attributed to an $\mathfrak{N}$-layer J_r "crashing into" a $\mathfrak{B}$-layer J_q. That is, we show the existence of an edge $(i', j') \in E_{\delta^+, \gamma/(4n)}$ for $i' \in J_q$ and $j' \in J_r$, where $r < q$ and $J_q \subseteq \mathfrak{B}$, $J_r \subseteq \mathfrak{N}$. This can be achieved by analyzing the matrix B used in the subroutine Verify-Lift.

For the layers J_q and J_r, we can use Lemma 4.4.2 to show that there exists an $i \in J_q$ where x_i^*/x_i is lower bounded, and there exists a $j \in J_r$ where s_j^*/s_j is lower bounded. The lower bound on $\varrho^{\mu^+}(i, j)$ and the upper bounds on the $\Psi^{\mu}(\ell, \ell')$ values can be shown by tracking the changes between the $\kappa^{\delta}(\ell, \ell')$ and $\kappa^{\delta^+}(\ell, \ell')$ values, and applying Lemma 4.4.1 both at w and at w^+.

Proof of Theorem 4.3.16. We analyze the overall potential function $\Psi(\mu)$. By the *iteration at* μ we mean the iteration where the normalized duality gap of the current iterate is μ.

By Proposition 4.3.4(ii) and Lemma 4.3.10(ii), the predictor step gives $w' \in \mathcal{N}(1/4)$ in every iteration, and thus by Proposition 4.3.4(iii), all iterates w^{c} after corrector step fulfill $w^{\mathrm{c}} \in \mathcal{N}(1/8)$. If $\mu^+ = 0$ at the end of an iteration, the algorithm terminates with an optimal solution. Recall from Lemma 4.3.10(v) that this happens if and only if $\epsilon^{\mathrm{ll}}(w) = 0$ at a certain iteration.

From now on, assume that $\mu^+ > 0$. We distinguish three cases at each iteration. These cases are well-defined even at iterations where affine scaling steps are used. At such iterations, $\xi^{\mathrm{ll}}_{\mathcal{J}}(w)$ still refers to the LLS residuals, even if these have not been computed by the algorithm. *(Case*

I) $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) \geq 4\gamma n$; *(Case II)* $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$ and $\ell^{\delta^+}(\mathcal{J}) \leq 4\gamma n$; and *(Case III)* $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$ and $\ell^{\delta^+}(\mathcal{J}) > 4\gamma n$.

Recall that the algorithm uses an LLS direction instead of the affine scaling direction whenever $\epsilon^{\mathrm{a}}(w) < 10n^{3/2}\gamma$. Consider now the case when an affine scaling direction is used, that is, $\epsilon^{\mathrm{a}}(w) \geq 10n^{3/2}\gamma$. According to Lemma 4.3.10(ii), $\|Rx^{\mathrm{ll}} - Rx^{\mathrm{a}}\|, \|Rs^{\mathrm{ll}} - Rs^{\mathrm{a}}\| \leq 6n^{3/2}\gamma$. This implies that $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) \geq 4n^{3/2}\gamma \geq 4n\gamma$. Therefore, in cases II and III, an LLS step will be performed.

Starting with any given iteration, in each case we will identify a set $J \subseteq [n]$ of indices with $|J| > 1$, and start a *phase* of $O(\sqrt{n}|J|\log(\bar{\chi}^* + n))$ iterations (that can be either affine scaling or LLS steps). In each phase, we will guarantee that Ψ increases by at least $|J| - 1$. By definition, $0 \leq \Psi(\mu) \leq n(n-1)(\log_2 n + 1)$, and if $\mu' < \mu$ then $\Psi(\mu') \geq \Psi(\mu)$. As we can partition the union of all iterations into disjoint phases, this yields the bound $O(n^{2.5}\log n \log(\bar{\chi}^* + n))$ on the total number of iterations.

We now consider each of the cases. We always let μ denote the normalized duality gap at the current iteration, and we let $q \in [p]$ be the layer such that $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) = \xi^{\mathrm{ll}}_{J_q}(w)$.

Case I: $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) \geq 4\gamma n$**.** Lemma 4.4.2 guarantees the existence of $x_i, s_j \in J_q$ such that $x^*_i/x_i, s^*_j/s_j \geq 4\gamma n/(3\sqrt{n}) > 1/(2^{10}n^{5.5})$. Further, according to Lemma 4.4.1, $\varrho^{\mu}(i,j) \geq -|J_q|$. Thus, Lemma 4.4.3 is applicable for $J = J_q$. The phase starting at μ comprises $O(\sqrt{n}|J_q|\log(\bar{\chi}^* + n))$ iterations, after which we get a normalized duality gap μ' such that $\Psi^{\mu'}(i,j) \geq 2|J_q|$, and for each $\ell \in [n] \setminus \{i,j\}$, either $\Psi^{\mu'}(i,\ell) \geq 2|J_q|$, or $\Psi^{\mu'}(\ell,j) \geq 2|J_q|$.

We can take advantage of these bounds for indices $\ell \in J_q$. Again by Lemma 4.4.1, for any $\ell, \ell' \in J_q$, we have $\Psi^{\mu}(\ell,\ell') \leq \varrho^{\mu}(\ell,\ell') \leq |J_q|$. Thus, there are at least $|J_q| - 1$ pairs of indices (ℓ,ℓ') for which $\Psi^{\mu}(\ell,\ell')$ increases by at least a factor 2 between iterations at μ and μ'. The increase in the contribution of these terms to $\Psi(\mu)$ is at least $|J_q| - 1$ during these iterations.

We note that this analysis works regardless whether an LLS step or an affine scaling step was performed in the iteration at μ.

Case II: $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$ **and** $\ell^{\delta^+}(\mathcal{J}) \leq 4\gamma n$**.** As explained above, in this case we perform an LLS step in the iteration at μ, and we let w^+ denote the iterate obtained by the LLS step. For $J = J_q$, Lemma 4.4.4 guarantees the existence of $i, j \in J_q$ such that $x^*_i/x^+_i, s^*_j/s^+_j > \beta/(16n^{3/2})$, and further, $\varrho^{\mu^+}(i,j) > -|J_q|$. We can therefore apply Lemma 4.4.3. The phase starting at μ includes the LLS step leading to μ^+ (and the subsequent centering step), and the additional $O(\sqrt{n}|J_q|\log(\bar{\chi}^* + n))$ iterations (β is a fixed constant in Algorithm 4.2) as in Lemma 4.4.3. As in Case I, we get the desired potential increase compared to the potentials at μ in layer J_q.

Case III: $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$ **and** $\ell^{\delta^+}(\mathcal{J}) > 4\gamma n$**.** Again, the iteration at μ will use an LLS step. We apply Lemma 4.4.5, and set $J = J_q \cup J_r$ as in the lemma. The argument is the same as in Case II, using that Lemma 4.4.5 explicitly states that $\Psi^{\mu}(\ell,\ell') \leq |J|$ for any $\ell, \ell' \in J$, $\ell \neq \ell'$. □

4.4.1 The iteration complexity bound for the Vavasis-Ye algorithm

We now show that the potential analysis described above also gives an improved bound $O(n^{2.5}\log n \log(\bar{\chi}_{\mathbf{A}} + n))$ for the original VY algorithm [VY96].

We recall the VY layering step. Order the variables via π such that $\delta_{\pi(1)} \leq \delta_{\pi(2)} \leq \ldots \leq \delta_{\pi(n)}$. The layers will be consecutive sets in the ordering; a new layer starts with $\pi(i+1)$ each time $\delta_{\pi(i+1)} > g\delta_{\pi(i)}$, for a parameter $g = \mathrm{poly}(n)\bar{\chi}_{\mathbf{A}}$.

As outlined in the Introduction, the VY algorithm can be seen as a special implementation of our algorithm by setting $\hat{\kappa}_{ij} = g\gamma/n$. With these edge weights, we have that $\hat{\kappa}^{\delta}_{ij} \geq \gamma/n$ precisely if $g\delta_j \geq \delta_i$.[2]

With these edge weights, it is easy to see that our LAYERING($\delta, \hat{\kappa}$) subroutine finds the exact same components as VY. Moreover, the layers will be the initial strongly connected components C_i of $G_{\delta,\gamma/n}$: due to the choice of g, this partition is automatically δ-balanced. There is no need to call VERIFY-LIFT.

The essential difference compared to our algorithm is that the values $\hat{\kappa}_{ij} = g\gamma/n$ are not lower bounds on κ_{ij} as we require, but upper bounds instead. This is convenient to simplify the construction of the layering. On the negative side, the strongly connected components of $\hat{G}_{\delta,\gamma/n}$ may not anymore be strongly connected in $G_{\delta,\gamma/n}$. Hence, we cannot use Lemma 4.4.1, and consequently, Lemma 4.4.3 does not hold.

Still, the $\hat{\kappa}_{ij}$ bounds are overestimating κ_{ij} by at most a factor $\text{poly}(n)\bar{\chi}_{\mathbf{A}}$. Therefore, the strongly connected components of $\hat{G}_{\delta,n/\gamma}$ are strongly connected in $G_{\delta,\sigma}$ for some $\sigma = 1/(\text{poly}(n)\bar{\chi}_{\mathbf{A}})$.

Hence, the entire argument described in this section is applicable to the VY algorithm, with a different potential function defined with $\bar{\chi}_{\mathbf{A}}$ instead of $\bar{\chi}^*_{\mathbf{A}}$. This is the reason why the iteration bound in Lemma 4.4.3, and therefore in Theorem 4.3.16, also changes to $\bar{\chi}_{\mathbf{A}}$ dependency.

It is worth noting that due to the overestimation of the κ_{ij} values, the VY algorithm uses a coarser layering than our algorithm. Our algorithm splits up the VY layers into smaller parts so that $\ell^{\delta}(\mathcal{J})$ remains small, but within each part, the gaps between the variables are bounded as a function of $\bar{\chi}^*_{\mathbf{A}}$ instead of $\bar{\chi}_{\mathbf{A}}$.

4.5 Properties of the layered least square step

This section is dedicated to the proofs of Proposition 4.3.8 on the duality of lifting scores and Lemma 4.3.10 on properties of LLS steps.

Proposition 4.3.8 (Restatement). *For a linear subspace $W \subseteq \mathbb{R}^n$ and index set $I \subseteq [n]$ with $J = [n] \setminus I$,*

$$\|L_I^W\| \leq \max\{1, \|L_J^{W^\perp}\|\}.$$

In particular, $\ell^W(I) = \ell^{W^\perp}(J)$.

Proof. We first treat the case where $\pi_I(W) = \{0\}$ or $\pi_J(W^\perp) = \{0\}$. If $\pi_I(W) = \{0\}$ then $\|L_I^W\| = \ell^W(I) = \mathbf{0}$. Furthermore, in this case $\mathbb{R}^I = \pi_I(W)^\perp = \pi_I(W^\perp \cap \mathbb{R}^n_I)$, and thus $\{(0, w_J) : w \in W^\perp\} \subseteq W^\perp$. In particular, $\|L_J^W\| \leq 1$ and $\ell^{W^\perp}(J) = 0$. Symmetrically, if $\pi_J(W^\perp) = \{\mathbf{0}\}$ then $\|L_J^{W^\perp}\| = \ell^{W^\perp}(J) = 0$, $\|L_I^W\| \leq 1$ and $\ell^W(I) = 0$.

We now restrict our attention to the case where both $\pi_I(W), \pi_J(W^\perp) \neq \{\mathbf{0}\}$. Under this assumption, we show that $\|L_I^W\| = \|L_J^{W^\perp}\|$ and thus that $\ell^W(I) = \ell^{W^\perp}(J)$. Note that by non-emptiness, we have that $\|L_I^W\|, \|L_J^{W^\perp}\| \geq 1$.

We formulate a more general claim. Let $\{\mathbf{0}\} \neq U, V \subset \mathbb{R}^n$ be linear subspaces such that $U + V = \mathbb{R}^n$ and $U \cap V = \{0\}$. Note that for the orthogonal complements in $\mathbb{R}^n$, we also have $\{0\} \neq U^\perp, V^\perp, U^\perp + V^\perp = \mathbb{R}^n$ and $U^\perp \cap V^\perp = \{\mathbf{0}\}$.

Claim 4.5.0.1. *Let $\{0\} \neq U, V \subset \mathbb{R}^n$ be linear subspaces such that $U + V = \mathbb{R}^n$ and $U \cap V = \{0\}$. Thus, for $z \in \mathbb{R}^n$, there are unique decompositions $z = u + v$ with $u \in U$, $v \in V$ and $z = u' + v'$ with $u' \in U^\perp$ and $v' \in V^\perp$. Let $T : \mathbb{R}^n \to V$ be the map sending $Tz = v$. Let $T' : \mathbb{R}^n \to V^\perp$ be the map sending $T'z = v'$. Then, $\|T\| = \|T'\|$.*

[2] For simplicity, in the Introduction we used $gx_i \geq x_j$ instead, which is almost the same in the proximity in the central path.

Proof. To prove the statement, we claim that it suffices to show that if $\|T\| > 1$ then $\|T'\| \geq \|T\|$. To prove sufficiency, note that by symmetry, we also get that if $\|T'\| > 1$ then $\|T\| \geq \|T'\|$. Note that $V, V^\perp \neq \{0\}$ by assumption, and $Tz = z$ for $z \in V$, $T'z = z$ for $z \in V^\perp$. Thus, we always have $\|T\|, \|T'\| \geq 1$, and therefore the equality $\|T\| = \|T'\|$ must hold in all cases. We now assume $\|T\| > 1$ and show $\|T'\| \geq \|T\|$.

Representing T as an $n \times n$ matrix, we write $T = \sum_{i=1}^{k} \sigma_i v_i u_i^\top$ using a singular value decomposition with $\sigma_1 \geq \cdots \geq \sigma_k > 0$. As such, $v_1, \ldots, v_k$ is an orthonormal basis of V, since the range$(T) = V$, and $u_1, \ldots, u_k$ is an orthonormal basis of $U^\perp$, since $\ker(T) = U$, noting that we have restricted to the singular vectors associated with positive singular values. By assumption, we have that $\|T\| = \|Tu_1\| = \sigma_1 > 1$.

The proof is complete by showing that

$$\|T'(v_1 - u_1/\sigma_1)\| \geq \sigma_1 \|v_1 - u_1/\sigma_1\|, \tag{4.27}$$

and that $\|v_1 - u_1/\sigma_1\| > 0$, since then the vector $v_1 - u_1/\sigma_1$ will certify that $\|T'\| \geq \sigma_1$.

The map T is a linear projection with $T^2 = T$. Hence $\langle u_i, v_i \rangle = \sigma_i^{-1}$ and $\langle u_i, v_j \rangle = 0$ for all $i \neq j$.

We show that $v_1 - \sigma_1^{-1} u_1$ can be decomposed as $v_1 - \sigma_1 u_1 + (\sigma_1 - \sigma_1^{-1}) u_1$ such that $v_1 - \sigma_1 u_1 \in V^\perp$ and $(\sigma_1 - \sigma_1^{-1}) u_1 \in U^\perp$. Therefore, $T'(v_1 - \sigma_1^{-1} u_1) = v_1 - \sigma_1 u_1$.

The containment $(\sigma_1 - \sigma_1^{-1}) u_1 \in U^\perp$ is immediate. To show $v_1 - \sigma_1 u_1 \in V^\perp$, we need that $\langle v_1 - \sigma_1 u_1, v_i \rangle = 0$ for all $i \in [k]$. For $i \geq 2$, this is true since $\langle u_i, v_j \rangle = 0$ and $\langle v_i, v_j \rangle = 0$. For $i = 1$, we have $\langle v_1 - \sigma_1 u_1, v_1 \rangle = 0$ since $\|v_1\| = 1$ and $\langle u_1, v_1 \rangle = \sigma_1^{-1}$. Consequently, $T'(v_1 - \sigma_1^{-1} u_1) = v_1 - \sigma_1 u_1$.

We compute $\left\|v_1 - \sigma_1^{-1} u_1\right\| = \sqrt{1 - \sigma_1^{-2}} > 0$, since $\sigma_1 > 1$, and $\|v_1 - \sigma_1 u_1\| = \sqrt{\sigma_1^2 - 1}$. This verifies (4.27), and thus $\|T'\| \geq \sigma_1 = \|T\|$. □

To prove the lemma, we define $\mathcal{J} = (J, I)$, $U = W_{\mathcal{J},1}^\perp \times W_{\mathcal{J},2}^\perp$ and $V = W$ and let $T : \mathbb{R}^n \to V$ and $T' : \mathbb{R}^n \to V^\perp$ be as in Claim 4.5.0.1. By assumption, $\{\mathbf{0}\} \neq \pi_I(W) \Rightarrow \{\mathbf{0}\} \neq V$ and $\{\mathbf{0}\} \neq \pi_J(W^\perp) = W_{\mathcal{J},1}^\perp \Rightarrow \{\mathbf{0}\} \neq U$. Applying Lemma 4.3.7, U, V satisfy the conditions of Claim 4.5.0.1 and $T = \mathrm{LLS}_{\mathcal{J}}^{W,1}$. In particular, $\|T'\| = \|T\|$. Using the fact that $U^\perp = W_{\mathcal{J},1} \times W_{\mathcal{J},2}$ and $V^\perp = W^\perp$, we similarly get that $T' = \mathrm{LLS}_{\bar{\mathcal{J}}}^{W^\perp,1}$, where $\bar{\mathcal{J}} = (I, J)$. By (4.14) we have, for any $t \in \pi_{\mathbb{R}_I^n}(W)$, that $Tt = \mathrm{LLS}_{\mathcal{J}}^{W,1}(t) = L_I^W(t_I)$. Thus, $\|T\| \geq \|L_I^W\| \geq 1$.

To finish the proof of the lemma from the claim, we show that $\|T\| \leq \|L_I^W\|$. By a symmetric argument we get $\|T'\| = \|L_J^{W^\perp}\|$.

If $x \in \mathbb{R}_J^n$, then $Tx \in W \cap \mathbb{R}_J^n$ because any $s \in W_{\mathcal{J},2}^\perp, t \in \pi_I(W)$ with $s + t = 0$ must have $s = t = 0$ since $W_{\mathcal{J},2}^\perp$ is orthogonal to $\pi_I(W)$. But $W \cap \mathbb{R}_J^n$ and $W_{\mathcal{J},1}^\perp$ are orthogonal, so $\|Tx\| \leq \|x\|$ because $x = Tx + (x - Tx)$ is an orthogonal decomposition.

If $y \in \mathbb{R}_I^n$, then $y_J = 0$ and hence $(Ty)_J = (Ty - y)_J$. Since $(Ty - y)_J \in W_{\mathcal{J},1}^\perp = \pi_J(W \cap \mathbb{R}_J^n)^\perp$, we see that $Ty \in (W \cap \mathbb{R}_J^n)^\perp$. As such, for any $x \in \mathbb{R}_J^n, y \in \mathbb{R}_I^n$, we see that $x \perp y$ and $Tx \perp Ty$. For $x, y \neq 0$, we thus have that

$$\frac{\|T(x+y)\|^2}{\|x+y\|^2} = \frac{\|T(x)\|^2 + \|T(y)\|^2}{\|x\|^2 + \|y^2\|} \leq \max\left\{\frac{\|T(x)\|^2}{\|x\|^2}, \frac{\|T(y)\|^2}{\|y\|^2}\right\} \leq \max\left\{1, \frac{\|T(y)\|^2}{\|y\|^2}\right\}.$$

Since $\|L_I^W\| \geq 1$, we must have that $\|Tt\|/\|t\|$ is maximized by some $t \in \mathbb{R}_I^n$. From $\ker(T) = U$ it is clear that $\|Tt\|/\|t\|$ is maximized by some $t \in U^\perp$. Now, $U^\perp \cap \mathbb{R}_I^n = \pi_{\mathbb{R}_I^n}(W)$, so any t maximizing $\|Tt\|/\|t\|$ satisfies $Tt = L_I^W(t_I)$. Therefore, $\|L_I^W\| \geq \|T\|$. □

Our next goal is to show Lemma 4.3.10: for a layering with small enough $\ell^\delta(\mathcal{J})$, the LLS step approximately satisfies (4.6), that is, $\delta \Delta x^{\mathrm{ll}} + \delta^{-1} \Delta s^{\mathrm{ll}} \approx -x^{1/2} s^{1/2}$. This also enables us to derive

bounds on the norm of the residuals and on the step-length. We start by proving a few auxiliary technical claims. The next simple lemma allows us to take advantage of low lifting scores in the layering.

Lemma 4.5.1. *Let $u, v \in \mathbb{R}^n$ be two vectors such that $u - v \in W$. Let $I \subseteq [n]$, and $\delta \in \mathbb{R}^n_{++}$. Then there exists a vector $u' \in W + u$ satisfying $u'_I = v_I$ and*

$$\|\delta_{[n]\setminus I}(u'_{[n]\setminus I} - u_{[n]\setminus I})\| \leq \ell^\delta(I)\|\delta_I(u_I - v_I)\| .$$

Proof. We let

$$u' := u + \delta^{-1}L_I^\delta(\delta_I(v_I - u_I)) .$$

The claim follows by the definition of the lifting score $\ell^\delta(I)$. □

The next lemma will be the key tool to prove Lemma 4.3.10. It is helpful to recall the characterization of the LLS step in Section 4.3.4.

Lemma 4.5.2. *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, let $\mu = \mu(w)$ and $\delta = \delta(w)$. Let $\mathcal{J} = (J_1, \ldots, J_p)$ be a $\delta(w)$-balanced layering, and let $\Delta w^{\text{ll}} = (\Delta x^{\text{ll}}, \Delta y^{\text{ll}}, \Delta s^{\text{ll}})$ denote the corresponding LLS direction. Let $\Delta x \in \bigtimes_{k=1}^p W_{\mathcal{J},k}$ and $\Delta s \in \bigtimes_{k=1}^p W^\perp_{\mathcal{J},k}$ as in (4.18) and (4.19), that is*

$$\delta\Delta x^{\text{ll}} + \delta^{-1}\Delta s + x^{1/2}s^{1/2} = \mathbf{0} , \tag{4.28}$$
$$\delta\Delta x + \delta^{-1}\Delta s^{\text{ll}} + x^{1/2}s^{1/2} = \mathbf{0} . \tag{4.29}$$

Then, there exist vectors $\Delta\bar{x} \in \bigtimes_{k=1}^p W_{\mathcal{J},k}$ and $\Delta\bar{s} \in \bigtimes_{k=1}^p W^\perp_{\mathcal{J},k}$ such that

$$\|\delta_{J_k}(\Delta\bar{x}_{J_k} - \Delta x^{\text{ll}}_{J_k})\| \leq 2n\ell^\delta(\mathcal{J})\sqrt{\mu} \quad \forall k \in [p] \quad \text{and} \tag{4.30}$$
$$\|\delta^{-1}_{J_k}(\Delta\bar{s}_{J_k} - \Delta s^{\text{ll}}_{J_k})\| \leq 2n\ell^\delta(\mathcal{J})\sqrt{\mu} \quad \forall k \in [p] . \tag{4.31}$$

Proof. Throughout, we use the shorthand notation $\lambda = \ell^\delta(\mathcal{J})$. We construct $\Delta\bar{x}$; one can obtain $\Delta\bar{s}$, using that the reverse layering has lifting score λ in $W^\perp\text{diag}(\delta^{-1})$ according to Lemma 4.3.9.

We proceed by induction, constructing $\Delta\bar{x}_{J_k} \in W_{\mathcal{J},k}$ for $k = p, p-1, \ldots, 1$. This will be given as $\Delta\bar{x}_{J_k} = \Delta x^{(k)}_{J_k}$ for a vector $\Delta x^{(k)} \in W$ such that $\Delta x^{(k)}_{J_{>k}} = \mathbf{0}$. We prove the inductive hypothesis

$$\left\|\delta_{J_{\leq k}}\left(\Delta x^{(k)}_{J_{\leq k}} - \Delta x^{\text{ll}}_{J_{\leq k}}\right)\right\| \leq 2\lambda\sqrt{\mu}\sum_{q=k+1}^{p}\sqrt{|J_q|} . \tag{4.32}$$

Note that (4.30) follows by restricting the norm on the LHS to J_k and since the sum on the RHS is $\leq n$.

For $k = p$, the RHS is 0. We simply set $\Delta x^{(p)} = \Delta x^{\text{ll}}$, that is, $\Delta\bar{x}_{J_p} = \Delta x^{\text{ll}}_{J_p}$, trivially satisfying the hypothesis. Consider now $k < p$, and assume that we have a $\Delta\bar{x}_{J_{k+1}} = \Delta x^{(k+1)}_{J_{k+1}}$ satisfying (4.32) for $k+1$. From (4.28) and the induction hypothesis, we get that

$$\begin{aligned}\|\delta_{J_{k+1}}\Delta\bar{x}_{J_{k+1}} + \delta^{-1}_{J_{k+1}}\Delta s_{J_{k+1}}\| &\leq \|x^{1/2}_{J_{k+1}}s^{1/2}_{J_{k+1}}\| + \|\delta_{J_{k+1}}(\Delta\bar{x}_{J_{k+1}} - \Delta x^{\text{ll}}_{J_{k+1}})\| \\ &\leq \|x^{1/2}_{J_{k+1}}s^{1/2}_{J_{k+1}}\| + 2\lambda\sqrt{\mu}\sum_{q=k+2}^{p}\sqrt{|J_q|} \leq \sqrt{1+\beta}\sqrt{\mu|J_{k+1}|} + 2n\lambda\sqrt{\mu} < 2\sqrt{\mu|J_{k+1}|} ,\end{aligned}$$

using also that $w \in \mathcal{N}(\beta)$, Proposition 4.3.2, and the assumptions $\beta \le 1/4$, $\lambda \le \beta/(32n^2)$. Note that $\Delta\bar{x}_{J_{k+1}} \in W_{\mathcal{J},k}$ and $\Delta s_{J_{k+1}} \in W^{\perp}_{\mathcal{J},k}$ are orthogonal vectors. The above inequality therefore implies

$$\|\delta_{J_{k+1}}\Delta\bar{x}_{J_{k+1}}\| \le 2\sqrt{\mu|J_{k+1}|}\,.$$

Let us now use Lemma 4.5.1 to obtain $\Delta x^{(k)}$ for $u = \Delta x^{(k+1)}$, $v = 0$, and $I = J_{>k}$. That is, we get $\Delta x^{(k)}_{J_{>k}} = 0$, $\Delta x^{(k)} \in W$, and

$$\begin{aligned}\|\delta_{J_{\le k}}(\Delta x^{(k)}_{J_{\le k}} - \Delta x^{(k+1)}_{J_{\le k}})\| &\le \lambda\|\delta_{J_{>k}}\Delta x^{(k+1)}_{J_{>k}}\| \\ &= \lambda\|\delta_{J_{k+1}}\Delta\bar{x}_{J_{k+1}}\| \le 2\lambda\sqrt{\mu|J_{k+1}|}\,.\end{aligned}$$

By the triangle inequality and the induction hypothesis (4.32) for $k+1$,

$$\begin{aligned}\|\delta_{J_{\le k}}(\Delta x^{(k)}_{J_{\le k}} - \Delta x^{\mathrm{ll}}_{J_{\le k}})\| &\le \|\delta_{J_{\le k}}(\Delta x^{(k)}_{J_{\le k}} - \Delta x^{(k+1)}_{J_{\le k}})\| + \|\delta_{J_{\le k}}(\Delta x^{(k+1)}_{J_{\le k}} - \Delta x^{\mathrm{ll}}_{J_{\le k}})\| \\ &\le 2\lambda\sqrt{\mu|J_{k+1}|} + 2\lambda\sum_{q=k+2}^{p}\sqrt{\mu|J_q|},\end{aligned}$$

yielding the induction hypothesis for k. □

Lemma 4.3.10 (Restatement). *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, let $\mu = \mu(w)$ and $\delta = \delta(w)$. Let $\mathcal{J} = (J_1, \dots, J_p)$ be a layering with $\ell^\delta(\mathcal{J}) \le \beta/(32n^2)$, and let $\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$ denote the LLS direction for the layering $\mathcal{J}$. Let furthermore $\epsilon^{\mathrm{ll}}(w) = \max_{i\in[n]}\min\{|Rx^{\mathrm{ll}}_i|, |Rs^{\mathrm{ll}}_i|\}$, and define the maximal step length as*

$$\alpha^* := \sup\big\{\,\alpha' \in [0,1] : \forall\bar{\alpha} \in [0,\alpha'] : w + \bar{\alpha}\Delta w^{\mathrm{ll}} \in \mathcal{N}(2\beta)\,\big\}\,.$$

Then the following properties hold.

(i) We have

$$\|\delta_{J_k}\Delta x^{\mathrm{ll}}_{J_k} + \delta^{-1}_{J_k}\Delta s^{\mathrm{ll}}_{J_k} + x^{1/2}_{J_k}s^{1/2}_{J_k}\| \le 6n\ell^\delta(\mathcal{J})\sqrt{\mu}\,, \quad \forall k \in [p],\ \text{and} \tag{4.20}$$

$$\|\delta\Delta x^{\mathrm{ll}} + \delta^{-1}\Delta s^{\mathrm{ll}} + x^{1/2}s^{1/2}\| \le 6n^{3/2}\ell^\delta(\mathcal{J})\sqrt{\mu}\,. \tag{4.21}$$

(ii) For the affine scaling direction $\Delta w^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta y^{\mathrm{a}}, \Delta s^{\mathrm{a}})$,

$$\|Rx^{\mathrm{ll}} - Rx^{\mathrm{a}}\|, \|Rs^{\mathrm{ll}} - Rs^{\mathrm{a}}\| \le 6n^{3/2}\ell^\delta(\mathcal{J})\,.$$

(iii) For the residuals of the LLS steps we have $\|Rx^{\mathrm{ll}}\|, \|Rs^{\mathrm{ll}}\| \le \sqrt{2n}$. For each $i \in [n]$, $\max\{|Rx^{\mathrm{ll}}_i|, |Rs^{\mathrm{ll}}_i|\} \ge \frac{1}{2} - \frac{3}{4}\beta$.

(iv) We have

$$\alpha^* \ge 1 - \frac{3\sqrt{n}\epsilon^{\mathrm{ll}}(w)}{\beta}\,, \tag{4.22}$$

and for any $\alpha \in [0,1]$

$$\mu(w + \alpha\Delta w^{\mathrm{ll}}) = (1-\alpha)\mu\,,$$

(v) We have $\epsilon^{\mathrm{ll}}(w) = 0$ if and only if $\alpha^* = 1$. These are further equivalent to $w + \Delta w^{\mathrm{ll}} = (x + \Delta x^{\mathrm{ll}}, y + \Delta y^{\mathrm{ll}}, s + \Delta s^{\mathrm{ll}})$ being an optimal solution to (LP).

Proof. Again, we use $\lambda = \ell^{\delta}(\mathcal{J})$.
Part (i). It is easy to see that (4.20) implies (4.21). To show (4.20), we use Lemma 4.5.2 to obtain $\Delta\bar{x}$ and $\Delta\bar{s}$ as in (4.30) and (4.31). We will also use $\Delta x \in \bigtimes_{k=1}^{p} W_{\mathcal{J},k}$ and $\Delta s \in \bigtimes_{k=1}^{p} W^{\perp}_{\mathcal{J},k}$ as in (4.28) and (4.29).

Select any layer $k \in [p]$. From (4.28), we get that

$$\|\delta_{J_k}\Delta\bar{x}_{J_k} + \delta_{J_k}^{-1}\Delta s_{J_k} + x_{J_k}^{1/2}s_{J_k}^{1/2}\| = \|\delta_{J_k}(\Delta\bar{x}_{J_k} - \Delta x^{\mathrm{ll}}_{J_k})\| \le 2n\lambda\sqrt{\mu}\,. \tag{4.33}$$

Similarly, from (4.29), we see that

$$\|\delta_{J_k}^{-1}\Delta\bar{s}_{J_k} + \delta_{J_k}\Delta x_{J_k} + x_{J_k}^{1/2}s_{J_k}^{1/2}\| = \|\delta_{J_k}^{-1}(\Delta\bar{s}_{J_k} - \Delta s^{\mathrm{ll}}_{J_k})\| \le 2n\lambda\sqrt{\mu}\,.$$

From the above inequalities, we see that

$$\|\delta_{J_k}(\Delta\bar{x}_{J_k} - \Delta x_{J_k}) + \delta_{J_k}^{-1}(\Delta s_{J_k} - \Delta\bar{s}_{J_k})\| \le 4n\lambda\sqrt{\mu}\,.$$

Since $\delta_{J_k}(\Delta\bar{x}_{J_k} - \Delta x_{J_k})$ and $\delta_{J_k}^{-1}(\Delta s_{J_k} - \Delta\bar{s}_{J_k})$ are orthogonal vectors, we have

$$\|\delta_{J_k}(\Delta\bar{x}_{J_k} - \Delta x_{J_k})\|,\ \|\delta_{J_k}^{-1}(\Delta s_{J_k} - \Delta\bar{s}_{J_k})\| \le 4n\lambda\sqrt{\mu}\,.$$

Together with (4.30), this yields $\|\delta_{J_k}(\Delta x^{\mathrm{ll}}_{J_k} - \Delta x_{J_k})\| \le 6n\lambda\sqrt{\mu}$. Combined with (4.19), we get

$$\|\delta_{J_k}\Delta x^{\mathrm{ll}}_{J_k} + \delta_{J_k}^{-1}\Delta s^{\mathrm{ll}}_{J_k} + x_{J_k}^{1/2}s_{J_k}^{1/2}\| = \|\delta_{J_k}(\Delta x^{\mathrm{ll}}_{J_k} - \Delta x_{J_k})\| \le 6n\lambda\sqrt{\mu}\,,$$

thus, (4.20) follows.

Part (ii). Recall from Lemma 4.3.5(i) that $\sqrt{\mu}Rx^{\mathrm{a}} + \sqrt{\mu}Rs^{\mathrm{a}} = x^{1/2}s^{1/2}$. From part (i), we can similarly see that

$$\|\sqrt{\mu}Rx^{\mathrm{ll}} + \sqrt{\mu}Rs^{\mathrm{ll}} - x^{1/2}s^{1/2}\| \le 6n^{3/2}\lambda\sqrt{\mu}\,.$$

From these, we get

$$\|(Rx^{\mathrm{ll}} - Rx^{\mathrm{a}}) + (Rs^{\mathrm{ll}} - Rs^{\mathrm{a}})\| \le 6n^{3/2}\lambda\,.$$

The claim follows since $Rx^{\mathrm{ll}} - Rx^{\mathrm{a}} \in \mathrm{diag}(\delta)W$ and $Rs^{\mathrm{ll}} - Rs^{\mathrm{a}} \in \mathrm{diag}(\delta^{-1})W^{\perp}$ are orthogonal vectors.

Part (iii). Both bounds follow from the previous part and Lemma 4.3.5(iii), using the assumption $\ell^{\delta}(\mathcal{J}) \le \beta/(32n^2)$.

Part (iv). Let $w^+ = w + \alpha\Delta w^{\mathrm{ll}}$. We need to find the largest value $\alpha > 0$ such that $w^+ \in \mathcal{N}(2\beta)$. To begin, we first show that the normalized duality gap $\mu(w^+)$ fulfills $\mu(w^+) = (1-\alpha)\mu$ for any $\alpha \in \mathbb{R}$. For this purpose, we use the decomposition:

$$(x + \alpha\Delta x^{\mathrm{ll}})(s + \alpha\Delta s^{\mathrm{ll}}) = (1-\alpha)xs + \alpha(x + \Delta x^{\mathrm{ll}})(s + \Delta s^{\mathrm{ll}}) - \alpha(1-\alpha)\Delta x^{\mathrm{ll}}\Delta s^{\mathrm{ll}}. \tag{4.34}$$

Recall from Part (i) that there exists $\Delta x \in \bigtimes_{k=1}^{p} W_{\mathcal{J},k}$ and $\Delta s \in \bigtimes_{k=1}^{p} W^{\perp}_{\mathcal{J},k}$ as in (4.28) and (4.29) such that $\delta\Delta x^{\mathrm{ll}} + \delta^{-1}\Delta s = -\delta x$ and $\delta\Delta x + \delta^{-1}\Delta s^{\mathrm{ll}} = -\delta^{-1}s$. In particular, $x + \Delta x^{\mathrm{ll}} = -\delta^{-2}\Delta s$ and

$s + \Delta s^{\mathrm{ll}} = -\delta^2 \Delta x$. Noting that $\Delta x^{\mathrm{ll}} \perp \Delta s^{\mathrm{ll}}$ and $\Delta x \perp \Delta s$, taking the average of the coordinates on both sides of (4.34), we get that

$$\begin{aligned}\mu(w + \alpha \Delta w^{\mathrm{ll}}) &= (1-\alpha)\mu(w) + \alpha\langle x + \Delta x^{\mathrm{ll}}, s + \Delta s^{\mathrm{ll}}\rangle/n - \alpha(1-\alpha)\langle \Delta x^{\mathrm{ll}}, \Delta s^{\mathrm{ll}}\rangle/n \\ &= (1-\alpha)\mu(w) + \alpha\langle \delta^{-2}\Delta s, \delta^2 \Delta x\rangle/n \\ &= (1-\alpha)\mu(w), \end{aligned} \tag{4.35}$$

as needed.

Let $\epsilon := \varepsilon^{\mathrm{ll}}(w)$. To obtain the desired lower bound on the step-length, given (4.35) it suffices to show that for all $0 \le \alpha < 1 - \frac{3\sqrt{n}\epsilon}{\beta}$ that

$$\left\| \frac{(x + \alpha \Delta x^{\mathrm{ll}})(s + \alpha \Delta s^{\mathrm{ll}})}{(1-\alpha)\mu} - e \right\| \le 2\beta\,. \tag{4.36}$$

We will need a bound on the product of the LLS residuals:

$$\begin{aligned}\left\| Rx^{\mathrm{ll}} Rs^{\mathrm{ll}} - \frac{1}{\mu}\Delta x^{\mathrm{ll}} \Delta s^{\mathrm{ll}} \right\| &= \left\| \frac{x^{1/2} s^{1/2}}{\sqrt{\mu}} \cdot \frac{\delta \Delta x^{\mathrm{ll}} + \delta^{-1}\Delta s^{\mathrm{ll}} + x^{1/2}s^{1/2}}{\sqrt{\mu}} \right\| \\ &\le 6(1+2\beta) n^{3/2}\lambda \le \frac{\beta}{4}\,, \end{aligned} \tag{4.37}$$

using Proposition 4.3.1, part (i), and the assumptions $\lambda \le \beta/(32n^2)$, $\beta \le 1/4$. Another useful bound will be

$$\begin{aligned}\|Rx^{\mathrm{ll}} Rs^{\mathrm{ll}}\|^2 &= \sum_{i\in[n]} \left|Rx_i^{\mathrm{ll}}\right|^2 \left|Rs_i^{\mathrm{ll}}\right|^2 \le \epsilon^2 \sum_{i\in[n]} \max\left\{ \left|Rx_i^{\mathrm{ll}}\right|^2, \left|Rs_i^{\mathrm{ll}}\right|^2 \right\} \\ &\le \epsilon^2 (\|Rx^{\mathrm{ll}}\|^2 + \|Rs^{\mathrm{ll}}\|^2) \le 2n\epsilon^2\,. \end{aligned} \tag{4.38}$$

The last inequality uses part (iii). With (4.34) we are ready to get the bound in (4.36), as

$$\begin{aligned}\left\| \frac{(x + \alpha \Delta x^{\mathrm{ll}})(s + \alpha \Delta s^{\mathrm{ll}})}{(1-\alpha)\mu} - \mathbf{1} \right\| &\le \beta + \left\| \frac{\alpha}{(1-\alpha)\mu}(x + \Delta x^{\mathrm{ll}})(s + \Delta s^{\mathrm{ll}}) - \frac{\alpha}{\mu}\Delta x^{\mathrm{ll}} \Delta s^{\mathrm{ll}} \right\| \\ &= \beta + \left\| \left(\frac{\alpha}{1-\alpha} - \alpha\right) Rx^{\mathrm{ll}} Rs^{\mathrm{ll}} + \alpha\left(Rx^{\mathrm{ll}} Rs^{\mathrm{ll}} - \frac{1}{\mu}\Delta x^{\mathrm{ll}} \Delta s^{\mathrm{ll}} \right) \right\| \\ &\le \beta + \frac{\alpha^2}{1-\alpha}\|Rx^{\mathrm{ll}} Rs^{\mathrm{ll}}\| + \alpha \left\| Rx^{\mathrm{ll}} Rs^{\mathrm{ll}} - \frac{1}{\mu}\Delta x^{\mathrm{ll}} \Delta s^{\mathrm{ll}} \right\| \\ &\le \beta + \frac{\sqrt{2n}\epsilon}{1-\alpha} + \frac{\beta}{4} \le \frac{5}{4}\beta + \frac{\sqrt{2n}\epsilon}{1-\alpha}\,. \end{aligned}$$

This value is $\le 2\beta$ whenever $2\sqrt{n}\epsilon/(1-\alpha) \le (3/4)\beta \Leftarrow \alpha < 1 - \frac{3\sqrt{n}\epsilon}{\beta}$, as needed.

Part (v). From part (iv), it is immediate that $\epsilon^{\mathrm{ll}}(w) = 0$ implies $\alpha = 1$. If $\alpha = 1$, we have that $w + \Delta w^{\mathrm{ll}}$ is the limit of (strictly) feasible solutions to (LP) and thus is also a feasible solution. Optimality of $w + \Delta w^{\mathrm{ll}}$ now follows from Part (iv), since $\alpha = 1$ implies $\mu(w + \Delta w^{\mathrm{ll}}) = 0$. The remaining implication is that if $w + \Delta w^{\mathrm{ll}}$ is optimal, then $\epsilon^{\mathrm{ll}}(w) = 0$. Recall that $Rx_i^{\mathrm{ll}} = \delta_i(x_i + \Delta x_i^{\mathrm{ll}})/\sqrt{\mu}$ and $Rs_i^{\mathrm{ll}} = \delta_i^{-1}(s_i + \Delta s_i^{\mathrm{ll}})/\sqrt{\mu}$. The optimality of $w + \Delta w^{\mathrm{ll}}$ means that for each $i \in [n]$, either $x_i + \Delta x_i^{\mathrm{ll}} = 0$ or $s_i + \Delta s_i^{\mathrm{ll}} = 0$. Therefore, $\epsilon^{\mathrm{ll}}(w) = 0$. □

4.6 Proofs of the main lemmas for the potential analysis

Lemma 4.4.2 (Restatement). *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$ and let $w^* = (x^*, y^*, s^*)$ be the optimal solution corresponding to $\mu^* = 0$ on the central path. Let further $\mathcal{J} = (J_1, \dots, J_p)$ be a $\delta(w)$-balanced layering (Definition 4.3.13), and let $\Delta w^{\mathrm{ll}} = (\Delta x^{\mathrm{ll}}, \Delta y^{\mathrm{ll}}, \Delta s^{\mathrm{ll}})$ be the corresponding LLS direction. Then the following statement holds for every $q \in [p]$:*

(i) There exists $i \in J_q$ such that

$$x_i^* \geq \frac{2x_i}{3\sqrt{n}} \cdot (\|Rx^{\mathrm{ll}}_{J_q}\| - 2\gamma n)\,. \tag{4.25}$$

(ii) There exists $j \in J_q$ such that

$$s_j^* \geq \frac{2s_j}{3\sqrt{n}} \cdot (\|Rs^{\mathrm{ll}}_{J_q}\| - 2\gamma n)\,. \tag{4.26}$$

Proof of Lemma 4.4.2. We prove part (i); part (ii) follows analogously using Lemma 4.3.9. Let z be a vector fulfilling the statement of Lemma 4.5.1 for $u = x^*$, $v = x + \Delta x^{\mathrm{ll}}$, and $I = J_{>q}$. Then $z \in W + x$, $z_{J_{>q}} = x_{J_{>q}} + \Delta x^{\mathrm{ll}}_{J_{>q}}$ and by $\ell^\delta(\mathcal{J}) \leq \gamma$

$$\left\|\delta_{J_{\leq q}}(x^*_{J_{\leq q}} - z_{J_{\leq q}})\right\| \leq \gamma \left\|\delta_{J_{>q}}\left(x^*_{J_{>q}} - (x_{J_{>q}} + \Delta x^{\mathrm{ll}}_{J_{>q}})\right)\right\|.$$

Restricting to the components in J_q, and dividing by $\sqrt{\mu}$, we get

$$\left\|\frac{\delta_{J_q}(x^*_{J_q} - z_{J_q})}{\sqrt{\mu}}\right\| \leq \gamma \left\|\frac{\delta_{J_{>q}}\left(x^*_{J_{>q}} - (x_{J_{>q}} + \Delta x^{\mathrm{ll}}_{J_{>q}})\right)}{\sqrt{\mu}}\right\| \leq \gamma \left\|\frac{\delta_{J_{>q}} x^*_{J_{>q}}}{\sqrt{\mu}}\right\| + \gamma \|Rx^{\mathrm{ll}}_{J_{>q}}\|\,. \tag{4.39}$$

Since $w \in \mathcal{N}(\beta)$, from Proposition 4.3.1 and (4.9) we see that for $i \in [n]$

$$\frac{\delta_i}{\sqrt{\mu}} \leq \frac{1}{\sqrt{1-2\beta}} \cdot \frac{\delta_i(w(\mu))}{\sqrt{\mu}} = \frac{1}{\sqrt{1-2\beta}} \cdot \frac{1}{x_i(\mu)}\,,$$

and therefore

$$\left\|\frac{\delta_{J_{>q}} x^*_{J_{>q}}}{\sqrt{\mu}}\right\| \leq \frac{1}{\sqrt{1-2\beta}} \left\|x(\mu)^{-1}_{J_{>q}} x^*_{J_{>q}}\right\| \leq \frac{1}{\sqrt{1-2\beta}} \cdot \left\|x(\mu)^{-1}_{J_{>q}} x^*_{J_{>q}}\right\|_1 \leq \frac{n}{\sqrt{1-2\beta}}\,,$$

where the last inequality follows by Lemma 4.3.3.

Using the above bounds with (4.39), along with $\|Rx^{\mathrm{ll}}_{J_{\geq q}}\| \leq \|Rx^{\mathrm{ll}}\| \leq \sqrt{2n}$ from Lemma 4.3.10(iii), we get

$$\left\|\frac{\delta_{J_q} z_{J_q}}{\sqrt{\mu}}\right\| \leq \left\|\frac{\delta_{J_q} x^*_{J_q}}{\sqrt{\mu}}\right\| + \frac{\gamma n}{\sqrt{1-2\beta}} + \gamma\sqrt{2n} \leq \left\|\frac{\delta_{J_q} x^*_{J_q}}{\sqrt{\mu}}\right\| + 2\gamma n\,,$$

using that $\beta \leq 1/8$ and $n \geq 3$. Note that z is a feasible solution to the least-squares problem which is optimally solved by $x^{\mathrm{ll}}_{J_q}$ for layer J_q and so

$$\|Rx^{\mathrm{ll}}_{J_q}\| \leq \left\|\frac{\delta_{J_q} z_{J_q}}{\sqrt{\mu}}\right\|.$$

It follows that

$$\left\| \frac{\delta_{J_q} x^*_{J_q}}{\sqrt{\mu}} \right\| \geq \|Rx^{\mathrm{ll}}_{J_q}\| - 2\gamma n\,.$$

Let us pick $i = \arg\max_{t\in J_q} |\delta_t x^*_t|$. Using Proposition 4.3.2,

$$\frac{x^*_i}{x_i} \geq \frac{1}{1+\beta}\cdot\frac{\delta_i x^*_i}{\sqrt{\mu}} \geq \frac{\|Rx^{\mathrm{ll}}_{J_q}\| - 2\gamma n}{(1+\beta)\sqrt{n}} \geq \frac{2}{3\sqrt{n}}\cdot(\|Rx^{\mathrm{ll}}_{J_q}\| - 2\gamma n)\,,$$

completing the proof. □

Lemma 4.4.3 (Restatement). *Let $w = (x, y, s) \in \mathcal{N}(2\beta)$ for $\beta \in (0, 1/8]$, let $\mu = \mu(w)$ and $\delta = \delta(w)$. Let $i, j \in [n]$ and $2 \leq \tau \leq n$ such that for the optimal solution $w^* = (x^*, y^*, s^*)$, we have $x^*_i \geq \beta x_i/(2^{10}n^{5.5})$ and $s^*_j \geq \beta s_j/(2^{10}n^{5.5})$, and assume $\varrho^{\mu}(i,j) \geq -\tau$. After $O(\beta^{-1}\sqrt{n}\tau\log(\bar{\chi}^* + n))$ further iterations the duality gap μ' fulfills $\Psi^{\mu'}(i,j) \geq 2\tau$, and for every $\ell \in [n]\setminus\{i,j\}$, either $\Psi^{\mu'}(i,\ell) \geq 2\tau$, or $\Psi^{\mu'}(\ell,j) \geq 2\tau$.*

Proof of Lemma 4.4.3. Let us select a value μ' such that

$$\log\mu - \log\mu' \geq 5\tau\log\left(\frac{4n\kappa^*}{\gamma}\right) + 31\log n + 44 - 4\log\beta\,.$$

The normalized duality gap decreases to such value within $O(\beta^{-1}\sqrt{n}\tau\cdot\log(\bar{\chi}^* + n))$ iterations, recalling that $\log(\bar{\chi}^* + n) = \Theta(\log(\kappa^* + n))$. The step-lengths for the affine scaling and LLS steps are stated in Proposition 4.3.4 and Lemma 4.3.10(iv). Whenever the algorithm chooses an LLS step, $\epsilon^{\mathrm{a}}(w) < 10n^{3/2}\gamma$. Thus, the progress in μ will be at least as much (in fact, much better) than the $1 - \beta/\sqrt{n}$ guarantee for the affine scaling step in Proposition 4.3.4.

Let $w' = (x', y', s')$ be the central path element corresponding to μ', and let $\delta' = \delta(w')$. From now on we use the shorthand notation

$$\Gamma := \log\left(\frac{4n\kappa^*}{\gamma}\right).$$

We first show that

$$\Gamma\varrho^{\mu'}(i,j) \geq 4\Gamma\tau + 18\log n + 22\log 2 - 2\log\beta \tag{4.40}$$

for μ', and therefore, $\Gamma\Psi^{\mu'}(i,j) \geq \min(2\Gamma n, 4\Gamma\tau + 18\log n + 22\log 2 - 2\log\beta) \geq 2\Gamma\tau$ as $\tau \leq n$. Recalling the definition $\kappa^{\delta}_{ij} = \kappa_{ij}\delta_j/\delta_i$, we see that according to Proposition 4.3.2,

$$\kappa^{\delta}_{ij} \leq \frac{\kappa_{ij}}{(1-\beta)^2}\cdot\frac{x_i s_j}{\mu}, \quad \text{and} \quad \kappa^{\delta'}_{ij} = \kappa_{ij}\cdot\frac{x'_i s'_j}{\mu'}\,.$$

Thus,

$$\begin{aligned}\Gamma\varrho^{\mu'}(i,j) &\geq \Gamma\varrho^{\mu}(i,j) + \log\mu - \log\mu' + 2\log(1-\beta) + \log x'_i - \log x_i + \log s'_j - \log s_j \\ &\geq \Gamma\varrho^{\mu}(i,j) + 5\Gamma\tau + 31\log n + 44 - 4\log\beta + 2\log(1-\beta) + \log x'_i - \log x_i + \log s'_j - \log s_j.\end{aligned}$$

Using the near-monotonicity of the central path (Lemma 5.2.3), we have $x'_i \geq x^*_i/n$ and $s'_j \geq s^*_j/n$. Together with our assumptions $x^*_i \geq \beta x_i/(2^{10}n^{5.5})$ and $s^*_i \geq \beta s_i/(2^{10}n^{5.5})$, we see that

$$\log x'_i - \log x_i + \log s'_j - \log s_j \geq -13\log n - 20\log 2 + 2\log\beta\,.$$

Using the assumption $\varrho^{\mu}(i,j) > -\tau$ of the lemma, we can establish (4.40) as $\beta < 1/8$.

Next, consider any $\ell \in [n] \setminus \{i, j\}$. From the triangle inequality Theorem 3.2.22(ii) it follows that $\kappa_{ij}^{\delta'} \le \kappa_{i\ell}^{\delta'} \cdot \kappa_{\ell j}^{\delta'}$, which gives $\varrho^{\mu'}(i,\ell) + \varrho^{\mu'}(\ell, j) \ge \varrho^{\mu'}(i, j)$. We therefore get

$$\max\{\Gamma\varrho^{\mu'}(i,\ell), \Gamma\varrho^{\mu'}(\ell, j)\} \ge \frac{1}{2}\Gamma\varrho^{\mu'}(i, j) \overset{(4.40)}{\ge} 2\Gamma\tau + 9\log n + 11\log 2 - \log\beta.$$

We next show that if $\Gamma\varrho^{\mu'}(i,\ell) \ge 2\Gamma\tau + 9\log n + 11\log 2 - \log\beta$, then $\Psi^{\mu'}(i,\ell) \ge 2\tau$. The case $\Gamma\varrho^{\mu'}(\ell, j) \ge 2\Gamma\tau + 9\log n + 11\log 2 - \log\beta$ follows analogously.

Consider any $0 < \bar{\mu} < \mu'$ with the corresponding central path point $\bar{w} = (\bar{x}, \bar{y}, \bar{s})$. The proof is complete by showing $\Gamma\varrho^{\bar{\mu}}(i,\ell) \ge \Gamma\varrho^{\mu'}(i,\ell) - 9\log n - 11\log 2 + \log\beta$. Recall that for central path elements, we have $\kappa_{ij}^{\delta'} = \kappa_{ij} x'_i / x'_j$, and $\kappa_{ij}^{\bar{\delta}} = \kappa_{ij}\bar{x}_i/\bar{x}_j$. Therefore

$$\Gamma\varrho^{\bar{\mu}}(i, j) = \Gamma\varrho^{\mu'}(i, j) + \log\bar{x}_i - \log x'_i - \log\bar{x}_j + \log x'_j\,.$$

Using Proposition 4.3.1, Lemma 4.3.3 and the assumption $x^*_i \ge \beta x_i/(2^{10}n^{5.5})$, we have $\bar{x}_j \le n x'_j$ and

$$\bar{x}_i \ge \frac{x^*_i}{n} \ge \frac{\beta x_i}{2^{10}n^{6.5}} \ge \frac{\beta(1-\beta)x'_i}{2^{10}n^{7.5}} \ge \frac{\beta x'_i}{2^{11}n^{7.5}}\,.$$

Using these bounds, we get

$$\Gamma\varrho^{\bar{\mu}}(i, j) \ge \Gamma\varrho^{\mu'}(i, j) - 9\log n - 11\log 2 + \log\beta,$$

completing the proof. □

It remains to prove Lemma 4.4.4 and Lemma 4.4.5, addressing the more difficult case $\xi^{\mathrm{ll}}_{\mathcal{J}} < 4\gamma n$. It is useful to decompose the variables into two sets. We let

$$\mathfrak{B} := \left\{ t \in [n] : |Rs^{\mathrm{ll}}_t| < 4\gamma n \right\}, \quad \text{and} \quad \mathfrak{R} := \left\{ t \in [n] : |Rx^{\mathrm{ll}}_t| < 4\gamma n \right\}. \tag{4.41}$$

The assumption $\xi^{\mathrm{ll}}_{\mathcal{J}} < 4\gamma n$ implies that for every layer J_k, either $J_k \subseteq \mathfrak{B}$ or $J_k \subseteq \mathfrak{R}$. The next two lemmas describe the relations between δ and δ^+.

Lemma 4.6.1. *Let $w \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$, and assume $\ell^\delta(\mathcal{J}) \le \gamma$ and $\epsilon^{\mathrm{ll}}(w) < 4\gamma n$. For the next iterate $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$, we have*

(i) For $i \in \mathfrak{B}$,

$$\frac{1}{2}\cdot\sqrt{\frac{\mu^+}{\mu}} \le \frac{\delta_i^+}{\delta_i} \le 2\cdot\sqrt{\frac{\mu^+}{\mu}} \quad \text{and} \quad \delta_i^{-1}s_i^+ \le \frac{3\mu^+}{\sqrt{\mu}}\,.$$

(ii) For $i \in \mathfrak{R}$,

$$\frac{1}{2}\cdot\sqrt{\frac{\mu}{\mu^+}} \le \frac{\delta_i^+}{\delta_i} \le 2\cdot\sqrt{\frac{\mu}{\mu^+}} \quad \text{and} \quad \delta_i x_i^+ \le \frac{3\mu^+}{\sqrt{\mu}}\,.$$

(iii) If $i, j \in \mathfrak{B}$ or $i, j \in \mathfrak{R}$, then

$$\frac{1}{4} \le \frac{\kappa_{ij}^{\delta}}{\kappa_{ij}^{\delta^+}} = \frac{\delta_i^+\delta_j}{\delta_i\delta_j^+} \le 4\,.$$

(iv) If $i \in \mathfrak{R}$ and $j \in \mathfrak{B}$, then

$$\frac{\kappa_{ij}^{\delta}}{\kappa_{ij}^{\delta^+}} \ge 4n^{3.5}\,.$$

Proof. **Part (i).** By Lemma 4.3.10(i), we see that

$$\begin{aligned}\|\delta_B \Delta x_B^{\mathrm{ll}}\|_\infty &\le \|\delta_B \Delta x_B^{\mathrm{ll}} + \delta_B^{-1}\Delta s_B^{\mathrm{ll}} + x_B^{1/2} s_B^{1/2}\|_\infty + \|\delta_B^{-1}(\Delta s_B^{\mathrm{ll}} + s_B)\|_\infty \\ &= \|\delta_B \Delta x_B^{\mathrm{ll}} + \delta_B^{-1}\Delta s_B^{\mathrm{ll}} + x_B^{1/2} s_B^{1/2}\|_\infty + \sqrt{\mu}\|Rs_B^{\mathrm{ll}}\|_\infty \\ &\le \sqrt{\mu}\Big(6n\ell^\delta(\mathcal{J}) + 4n\gamma\Big) \le 10n\gamma\sqrt{\mu} \le \sqrt{\mu}/64\,,\end{aligned}$$

by the assumption on $\ell^\delta(\mathcal{J})$ and the definition of $\mathfrak{B}$.

By construction of the LLS step, $|x_i^+ - x_i| = \alpha^+|\Delta x_i^{\mathrm{ll}}| \le |\Delta x_i^{\mathrm{ll}}|$, recalling that $0 \le \alpha^+ \le 1$. Using the bound derived above, for $i \in \mathfrak{B}$ we get

$$\left|\frac{x_i^+}{x_i} - 1\right| \le \left|\frac{\Delta x_i^{\mathrm{ll}}}{x_i}\right| = \frac{|\delta_i \Delta x_i^{\mathrm{ll}}|}{\delta_i x_i} \le \frac{\sqrt{\mu}}{64\delta_i x_i} \le \frac{1}{32}\,,$$

where the last inequality follows from Proposition 4.3.2. As

$$\frac{\delta_i^+}{\delta_i} = \sqrt{\frac{x_i^+ s_i^+}{x_i s_i}} \cdot \frac{x_i}{x_i^+} \quad \text{and} \quad \frac{1-2\beta}{1+\beta}\frac{\sqrt{\mu^+}}{\sqrt{\mu}} \le \sqrt{\frac{x_i^+ s_i^+}{x_i s_i}} \le \frac{1+2\beta}{1-\beta}\frac{\sqrt{\mu^+}}{\sqrt{\mu}}$$

by Proposition 4.3.2 the claimed bounds follow with $\beta \le 1/8$.

To get the upper bound on $\delta_i^{-1}s_i^+$, again with Proposition 4.3.2

$$\delta_i^{-1}s_i^+ = \frac{\delta_i^+}{\delta_i \delta_i^+}s_i^+ = \frac{\delta_i^+}{\delta_i} \cdot \sqrt{x_i^+ s_i^+} \le 2\sqrt{\frac{\mu^+}{\mu}} \cdot (1+2\beta)\sqrt{\mu^+} \le \frac{3\mu^+}{\sqrt{\mu}}\,.$$

Part (ii). Analogously to (i).

Part (iii). Immediate from parts (i) and (ii).

Part (iv). Follows by parts (i) and (ii), and by the lower bound on $\sqrt{\mu/\mu^+}$ obtained from Lemma 4.3.10(iv) as follows

$$\frac{\kappa_{ij}^\delta}{\kappa_{ij}^{\delta^+}} = \frac{\delta_i^+ \delta_j}{\delta_i \delta_j^+} \ge \frac{\mu}{4\mu^+} = \frac{1}{4(1-\alpha^+)} \ge \frac{\beta}{12\sqrt{n}\epsilon^{\mathrm{ll}}(w)} \ge 4n^{3.5}.$$ □

Lemma 4.4.4 (Restatement). *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$, and let $\mathcal{J} = (J_1, \dots, J_p)$ be a $\delta(w)$-balanced partition. Assume that $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) < 4\gamma n$, and let $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$ be the next iterate obtained by the LLS step with $\mu^+ = \mu(w^+)$ and assume $\mu^+ > 0$. Let $q \in [p]$ such that $\xi_{\mathcal{J}}^{\mathrm{ll}}(w) = \xi_{J_q}^{\mathrm{ll}}(w)$. If $\ell^{\delta^+}(\mathcal{J}) \le 4\gamma n$, then there exist $i, j \in J_q$ such that $x_i^* \ge \beta x_i^+/(16n^{3/2})$ and $s_j^* \ge \beta s_j^+/(16n^{3/2})$. Further, for any $\ell, \ell' \in J_q$, we have $\varrho^{\mu^+}(\ell, \ell') \ge -|J_q|$.*

Proof of Lemma 4.4.4. Without loss of generality, let $\xi_{\mathcal{J}}^{\mathrm{ll}} = \xi_{J_q}^{\mathrm{ll}} = \|Rx_{J_q}^{\mathrm{ll}}\|$ for a layer q with $J_q \subseteq \mathfrak{R}$. The case $\xi_{J_q}^{\mathrm{ll}} = \|Rs_{J_q}^{\mathrm{ll}}\|$ and $J_q \subseteq \mathfrak{B}$ can be treated analogously.

By Lemma 4.3.10(iii), $\|Rs_{J_q}^{\mathrm{ll}}\| \ge \frac{1}{2} - \frac{3}{4}\beta > \frac{1}{4} + 2n\gamma$, and therefore Lemma 4.4.2 provides a $j \in J_q$ such that $s_j^*/s_j \ge 1/(6\sqrt{n})$. Using Lemma 5.2.3 and Proposition 4.3.1 we find that $s_j^+/s_j \le 2n$ and so $s_j^*/s_j^+ = s_j^*/s_j \cdot s_j/s_j^+ \ge 1/(12n^{3/2}) > 1/(16n^{3/2})$.

The final statement $\varrho^{\mu^+}(\ell, \ell') \geq -|J_q|$ for any $\ell, \ell' \in J_q$ is also straightforward. From Lemma 4.6.1(iii) and the strong connectivity of J_q in $G_{\delta,\gamma/n}$, we obtain that J_q is strongly connected in $G_{\delta^+,\gamma/(4n)}$. Hence, $\varrho^{\mu^+}(\ell, \ell') \geq -|J_q|$ follows by Lemma 4.4.1.

The rest of the proof is dedicated to showing the existence of an $i \in J_q$ such that $x^*_i \geq \beta x^+_i/(16n^{3/2})$. For this purpose, we will prove following claim.

Claim 1. $\|\delta_{J_q} x^*_{J_q}\| \geq \frac{\beta\mu^+}{8\sqrt{n\mu}}$.

In order to prove Claim 1, we define

$$z := (\delta^+)^{-1} L^{\delta^+}_{J_{>q}} \left(\delta^+_{J_{>q}} (x^*_{J_{>q}} - x^+_{J_{>q}})\right) \quad \text{and } w := x^* - x^+ - z\,,$$

as in Lemma 4.5.1. By construction, $w \in W$ and $w_{J_{>q}} = 0$. Thus, $w_{J_q} \in W_{\mathcal{J},q}$ as defined in Section 4.3.4.

Using the triangle inequality, we get

$$\|\delta_{J_q} x^*_{J_q}\| \geq \|\delta_{J_q}(x^+_{J_q} + w_{J_q})\| - \|\delta_{J_q} z_{J_q}\|\,. \tag{4.42}$$

We bound the two terms separately, starting with an upper bound on $\|\delta_{J_q} z_{J_q}\|$. Since $\ell^{\delta^+}(\mathcal{J}) \leq 4\gamma n$, we have with Lemma 4.5.1 that

$$\begin{aligned}
\left\|\delta^+_{J_q} z_{J_q}\right\| &\leq \ell^{\delta^+}(\mathcal{J}) \left\|\delta^+_{J_{>q}} \left(x^*_{J_{>q}} - x^+_{J_{>q}}\right)\right\| \\
&\leq 4n\gamma \left\|\delta^+_{J_{>q}} \left(x^*_{J_{>q}} - x^+_{J_{>q}}\right)\right\| \\
&= 4n\gamma \left\|\delta^+_{J_{>q}} x^+_{J_{>q}} \left(\frac{x^*_{J_{>q}}}{x^+_{J_{>q}}} - \mathbf{1}\right)\right\| \\
&\leq 4n\gamma \left(\|\delta^+ x^+\|_\infty \cdot \left\|\frac{x^*}{x^+}\right\|_1 + \sqrt{n\mu^+}\right) \\
&\leq 4n\gamma \left(\frac{3}{2}\sqrt{\mu^+} \cdot \frac{4}{3} n + \sqrt{n\mu^+}\right) \\
&\leq 16n^2\sqrt{\mu^+}\gamma,
\end{aligned} \tag{4.43}$$

where the penultimate inequality follows by Proposition 4.3.2 and Lemma 5.2.3. We can use this and Lemma 4.6.1(ii) to obtain

$$\|\delta_{J_q} z_{J_q}\| \leq \|\delta_{J_q}/\delta^+_{J_q}\|_\infty \cdot \|\delta^+_{J_q} z_{J_q}\| \leq \frac{32n^2\gamma\mu^+}{\sqrt{\mu}} \leq \frac{\beta\mu^+}{32n^3\sqrt{\mu}}\,, \tag{4.44}$$

using the definition of γ.

The first RHS term in (4.42) will be bounded as follows.

Claim 2. $\|\delta_{J_q}(x^+_{J_q} + w_{J_q})\| \geq \frac{1}{2}\sqrt{\mu}\xi^{\mathrm{ll}}_{\mathcal{J}}$.

Proof of Claim 2. We recall the characterization (4.18) of the LLS step $\Delta x^{\mathrm{ll}} \in W$. Namely, there exists $\Delta s \in W^\perp_{\mathcal{J},1} \times \cdots \times W^\perp_{\mathcal{J},q}$ that is the unique solution to $\delta^{-1}\Delta s + \delta\Delta x^{\mathrm{ll}} = -\delta x$. From the above, note that

$$\|\delta^{-1}_{J_q}\Delta s_{J_q}\| = \|\delta_{J_q}(x_{J_q} + \Delta x^{\mathrm{ll}}_{J_q})\| = \sqrt{\mu}\|Rx^{\mathrm{ll}}_{J_q}\| = \sqrt{\mu}\xi^{\mathrm{ll}}_{\mathcal{J}}\,.$$

From the Cauchy-Schwarz inequality,

$$\begin{aligned}\|\delta_{J_q}^{-1}\Delta s_{J_q}\| \cdot \|\delta_{J_q}(x^+_{J_q} + w_{J_q})\| &\geq \left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q}(x^+_{J_q} + w_{J_q})\right\rangle\right| \\ &= \left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q} x^+_{J_q}\right\rangle\right|.\end{aligned} \tag{4.45}$$

Here, we used that $\Delta s_{J_q} \in W^{\perp}_{\mathcal{J},q}$ and $w_{J_q} \in W_{\mathcal{J},q}$. Note that

$$x^+ = x + \alpha\Delta x^{\mathrm{ll}} = x + \Delta x^{\mathrm{ll}} - (1-\alpha)\Delta x^{\mathrm{ll}} = -\delta^{-2}\Delta s - (1-\alpha)\Delta x^{\mathrm{ll}}.$$

Therefore,

$$\begin{aligned}\left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q} x^+_{J_q}\right\rangle\right| &= \left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, -\delta_{J_q}^{-1}\Delta s_{J_q} - (1-\alpha)\delta_{J_q}\Delta x^{\mathrm{ll}}_{J_q}\right\rangle\right| \\ &\geq \|\delta_{J_q}^{-1}\Delta s_{J_q}\|^2 - (1-\alpha)\left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q}\Delta x^{\mathrm{ll}}_{J_q}\right\rangle\right|.\end{aligned}$$

By Lemma 4.5.2, there exists $\Delta\bar{x} \in W_{\mathcal{J},1} \times \cdots \times W_{\mathcal{J},p}$ such that $\|\delta_{J_q}(\Delta x^{\mathrm{ll}}_{J_q} - \Delta\bar{x}_{J_q})\| \leq 2n\ell^{\delta}(\mathcal{J})\sqrt{\mu}$. Therefore, using the orthogonality of Δs_{J_q} and $\Delta\bar{x}_{J_q}$, we get that

$$\left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q}\Delta x^{\mathrm{ll}}_{J_q}\right\rangle\right| = \left|\left\langle \delta_{J_q}^{-1}\Delta s_{J_q}, \delta_{J_q}(\Delta x^{\mathrm{ll}}_{J_q} - \Delta\bar{x}^{\mathrm{ll}}_{J_q})\right\rangle\right| \leq 2n\ell^{\delta}(\mathcal{J})\sqrt{\mu}\cdot\|\delta_{J_q}^{-1}\Delta s_{J_q}\|.$$

From the above inequalities, we see that

$$\|\delta_{J_q}(x^+_{J_q} + w_{J_q})\| \geq \|\delta_{J_q}^{-1}\Delta s_{J_q}\| - 2(1-\alpha)n\ell^{\delta}(\mathcal{J})\sqrt{\mu} = \sqrt{\mu}\xi^{\mathrm{ll}}_{\mathcal{J}} - 2(1-\alpha)n\ell^{\delta}(\mathcal{J})\sqrt{\mu}.$$

It remains to show $(1-\alpha)n\ell^{\delta}(\mathcal{J}) \leq \xi^{\mathrm{ll}}_{\mathcal{J}}/4$. From Lemma 4.3.10(iv), we obtain

$$(1-\alpha)n\ell^{\delta}(\mathcal{J}) \leq 3n^{3/2}\ell^{\delta}(\mathcal{J})\xi^{\mathrm{ll}}_{\mathcal{J}}\beta^{-1},$$

using $\xi^{\mathrm{ll}}_{\mathcal{J}} \geq \varepsilon^{\mathrm{ll}}$. The claim now follows by the assumption $\ell^{\delta}(\mathcal{J}) \leq \gamma$, and the choice of γ. ■

Proof of Claim 1. Using Lemma 4.3.10(iv),

$$\mu^+ \leq \frac{3\sqrt{n}\xi^{\mathrm{ll}}_{\mathcal{J}}\mu}{\beta},$$

implying $\|\delta_{J_q}(x^+_{J_q} + w_{J_q})\| \geq \beta\mu^+/(6\sqrt{n\mu})$ by Claim 2. Now the claim follows using (4.42) and (4.44). ■

By Lemma 4.6.1(ii), we see that

$$\|\delta_{J_q} x^+_{J_q}\| \leq \sqrt{n}\|\delta_{J_q} x^+_{J_q}\|_{\infty} \leq \frac{3\sqrt{n}\mu^+}{\sqrt{\mu}}.$$

Thus, the lemma follows immediately from Claim 1: for at least one $i \in J_q$, we must have

$$\frac{x^*_i}{x_i} \geq \frac{\|\delta_{J_q} x^*_{J_q}\|}{\|\delta_{J_q} x^+_{J_q}\|} \geq \frac{\beta}{24n} \geq \frac{\beta}{16n^{3/2}}.$$

□

Lemma 4.4.5 (Restatement). *Let $w = (x, y, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/8]$, and let $\mathcal{J} = (J_1, \ldots, J_p)$ be a $\delta(w)$-balanced partition. Assume that $\xi^{\mathrm{ll}}_{\mathcal{J}}(w) < 4\gamma n$, and let $w^+ = (x^+, y^+, s^+) \in \mathcal{N}(2\beta)$ be the next iterate obtained by the LLS step with $\mu^+ = \mu(w^+)$ and assume $\mu^+ > 0$. If $\ell^{\delta^+}(\mathcal{J}) > 4\gamma n$, then there exist*

two layers J_q and J_r and $i \in J_q$ and $j \in J_r$ such that $x^*_i \geq x^+_i/(8n^{3/2})$, and $s^*_j \geq s^+_j/(8n^{3/2})$. Further, $\varrho^{\mu^+}(i,j) \geq -|J_q \cup J_r|$, and for all $\ell, \ell' \in J_q \cup J_r$, $\ell \neq \ell'$ we have $\Psi^\mu(\ell,\ell') \leq |J_q \cup J_r|$.

Proof of Lemma 4.4.5. Recall the sets $\mathfrak{B}$ and $\mathfrak{R}$ defined in (4.41). The key is to show the existence of an edge

$$(i', j') \in E_{\delta^+,\gamma/(4n)} \quad \text{such that} \quad i' \in J_q \subseteq \mathfrak{B}, \quad j' \in J_r \subseteq \mathfrak{R}, \quad r < q\,. \tag{4.46}$$

Before proving the existence of such i' and j', we show how the rest of the statements follow. Note that $x^+ \leq (1-\beta)^{-1}(1+2\cdot 2\beta)nx \leq \frac{7}{4}nx$ by Lemma 4.3.3 and Proposition 4.3.1. Further, we have $\|Rx^{\mathrm{II}}_{J_q}\| - 2\gamma n \geq \frac{1}{2} - \frac{3}{4}\beta - 2\gamma n \geq \frac{2}{5}$ by Lemma 4.3.10 (iii). The existence of $i \in J_q$ such that $x^*_i \geq x^+_i/(8n^{3/2})$ now follows immediately from Lemma 4.4.2, as there is an $i \in J_q$ such that

$$x^*_i \geq \frac{2x_i}{3\sqrt{n}} \cdot (\|Rx^{\mathrm{II}}_{J_q}\| - 2\gamma n) \geq \frac{2}{3\sqrt{n}} \frac{4x^+_i}{7n} \frac{2}{5} \geq \frac{x^+_i}{8n^{3/2}}. \tag{4.47}$$

With analogous argumentation it can be shown that there exists $j \in J_r$ such that $s^*_j \geq s^+_j/(8n^{3/2})$. The other statements are that $\varrho^{\mu^+}(i,j) \geq -|J_q \cup J_r|$, and for each $\ell, \ell' \in J_q \cup J_r$, $\ell \neq \ell'$, $\Psi^\mu(\ell,\ell') \leq |J_q \cup J_r|$. According to Lemma 4.4.1, the latter is true (even with the stronger bound $\max\{|J_q|, |J_r|\}$) whenever $\ell, \ell' \in J_q$, or $\ell, \ell' \in J_r$, or if $\ell \in J_q$ and $\ell' \in J_r$. It is left to show the lower bound on $\varrho^{\mu^+}(i,j)$ and $\Psi^\mu(\ell,\ell') \leq |J_q \cup J_r|$ for $\ell' \in J_q$ and $\ell \in J_r$.

From Lemma 4.6.1(iii), we have that if $\ell, \ell' \in J_q \subseteq \mathfrak{B}$ or $\ell, \ell' \in J_r \subseteq \mathfrak{R}$, then $\kappa^\delta_{\ell\ell'}/4 \leq \kappa^{\delta^+}_{\ell\ell'}$. Hence, the strong connectivity of J_r and J_q in $G_{\delta,\gamma}$ implies the strong connectivity of these sets in $G_{\delta^+,\gamma/(4n)}$. Together with the edge (i', j'), we see that every $\ell' \in J_q$ can reach every $\ell \in J_r$ on a directed path of length $\leq |J_q \cup J_r| - 1$ in $G_{\delta^+,\gamma/(4n)}$. Applying Lemma 4.4.1 for this setting, we obtain $\Psi^\mu(\ell,\ell') \leq \varrho^{\mu^+}(\ell,\ell') \leq |J_q \cup J_r|$ for all such pairs, and also $\varrho^{\mu^+}(i,j) \geq -|J_q \cup J_r|$.

The rest of the proof is dedicated to showing the existence of i' and j' as in (4.46). We let $k \in [p]$ such that $\ell^{\delta^+}(J_{\geq k}) = \ell^{\delta^+}(\mathcal{J}) > 4n\gamma$. To simplify the notation, we let $I = J_{\geq k}$.

When constructing $\mathcal{J}$ in Layering$(\delta, \hat{\kappa})$, the subroutine Verify-Lift$(\mathrm{diag}(\delta)W, I, \gamma)$ was called for the set $I = J_{\geq k}$, with the answer 'pass'. Besides $\ell^\delta(I) \leq \gamma$, this guaranteed the stronger property that $\max_{ji} |\mathbf{B}_{ji}| \leq \gamma$ for the matrix $\mathbf{B}$ implementing the lift (see Remark 4.2.3).

Let us recall how this matrix $\mathbf{B}$ was obtained. The subroutine starts by finding a minimal $I' \subset I$ such that $\dim(\pi_{I'}(W)) = \dim(\pi_I(W))$. Recall that $\pi_{I'}(W) = \mathbb{R}^{I'}$ and $L^\delta_I(p) = L^\delta_{I'}(p_{I'})$ for every $p \in \pi_I(\mathrm{diag}(\delta)W)$.

Consider the optimal lifting $L^\delta_I : \pi_I(\mathrm{diag}(\delta)W) \to \mathrm{diag}(\delta)W$. We defined $B \in \mathbb{R}^{([n]\setminus I)\times I'}$ as the matrix sending any $q \in \pi_{I'}(\mathrm{diag}(\delta)W)$ to the corresponding vector $[L^\delta_{I'}(q)]_{[n]\setminus I}$. The column $\mathbf{B}_i$ can be computed as $[L^\delta_{I'}(e^i)]_{[n]\setminus I}$ for $e^i \in \mathbb{R}^{I'}$.

We consider the transformation

$$\bar{\mathbf{B}} \coloneqq \mathrm{diag}(\delta^+\delta^{-1})\mathbf{B}\mathrm{diag}\big((\delta^+_{I'})^{-1}\delta_{I'}\big).$$

This maps $\pi_{I'}(\mathrm{diag}(\delta^+)W) \to \pi_{[n]\setminus I}(\mathrm{diag}(\delta^+)W)$.

Let $z \in \pi_I(\mathrm{diag}(\delta^+)W)$ be the singular vector corresponding to the maximum singular value of $L^{\delta^+}_I$, namely, $\|[L^{\delta^+}_I(z)]_{[n]\setminus I}\| > 4n\gamma\|z\|$. Let us normalize z such that $\|z_{I'}\| = 1$. Thus,

$$\left\|[L^{\delta^+}_{I'}(z_{I'})]_{[n]\setminus I}\right\| > 4n\gamma\,.$$

Let us now apply $\bar{\mathbf{B}}$ to $z_{I'} \in \pi_{I'}(\mathrm{diag}(\delta^+)W)$. Since $L_I^{\delta^+}$ is the minimum-norm lift operator, we see that

$$\left\|\bar{\mathbf{B}} z_{I'}\right\| \geq \left\|[L_{I'}^{\delta^+}(z_{I'})]_{n\setminus I}\right\| > 4n\gamma\,.$$

We can upper bound the operator norm by the Frobenius norm $\|\bar{\mathbf{B}}\| \leq \|\bar{\mathbf{B}}\|_F = \sqrt{\sum_{ji} \bar{\mathbf{B}}_{ji}{}^2} \leq n \max_{ji} |\bar{B}_{ji}|$, and therefore

$$\max_{ji} |\bar{\mathbf{B}}_{ji}| > 4\gamma\,.$$

Let us fix $i' \in I'$ and $j' \in [n] \setminus I$ as the indices giving the maximum value of $\bar{\mathbf{B}}$. Note that $\bar{\mathbf{B}}_{j'i'} = \mathbf{B}_{j'i'}\delta^+_{j'}\delta_{i'}/(\delta^+_{i'}\delta_{j'})$.

Let us now use Lemma 3.3.9 for the pair i', j', the matrix $\mathbf{B}$ and the subspace $\mathrm{diag}(\delta)W$. Noting that $\mathbf{B}_{j'i'} = [L_{I'}^{\delta}(e^{i'})]_{j'}$, we obtain $\kappa^{\delta}_{i'j'} \geq |\mathbf{B}_{j'i'}|$. Now,

$$\kappa^{\delta^+}_{i'j'} = \kappa^{\delta}_{i'j'} \cdot \frac{\delta^+_{j'}\delta_{i'}}{\delta^+_{i'}\delta_{j'}} \geq |\mathbf{B}_{j'i'}| \cdot \frac{\delta^+_{j'}\delta_{i'}}{\delta^+_{i'}\delta_{j'}} = |\bar{\mathbf{B}}_{j'i'}| > 4\gamma\,. \tag{4.48}$$

The next claim finishes the proof.

Claim 4.6.1.1. *For i' and j' selected as above, (4.46) holds.*

Proof. $(i', j') \in E_{\delta^+,\gamma/(4n)}$ holds by (4.48). From the above, we have

$$|\mathbf{B}_{j'i'}| > 4\gamma \cdot \frac{\delta^+_{i'}\delta_{j'}}{\delta_{i'}\delta^+_{j'}}\,.$$

According to Remark 4.2.3, $|\mathbf{B}_{j'i'}| \leq \gamma$ follows since Verify-Lift($\mathrm{diag}(\delta)W, I, \gamma$) returned with 'pass'. We thus have

$$\frac{\delta^+_{i'}\delta_{j'}}{\delta_{i'}\delta^+_{j'}} < \frac{1}{4}.$$

Lemma 4.6.1 excludes the scenarios $i', j' \in \mathfrak{R}$, $i', j' \in \mathfrak{B}$, and $i' \in \mathfrak{R}$, $j' \in \mathfrak{B}$, leaving $i' \in \mathfrak{B}$ and $j' \in \mathfrak{R}$ as the only possibility. Therefore, $i' \in J_q \subseteq \mathfrak{B}$ and $j' \in J_r \subseteq \mathfrak{R}$. We have $r < q$ since $i \in I = J_{\geq k}$ and $j \in [n] \setminus I = J_{<k}$. □

4.7 Initialization

Our main algorithm (Algorithm 4.2 in Section 4.3.6), requires an initial solution $w^0 = (x^0, y^0, s^0) \in \mathcal{N}(\beta)$. In this section, we remove this assumption by adapting the initialization method of [VY96] to our setting.

We use the "big-M method", a standard initialization approach for path-following interior point methods that introduces an auxiliary system whose optimal solutions map back to the optimal solutions of the original system. The primal-dual system we consider is

$$\begin{aligned} \min\ & c^\top x + M\mathbf{1}^\top \underline{x} & \max\ & y^\top b + 2M\mathbf{1}^\top z \\ & \mathbf{A}x - \mathbf{A}\underline{x} = b & & \mathbf{A}^\top y + z + s = c \\ & x + \bar{x} = 2M\mathbf{1} & & z + \bar{s} = \mathbf{0} \\ & x, \bar{x}, \underline{x} \geq \mathbf{0} & & -\mathbf{A}^\top y + \underline{s} = M\mathbf{1} \\ & & & s, \bar{s}, \underline{s} \geq \mathbf{0}. \end{aligned} \tag{Init-LP}$$

The constraint matrix used in this system is

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & -\mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

The next lemma asserts that the $\bar{\chi}$ condition number of $\hat{\mathbf{A}}$ is not much bigger than that of $\mathbf{A}$ of the original system 1.1.

Lemma 4.7.1 ([VY96, Lemma 23]). $\bar{\chi}_{\hat{\mathbf{A}}} \leq 3\sqrt{2}(\bar{\chi} + 1)$.

We extend this bound for $\bar{\chi}^*$.

Lemma 4.7.2. $\bar{\chi}^*_{\hat{\mathbf{A}}} \leq 3\sqrt{2}(\bar{\chi}^*_{\mathbf{A}} + 1)$.

Proof. Let $\mathbf{D} \in \mathfrak{D}_n$ and let $\hat{\mathbf{D}} \in \mathfrak{D}_{3n}$ the matrix consisting of three copies of $\mathbf{D}$, i.e.

$$\hat{\mathbf{D}} = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{D} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{D} \end{bmatrix}.$$

Then

$$\hat{\mathbf{A}}\hat{\mathbf{D}} = \begin{bmatrix} \mathbf{AD} & -\mathbf{AD} & \mathbf{0} \\ \mathbf{D} & \mathbf{0} & \mathbf{D} \end{bmatrix}.$$

Row-scaling does not change $\bar{\chi}$ as the kernel of the matrix remains unchanged. Thus, we can rescale the last n rows of $\hat{A}\hat{D}$, to the identity matrix, i.e., multiplying by $\begin{bmatrix} \mathbf{I} & \mathbf{D}^{-1} \end{bmatrix}$ from the left hand side. We observe that

$$\bar{\chi}_{\hat{A}\hat{D}} = \bar{\chi}\left(\begin{bmatrix} \mathbf{AD} & -\mathbf{AD} & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{I} \end{bmatrix}\right) \leq 3\sqrt{2}(\bar{\chi}_{\mathbf{AD}} + 1)$$

where the inequality follows from Lemma 4.7.1. The lemma now readily follows as

$$\bar{\chi}^*_{\hat{\mathbf{A}}} = \inf\left\{ \bar{\chi}_{\hat{\mathbf{A}}\hat{\mathbf{D}}} : \mathbf{D} \in \mathfrak{D}_{3n} \right\} \leq \inf\left\{ 3\sqrt{2}(\bar{\chi}_{\mathbf{AD}} + 1) : \mathbf{D} \in \mathfrak{D}_n \right\} = 3\sqrt{2}(\bar{\chi}^*_{\mathbf{A}} + 1). \qquad \square$$

We show next that the optimal solutions of the original system are preserved for sufficiently large M. We let d be the min-norm solution to $\mathbf{A}x = b$, i.e., $d = \mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top)^{-1}b$.

Proposition 4.7.3. *Assume both primal and dual of System 1.1 are feasible, and* $M > \max\{(\bar{\chi}_{\mathbf{A}} + 1)\|c\|, \bar{\chi}_{\mathbf{A}}\|d\|\}$. *Every optimal solution* (x, y, s) *to System 1.1, can be extended to an optimal solution*

$(x, \underline{x}, \bar{x}, y, z, s, \underline{s}, \bar{s})$ *to* (Init-LP)*; and conversely, from every optimal solution* $(x, \underline{x}, \bar{x}, y, z, s, \underline{s}, \bar{s})$ *to* (Init-LP)*, we obtain an optimal solution* (x, y, s) *by deleting the auxiliary variables.*

Proof. If system System 1.1 is feasible, it admits a basic optimal solution (x^*, y^*, s^*) with basis B such that $\mathbf{A}_B x_B^* = b, x^* \geq \mathbf{0}, \mathbf{A}_B^\top y^* = c$ and $\mathbf{A}^\top y^* \leq c$. Using Proposition 3.3.1 we see that

$$\|x_B^*\| = \|\mathbf{A}_B^{-1} b\| = \|\mathbf{A}_B^{-1}\mathbf{A}d\| \leq \bar{\chi}_\mathbf{A}\|d\| < M\,, \tag{4.49}$$

and using that $\|\mathbf{A}\| = \|\mathbf{A}^\top\|$ we observe

$$\|\mathbf{A}^\top y^*\| = \|\mathbf{A}^\top \mathbf{A}_B^{-\top} c\| \leq \|\mathbf{A}^\top \mathbf{A}_B^{-\top}\|\|c\| = \|\mathbf{A}_B^{-1}\mathbf{A}\|\|c\| \leq \bar{\chi}\|c\| < M. \tag{4.50}$$

We can extend this solution to a solution of system (Init-LP) via setting $\bar{x}^* = 2M\mathbf{1} - x^*, \underline{x}^* = 0, z^* = \bar{s}^* = 0$ and $\underline{s}^* = M\mathbf{1} + \mathbf{A}^\top y^*$. Observe that $\bar{x}^* > 0$ and $\underline{s}^* > 0$ by (4.49) and (4.50). Furthermore observe that by complementary slackness this extended solution for (Init-LP) is an optimal solution. The property that $\underline{s}^* > 0$ immediately tells us that $\underline{x}$ vanishes for all optimal solutions of (Init-LP) and thus all optimal solutions of System 1.1 coincide with the optimal solutions of (Init-LP), with the auxiliary variables removed. □

The next lemma is from [MT03, Lemma 4.4]. Recall that $w = (x, y, s) \in \mathcal{N}(\beta)$ if $\|xs/\mu(w) - e\| \leq \beta$.

Lemma 4.7.4. *Let* $w = (x, y, s) \in \mathcal{P}^{++} \times \mathcal{D}^{++}$*, and let* $\nu > 0$*. Assume that* $\|xs/\nu - e\| \leq \tau$*. Then* $(1 - \tau/\sqrt{n})\nu \leq \mu(w) \leq (1 + \tau/\sqrt{n})\nu$ *and* $w \in \mathcal{N}(\tau/(1-\tau))$*.*

The new system has the advantage that we can easily initialize the system with a feasible solution in close proximity to central path:

Proposition 4.7.5. *We can initialize system* (Init-LP) *close to the central path with initial solution* $w^0 = (x^0, y^0, s^0) \in \mathcal{N}(1/8)$ *and parameter* $\mu(w^0) \approx M^2$ *if* $M > 15\max\{(\bar{\chi}_\mathbf{A} + 1)\|c\|, \bar{\chi}_\mathbf{A}\|d\|\}$*.*

Proof. The initialization follows along the lines of [VY96, Section 10]. We let d as above, and set

$$\begin{aligned}
&\bar{x}^0 = M\mathbf{1}, x^0 = M\mathbf{1}, \underline{x}^0 = M\mathbf{1} - d\\
&y^0 = 0, z^0 = -M\mathbf{1}\\
&\bar{s}^0 = M\mathbf{1}, s^0 = M\mathbf{1} + c, \underline{s}^0 = M\mathbf{1}.
\end{aligned}$$

This is a feasible primal-dual solution to system (Init-LP) with parameter

$$\mu^0 = (3n)^{-1}(\langle x^0, s^0\rangle + \langle \underline{x}^0, \underline{s}^0\rangle + \langle \bar{x}^0, \bar{s}^0\rangle) = (3n)^{-1}(3nM^2 + Mc^\top\mathbf{1} - Md^\top\mathbf{1}) \approx M^2\,.$$

We see that

$$\left\| \frac{1}{M^2}\begin{bmatrix} \bar{x}^0\bar{s}^0 \\ x^0 s^0 \\ \underline{x}^0\underline{s}^0 \end{bmatrix} - \mathbf{1} \right\|^2 = M^{-2}\|c\|^2 + M^{-2}\|d\|^2 \leq \frac{1}{9^2\bar{\chi}_\mathbf{A}^2} \leq \frac{1}{9^2}.$$

With Lemma 4.7.4 we conclude that $w^0 = (x^0, y^0, s^0) \in \mathcal{N}\left(\frac{1/9}{1-1/9}\right) = \mathcal{N}(1/8)$. □

Detecting infeasibility. To use the extended system (Init-LP), we still need to assume that both the primal and dual programs in System 1.1 are feasible. For arbitrary instances, we first need to check if this is the case, or conclude that the primal or the dual (or both) are infeasible.

This can be done by employing a two-phase method. The first phase decides feasibility by running (Init-LP) with data $(\mathbf{A}, b, 0)$ and $M > \bar{\chi}_{\mathbf{A}}\|d\|_1$. The objective value of the optimal primal-dual pair is 0 if and only if System 1.1 has a feasible solution. If the optimal primal/dual solution $(x^*, \underline{x}^*, \bar{x}^*, y^*, z^*, s^*, \underline{s}^*, \bar{s}^*)$ has positive objective value, we can extract an infeasibility certificate in the following way.

We can w.l.o.g. assume that x^* is supported on some basis B of $\mathbf{A}$. Note that the objective function of the primal is equivalent to $\|\underline{x}\|_1$. Therefore, $\|\underline{x}^*\|_1 \leq -\sum_{i:d_i<0} d_i \leq \|d\|_1$ and so $\|\underline{x}^*\| \leq \|d\|_1$. Due to the constraint $\mathbf{A}x^* - \mathbf{A}\underline{x}^* = b = \mathbf{A}d$ we get that

$$\|x^*\| = \|\mathbf{B}^{-1}\mathbf{A}(d + \underline{x}^*)\| \leq \|\mathbf{B}^{-1}\mathbf{A}\|(\|d\| + \|\underline{x}^*\|) \leq 2\bar{\chi}_{\mathbf{A}}\|d\|_1. \tag{4.51}$$

Therefore, if $M > \bar{\chi}_{\mathbf{A}}\|d\|_1$, then $\bar{x}^* = 2M\mathbf{1} - \|x^*\| > 0$ so by strong duality, $\bar{s}^* = 0$. From the dual, we conclude that $z^* = 0$, and therefore $\mathbf{A}^\top y^* \leq \mathbf{A}^\top y^* + s^* + z^* = c = \mathbf{0}$. On the other hand, by assumption the objective value of the dual is positive, and so $(y^*)^\top b \geq (y^*)^\top b + 2Me^\top z^* > 0$. Hence, y^* is the desired certificate.

Feasibility of the dual of System 1.1 can be decided by running (Init-LP) on data $(\mathbf{A}, 0, c)$ and $M > (\bar{\chi}_{\mathbf{A}} + 1)\|c\|$ with the same argumentation: Either the objective value of the dual is 0 and therefore the dual optimal solution $(y^*, z^*, \underline{s}^*, s^*, \bar{s}^*)$ corresponds to a feasible dual solution of System 1.1 or the objective value is negative and we extract a dual infeasibility certificate in the following way: For the optimal corresponding primal solution $(x^*, \underline{x}^*, \bar{x}^*)$ we have by assumption $\langle c, x^*\rangle \leq \langle c, x^*\rangle + \langle M\mathbf{1}, \underline{x}^*\rangle < 0$. Furthermore, w.l.o.g. the support of s^* is contained in a basis which allows us to conclude that $\underline{s}^* > \mathbf{0}$ and therefore $\underline{x}^* = \mathbf{0}$. So we have $\mathbf{A}x^* = \mathbf{0} + \mathbf{A}\underline{x}^* = \mathbf{0}$, which together with $\langle c, x^*\rangle < 0$ yields the certificate of dual infeasibility.

Finding the right value of M. While Algorithm 4.2 does not require any estimate on $\bar{\chi}^*$ or $\bar{\chi}$, the initialization needs to set $M \geq \max\{(\bar{\chi}_{\mathbf{A}} + 1)\|c\|, \bar{\chi}_{\mathbf{A}}\|d\|\}$ as in Proposition 4.7.3.

A straightforward guessing approach (attributed to J. Renegar in [VY96]) starts with a constant guess, say $\bar{\chi}_{\mathbf{A}} = 100$, constructs the extended system, and runs the algorithm. In case the optimal solution to the extended system does not map to an optimal solution of System 1.1, we restart with $\bar{\chi}_{\mathbf{A}} = 100^2$ and try again; we continue squaring the guess until an optimal solution is found.

This would still require a series of $\log\log \bar{\chi}_{\mathbf{A}}$ guesses, and thus, result in a dependence on $\bar{\chi}_{\mathbf{A}}$ in the running time. However, if we initially rescale our system using the near-optimal rescaling Theorem 3.4.7, then we can turn the dependence from $\bar{\chi}_{\mathbf{A}}$ to $\bar{\chi}^*_{\mathbf{A}}$. The overall iteration complexity remains $O(n^{2.5} \log n \log(\bar{\chi}^*_{\mathbf{A}} + n))$, since the running time for the final guess on $\bar{\chi}^*_{\mathbf{A}}$ dominates the total running time of all previous computations due to the repeated squaring.

An alternative approach, that does not rescale the system, is to use Theorem 3.4.7 to approximate $\bar{\chi}_{\mathbf{A}}$. In this case we repeatedly square a guess of $\bar{\chi}^*_{\mathbf{A}}$ instead of $\bar{\chi}_{\mathbf{A}}$ which takes $O(\log\log \bar{\chi}^*_{\mathbf{A}})$ iterations until our guess corresponds to a valid upper bound for $\bar{\chi}_{\mathbf{A}}$.

Note that either guessing technique can handle bad guesses gracefully. For the first phase, if neither a feasible solution to System 1.1 is returned nor a Farkas' certificate can be extracted, we have proof that the guess was too low by the above paragraph. Similarly, in phase two, when feasibility was decided in the affirmative for primal and dual, an optimal solution to (Init-LP) that corresponds to an infeasible solution to System 1.1 serves as a certificate that another squaring of the guess is necessary.

5 Interior Point Methods are not worse than Simplex

Whereas interior point methods provide polynomial-time linear programming algorithms, the running time bounds depend on bit-complexity or condition measures that can be unbounded in the problem dimension. This is in contrast with the simplex method that always admits an exponential bound. We introduce a new polynomial-time path-following interior point method where the number of iterations also admits a combinatorial upper bound $O(2^n n^{1.5} \log n)$ for an n-variable linear program in standard form. This complements previous work by [ABGJ18] that exhibited a family of instances where any path-following method must take exponentially many iterations.

The number of iterations of our algorithm is at most $O(n^{1.5} \log n)$ times the number of segments of any piecewise linear curve in the wide neighborhood of the central path. In particular, it matches the number of iterations of any path following interior point method up to this polynomial factor. The overall exponential upper bound derives from studying the 'max central path', a piecewise-linear curve with the number of pieces bounded by the total length of $2n$ shadow vertex simplex paths.

From the existence of a line segment in the wide neighborhood we derive strong implications on the structure of the corresponding segment of the central path. Our algorithm is able to detect this structure from the local geometry at the current iterate, and constructs a step direction that descends along this segment. The bound $O(n^{1.5} \log n)$ that applies for arbitrarily long line segments is derived from a combinatorial progress measure.

Our algorithm falls into the family of layered least squares interior point methods introduced by Vavasis and Ye [VY96]. In contrast to previous layered least squares methods that partition the kernel of the constraint matrix into coordinate subspaces, our method creates layers based on a general subspace providing more flexibility. Our result also implies the same bound on the number of iterations of the trust region interior point method by Lan, Monteiro, and Tsuchiya [LMT09].

This chapter is based on joint work with Daniel Dadush, Georg Loho, László A. Végh, and Xavier Allamigeon [All+22].

Contents

5.1 Introduction

In this chapter, we explore connections between interior point methods and the simplex method, the two most commonly used classes of algorithms for linear programming. We consider linear programming (LP) in the primal-dual form as in System 1.1. We let

$$\mathcal{P} = \{ x \in \mathbb{R}^n : \mathbf{A}x = b, x \geq \mathbf{0} \}, \quad \mathcal{D} = \left\{ s \in \mathbb{R}^n : \exists y : \mathbf{A}^\top y + s = c, s \geq \mathbf{0} \right\}$$

denote the primal and dual feasible regions. Our focus is on LP algorithms that find exact primal and dual optimal solutions.

Breakthrough developments in the the seventies and eighties led to the first polynomial-time algorithms for linear programming: the ellipsoid method by Khachiyan [Kha79], and interior point methods introduced by Karmarkar [Kar84]. The running time of these algorithms is poly(n, L), where L denotes the encoding-length L of the rational input $(\mathbf{A}, b, c)$ of LP.

Even though LPs with exponential encoding length do not frequently appear in practice, there are examples when the binary encoding is exponential yet one could efficiently implement arithmetic operations using a different encoding, see Megiddo [Meg82]. The net present value problem in project scheduling is a particular example of a natural optimization problem that can be reformulated as an LP of exponential encoding length, see Grinold [Gri72]. From a theoretical perspective, finding an interior point method with an absolute bound $f(n)$ on the number of iterations connects to the fundamental open question on finding a strongly polynomial algorithm to solve linear programming. Besides being polynomial time, such an algorithm must achieve a number of arithmetic operations in poly(m, n). This question takes its roots in the development of the simplex method, and appears in Smale's list of open problems for the 21$^{\text{st}}$ century [Sma98].

Interior point methods and the central path. Whereas the simplex method moves on the boundary of the feasible region $\mathcal{P}$, interior point methods (IPM) reach an optimal solution by iterating through the strict interior of $\mathcal{P}$. Path-following interior point methods are driven to an optimal point by following a smooth trajectory called the central path. Recall the most standard

setting [Ren88], where the latter is defined as the parametric curve $\mu \in (0,\infty) \mapsto z(\mu) := (x(\mu), s(\mu))$, where $x(\mu)$ and $(y(\mu), s(\mu))$ are the unique solutions to the system

$$\begin{aligned} \mathbf{A}x(\mu) &= b\,, \quad x(\mu) > \mathbf{0} \\ \mathbf{A}^\top y(\mu) + s(\mu) &= c\,, \quad s(\mu) > \mathbf{0} \\ x(\mu)_i s(\mu)_i &= \mu \quad \text{for all } \ i \in [n]\,. \end{aligned} \tag{5.1}$$

This system arises from the optimality conditions of convex problems obtained by penalizing the original linear programs with the logarithmic barrier, i.e., respectively adding terms of the form $-\mu \sum_{i=1}^n \log x_i$ and $\mu \sum_{i=1}^n \log s_i$ to the objective functions of the primal and dual LP. The weight of the penalty is given by the parameter $\mu > 0$. When $\mu \searrow 0$, the central path $z(\mu)$ converges to a pair of optimal solutions $(x^\star, s^\star)$ of LP, which can be easily deduced from the fact that the duality gap of $z(\mu)$ is given by $\langle c, x(\mu)\rangle - \langle b, y(\mu)\rangle = \langle x(\mu), s(\mu)\rangle = n\mu$. Accordingly, we define the quantity $\overline{\mu}(z) := \langle x, s\rangle / n$ for any feasible point $z = (x, s) \in \mathcal{P} \times \mathcal{D}$, which we refer to as the *normalized duality gap* of z.

Interior point methods iteratively compute approximations of the points on the central path associated with successive values of μ that decrease geometrically; at most $O(\sqrt{n}\log(\mu/\mu'))$ iterations are needed to decrease the normalized duality gap from μ to μ'. The iterations follow an improvement direction, e.g., a Newton step, while remaining in a certain neighborhood of the central path, and can be implemented in polynomial time. The classical analysis yields a running time $O(n^{3.5}L)$ for solving LP for a rational input $(\mathbf{A}, b, c)$ of total encoding length L. There have been significant improvements in recent years both for general LP as well as for special classes, see Section 5.1.3.

A running time bound dependent on L requires a rational input; in contrast, the simplex method can be implemented in $2^n \text{poly}(n)$ even in the real model of computation. Whereas standard IPMs use bit-complexity arguments to terminate, they have also been extended to the real model of computation, e.g., by Vavasis and Ye [VY95]. The running time of such algorithms is parametrized by *condition numbers* that capture geometric properties of the input. In a remarkable paper, Vavasis and Ye [VY96] introduced a *layered least squares (LLS)* interior point method that runs in $O(n^{3.5}\log(\bar{\chi}_\mathbf{A} + n))$ iterations, where $\bar{\chi}_\mathbf{A}$ is the Dikin–Stuart–Todd condition number associated with the kernel of $\mathbf{A}$ (but .independent of b and c). As a consequence, they also derive a structural characterization of the central path: there are at most $\binom{n}{2}$ 'short and curved' segments, possibly separated by 'long and straight' segments. The LLS directions are refined Newton steps that can traverse the latter segments.

Lan, Monteiro and Tsuchiya [LMT09] gave a scaling invariant trust region IPM taking $O(n^{3.5}\log(\bar{\chi}^*_\mathbf{A} + n))$ iterations. Here, $\bar{\chi}^*_\mathbf{A}$ is the minimum value of $\bar{\chi}_\mathbf{A}$ that can be achieved by any column rescaling. However, computing the step directions in this algorithm has a weakly polynomial dependence on the right hand side. In Chapter 4 we desribed the scaling invariant LLS algorithm of [DHNV20] with iteration bound $O(n^{2.5}\log(n)\log(\bar{\chi}^*_\mathbf{A} + n))$, where the step directions can be computed in strongly polynomial time, by solving linear systems. We discuss the literature on such IPM methods in more detail in Section 5.1.3.

Lower bounds on interior point methods. LLS methods provide strongly polynomial LP algorithms whenever $\bar{\chi}^*_\mathbf{A} \in 2^{\text{poly}(n)}$; this is always the case if the encoding-length of $\mathbf{A}$ is polynomially bounded. One may wonder if some variant of IPM could be strongly polynomial for all LPs. A negative answer to this question was given in recent work by Allamigeon, Benchimol, Gaubert, and Joswig: they used tropical geometry to build pathological linear programs on which the

number of iterations of IPM has to be exponential (in m, n) [ABGJ18; ABGJ21]. Their construction shows that, when the entries of $\mathbf{A}$, b, and c are of very different orders of magnitude, the central path can be significantly deformed to the boundary of the feasible set. Allamigeon, Gaubert and Vandame later extended this result to the broad class of path-following IPMs using any self-concordant barrier function [AGV22]. They exhibited a counterexample where the feasible set is an n-dimensional combinatorial cube and the shape of the central path is analogous to the simplex paths on pathological instances of LP for the simplex method, akin to the Klee–Minty cube [KM72].

The shadow vertex simplex rule. We introduce a new IPM in this chapter whose analysis can be related to the *shadow vertex simplex rule*. Originally dubbed 'parametric simplex' by Gass and Saaty [GS55], this is one of the most extensively analyzed simplex rules from a theoretical perspective. The shadow vertex rule was used in Borgwardt's average case analysis [Bor12] and in Spielman and Teng's smoothed analysis [ST04b]. The interested reader may refer to the recent survey for a detailed exposition [DH20].

Given a pointed polyhedron $\mathcal{P} \subseteq \mathbb{R}^n$ and two objectives $c^{(1)}, c^{(2)} \in \mathbb{R}^n$, the shadow vertex rule consists in iterating over the vertices of $\mathcal{P}$ successively maximizing the objectives $(1-\lambda)c^{(1)} + \lambda c^{(2)}$ as λ goes from 0 to 1. Under non-degeneracy assumptions, the vertices of the path correspond to those vertices of the two-dimensional projection $\left\{\left(\langle c^{(1)}, x\rangle, \langle c^{(2)}, x\rangle\right) : x \in \mathcal{P}\right\}$ that maximize some open interval of objectives $(1-\lambda)e^1 + \lambda e^2$, $\lambda \in [0,1]$ (where e^1 and e^2 stand for the unit vectors of $\mathbb{R}^2$). We denote by $S_{\mathcal{P}}(c^{(1)}, c^{(2)})$ the number of vertices of the projection of the simplex path in this two-dimensional projection; this corresponds to the number of non-degenerate pivots.

5.1.1 Contributions

The purpose of this work is to establish a natural connection between the complexity of IPM and that of the simplex method, and deduce combinatorial bounds on the number of iterations. To this end, we introduce an interior point method called IPM with subspace LLS (see Algorithm 5.1), whose number of iterations is, up to a factor $O(n^{1.5}\log n)$, bounded by the number of pieces of any piecewise linear curve contained a *wide neighborhood* of the central path. This neighborhood is defined as

$$\mathcal{N}^{-\infty}(\theta) := \{\, z = (x,s) \in \mathcal{P} \times \mathcal{D} : xs \geq (1-\theta)\overline{\mu}(z)\mathbf{1} \,\}\,, \qquad (0 < \theta < 1) \tag{5.2}$$

where $xs \in \mathbb{R}^n$ denotes the Hadamard-product and $\mathbf{1} \in \mathbb{R}^n$ is the n-dimensional all-one vector. Our algorithm will however navigate through the narrower ℓ_2-neighborhood of the central path:

$$\mathcal{N}(\beta) := \left\{ z = (x,s) \in \mathcal{P} \times \mathcal{D} : \left\| \frac{xs}{\overline{\mu}(z)} - \mathbf{1} \right\| \leq \beta \right\}, \qquad (0 < \beta < 1/4)\,. \tag{5.3}$$

Theorem 5.1.1. *Assume that there exists $\Gamma : (0, \mu_0) \to \mathcal{N}^{-\infty}(\theta)$, $\theta \in (0,1)$, a piecewise linear curve satisfying $\overline{\mu}(\Gamma(\mu)) = \mu$, $\forall \mu \in (0, \mu_0)$. Starting from any point $z^0 \in \mathcal{N}(\beta)$ such that $\overline{\mu}(z^0) \leq \mu_0$, the algorithm IPM with subspace LLS finds an optimal solution of LP in $O(n^{1.5}\log(\frac{n}{1-\theta})T)$, where T is the number of linear segments in Γ.*

At a high level, our strategy is to show that any 'somewhat straight' segment of the central path, corresponding to a single straight segment in the wide neighborhood $\mathcal{N}^{-\infty}(\theta)$, can be decomposed into at most n short segments of length $\mathrm{poly}(n/(1-\theta))$ (as measured by the ratio of the start and end parameter), where consecutive short segments are possibly separated by 'long and straight'

segments. To traverse the long and straight segments we develop a novel *subspace* LLS step, which generalizes prior LLS steps from coordinate subspaces to general ones. Before describing this in more details, we present two applications of Theorem 5.1.1.

An exponential upper bound on the number of iterations. The first application relies on a piecewise linear curve that we call the *max central path*, and that is related with $2n$ simplex paths. It is defined as the parametric curve $g \mapsto z^{\mathrm{m}}(g) := (x^{\mathrm{m}}(g), s^{\mathrm{m}}(g)) \in \mathbb{R}^{2n}_{+}$, where $x^{\mathrm{m}}_i(g)$ and $s^{\mathrm{m}}_i(g)$ are the optimal values of the following parametric LP, respectively:

$$\begin{array}{ll} \max\ x_i & \max\ s_i \\ \quad \mathbf{A}x = b\ ,\ x \geq \mathbf{0} & \quad \mathbf{A}^\top y + s = c\ ,\ s \geq \mathbf{0} \\ \quad \langle c, x\rangle \leq v^\star + g\ , & \quad \langle b, y\rangle \geq v^\star - g\ , \end{array} \tag{5.4}$$

where we denote by $v^\star$ the optimal value of LP. As we show in Section 5.5, the maps $x^{\mathrm{m}}_i(g)$ and $s^{\mathrm{m}}_i(g)$ are piecewise linear, and the number of pieces can be related to the complexity of the simplex method with the shadow vertex rule.

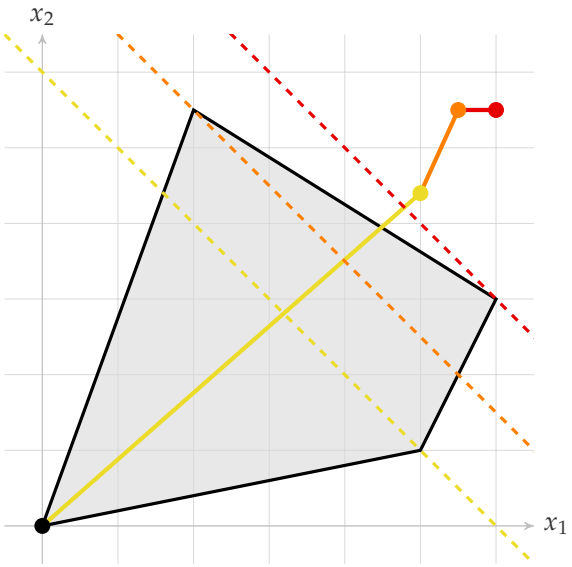


Figure 5.1: The max-central path in the coordinates (x_1, x_2) with cost function $x_1 + x_2$. Dashed lines correspond to level sets at breakpoints.

Recall that $(x^\star, s^\star)$ is the optimal solution of LP at the central path limit point, and $S_{\mathcal{P}}(-s^\star, e^i)$ and $S_{\mathcal{D}}(-x^\star, e^i)$ denote the number of nondegenerate pivots in the primal and dual shadow vertex paths for the indicated objective functions. We let $\mathcal{V}_{\mathcal{P}}$ and $\mathcal{V}_{\mathcal{D}}$ denote the number of vertices of the primal and dual feasible polytopes, respectively.

In Lemma 5.5.4, we show that the number of pieces in the max-central path is bounded by the sum of the quantities $S_{\mathcal{P}}(-s^\star, e^i)$ and $S_{\mathcal{D}}(-x^\star, e^i)$ for $i \in [n]$. Whereas the max-central path does not necessarily lie in the primal-dual feasible set $\mathcal{P} \times \mathcal{D}$, we prove that the corresponding shadow-vertex paths induce a piecewise linear curve in the wide neighborhood $\mathcal{N}^{-\infty}(\theta)$, for $\theta = 1 - \frac{1}{2n}$, with the same number of pieces; see Theorem 5.5.6 and an illustration in Figure 5.2. As a consequence, we obtain the following bound:

Theorem 5.1.2. *From any point $z^0 \in \mathcal{N}(\beta)$, the algorithm IPM with subspace LLS finds an optimal solution of LP in a number of iterations bounded by*

$$O\left(n^{1.5}\log n \ \min\left\{\sum_{i=1}^{n} S_{\mathcal{P}}(-s^\star, e^i) + S_{\mathcal{D}}(-x^\star, e^i), \mathcal{V}_{\mathcal{P}} + \mathcal{V}_{\mathcal{D}}\right\}\right).$$

This in particular implies an $O(2^n n^{1.5}\log n)$ iteration bound for IPM with subspace LLS. Theorem 5.1.2 thus complements the results of [ABGJ18; ABGJ21] by giving a singly exponential upper bound. We note that the max central path also plays an important if implicit role in the papers [ABGJ18; ABGJ21; AGV22], as it can be directly related to the tropical central path by the log-limit, see discussion in Section 5.1.3.

Theorem 5.1.2 assumes that a feasible starting point $z^0 \in \mathcal{N}(\beta)$ is given. This assumption can be removed e.g. by using the standard homogeneous self-dual embedding [Ye97, Section 5.3.1]. Then, the bounds in the theorem will refer to the shadow vertex paths and the number of vertices in the self-dual program.

Matching the complexity of any path following method. The second implication of Theorem 5.1.1 provides polynomial-time bounds in case the bit-complexity or a condition number such as $\bar{\chi}^*{}_{\mathbf{A}}$ is bounded. We show that—apart from a factor $O(n^{1.5}\log\frac{n}{1-\theta})$—the number of iterations of IPM with subspace LLS *is at most that of any IPM that stays in the wide neighborhood.*

Indeed, any IPM induces a piecewise linear curve formed by the line segments between the successive iterates. Already the wide neighborhood $\mathcal{N}^{-\infty}(1/2)$ is known to contain this piecewise linear curve for a large class of IPM based on the logarithmic barrier; we refer to [ABGJ18, Section 2] for a detailed discussion. We note that our algorithm matches—up to a polynomial factor—even any IPM that only stays in the extremely wide neighbourhood $\mathcal{N}^{-\infty}\left(1 - 1/2^{\text{poly}(n)}\right)$.

Theorem 5.1.3. *Suppose that an IPM reduces the duality gap from μ_0 to $\mu_1 \geq 0$ in T iterations staying throughout in the wide neighborhood $\mathcal{N}^{-\infty}(\theta)$ for some $0 < \theta < 1$. Then, from any point $z^0 \in \mathcal{N}(\beta)$ satisfying $\overline{\mu}(z^0) \leq \mu_0$, the algorithm IPM with subspace LLS finds a solution z^1 with $\overline{\mu}(z^1) \leq \mu_1$ in at most $O(n^{1.5}\log(\frac{n}{1-\theta})\,T)$ iterations.*

Comparison to the Trust Region IPM. IPM with subspace LLS also has an interesting relation to the Trust Region IPM algorithm by Lan, Monteiro, and Tsuchiya [LMT09]. The trust region steps are obtained as optimal solutions to primal and dual quadratic programs (5.6). These programs in essence capture the longest possible step achievable at the current point (up to a certain factor). However, solving these programs to sufficient accuracy requires weakly polynomial dependence on the input. Lan, Monteiro, and Tsuchiya show in [LMT09] that the number of iterations of the trust region algorithm can be bounded as $O(n^{3.5}\log(\bar{\chi}^*{}_{\mathbf{A}} + n))$, by adapting the analysis of the LLS methods [MT03; VY96].

The step directions used by our algorithm are feasible solutions to (5.6) for a suitable parameter. This implies that the steps of the Trust Region algorithm are always at least as long as the steps in our algorithm; as a consequence, the iteration bounds of our algorithm are also applicable to the Trust Region algorithm. Whereas any individual step of our algorithm could be arbitrarily worse than the one using the trust region step, Theorem 5.1.3 implies that overall we may only take $O(n^{1.5}\log n)$ more iterations. We emphasize that [LMT09] only provides the $\bar{\chi}^*{}_{\mathbf{A}}$ dependent iteration bound, and we do not see any obvious ways to obtain any $f(n)$ bound on their algorithm, other than comparing it to IPM with subspace LLS.

A further advantage of our algorithm is that the iterations can be implemented in strongly polynomial time, using simple projection steps. The description of this chapter requires a subspace V that can be obtained as a singular value subspace from a singular value decomposition. However, it suffices to compute rough approximations on the singular values, see Section 5.4.1.

5.1.2 Techniques

We now explain the key ideas behind the proof of Theorem 5.1.1.

Polarization of the Central Path

The first idea underlying to the proof is the following: every linear segment in the wide neighborhood gives rise to a *polarized segment* of the central path. A segment of the central path $\mathrm{CP}[\mu_1, \mu_0] := \{ z(\mu) : \mu \in [\mu_1, \mu_0] \}$, $0 \leq \mu_0 < \mu_1$, is polarized if it admits a partition $B \cup N = [n]$ such that the primal variables in B are essentially fixed and those in N are scaling down linearly with the parameter μ (vice versa for the dual variables). More precisely, $\forall \mu \in [\mu_1, \mu_0]$, we require

$$\begin{aligned} \gamma x_i(\mu_0) \leq x_i(\mu) \leq n x_i(\mu_0)\,, \quad \forall i \in B\,, \\ \frac{\mu}{n\mu_0} x_i(\mu_0) \leq x_i(\mu) \leq \frac{\mu}{\gamma\mu_0} x_i(\mu_0)\,, \quad \forall i \in N\,, \end{aligned} \tag{5.5}$$

where $\gamma \in [0, 1]$ is a *polarization parameter* (see Definition 5.3.1 and Corollary 5.3.3). By definition of the central path, the same relation holds for dual variables $s(\mu)$, $\mu \in [\mu_1, \mu_0]$, with the roles of N and B swapped. We note that the upper bound on $x_i(\mu)$ for $i \in B$ and the lower bound on $x_i(\mu)$ for $i \in N$ hold for any point of the central path by the near-monotonicity property (Lemma 5.2.3); the important parts of the definition are the other two bounds.

For simplicity of notation, let us restrict to line segments between two points on the central path. To relate polarization to the wide neighborhood, we show that if the line segment $[z(\mu_1), z(\mu_0)]$ between central path points is contained in the wide neighborhod $\mathcal{N}^{-\infty}(\theta)$, then the corresponding segment of the central path is polarized with $\gamma = \frac{(1-\theta)^2}{16n^3}$ with respect to some partition $B \cup N = [n]$ (see Lemma 5.3.5 for the general statement).

One should read this last statement as saying that segment of central path is 'approximately linear' if and only if it is polarized (in fact, a segment is 1-polarized if and only if it is linear, see Lemma 5.3.7). The link between polarization and linearity is surprisingly elementary; it follows from the analysis of the inequalities

$$((1-\alpha)x_i(\mu_0) + \alpha x_i(\mu_1))\,((1-\alpha)s_i(\mu_0) + \alpha s_i(\mu_1)) \geq (1-\theta)((1-\alpha)\mu_0 + \alpha\mu_1)\,,\ \forall \alpha \in [0,1]\,,\ i \in [n],$$

where we recall that $z(\mu_0) = (x(\mu_0), s(\mu_0)), z(\mu_1) = (x(\mu_1), s(\mu_1))$. For example, if $\theta = 0$, it is not hard to check that for each $i \in [n]$, one must have either $x_i(\mu_0) = x_i(\mu_1)$ and $s_i(\mu_1) = \frac{\mu_1}{\mu_0} s_i(\mu_0)$ (i.e., $i \in B$) or $x_i(\mu_0) = \frac{\mu_1}{\mu_0} x_i(\mu_1)$ and $s_i(\mu_1) = s_i(\mu_0)$ (i.e., $i \in N$).

Given the above, the main task in Theorem 5.1.1, namely traversing linear segments in the wide-neighborhood, can be reduced to traversing γ-polarized segments of the central path. The main guarantee of our algorithm IPM with subspace LLS is in fact that it can traverse any γ-polarized segment of the path in $O(n^{1.5} \log(n/\gamma))$ iterations.

We note that polarization plays an important if implicit role in prior layered least squares analyses [DHNV20; MT05; VY96]. In particular, the 'long and straight' segments in these works are all polarized. What was unclear in these works, however, is whether polarization *by itself* was sufficient to make a segment easy to traverse. Indeed, these works all crucially rely upon

numerical condition numbers of the instance which can be effectively unbounded in the present context. Beyond the LLS context, we are further unaware of central path analyses exploiting the tight connection between approximate linearity and polarization, and we hope this will encourage future study.

As is clear from the definition, polarization provides us extremely useful 'long-range' control over the evolution of variables on a segment. We note that γ-polarization is mostly interesting when the segment itself is *long*, namely, when $\mu_0/\mu_1 \gg 1/\gamma$. We now explain how to leverage this control to traverse any γ-polarized segment using *subspace LLS steps*.

Traversing a Polarized Segment

Let $\mathrm{CP}[\mu_1, \mu_0]$, $0 \le \mu_1 < \mu_0$, be a γ-polarized segment with partition $B \cup N = [n]$.

For simplicity of presentation, let us assume that given any iterate (x, s) in the narrow neighborhood $\mathcal{N}(1/6)$ used in our algorithm, we can jump to the exact central path point $z(\mu) \in \mathrm{CP} = \mathcal{N}(0)$ with $\mu = \overline{\mu}(x, s)$ for free. Let us further assume that the algorithm knows the partition B, N (we discuss how to effectively compute it at the end) and that we are given the starting point $z^{(0)} := z(\mu_0)$.

Our abstract algorithm will thus compute iterates $z^{(0)}, z^{(1)}, \ldots$ on the central path CP with $\overline{\mu}(z^{(0)}) > \overline{\mu}(z^{(1)}) > \ldots$. To move from $z^{(t)}$ to $z^{(t+1)}$, we first compute a *movement direction*

$$\Delta z^{(t)} = (\Delta x^{(t)}, \Delta s^{(t)}) \in \ker(\mathbf{A}) \times \mathrm{im}(\mathbf{A}^\top) =: W \times W^\perp,$$

together with a step-length $\alpha^{(t)} \in [0, 1]$, chosen such that $z^{(t)} + \alpha \Delta z^{(t)} \in \mathcal{N}(1/6), 0 \le \alpha \le \alpha^{(t)}$. Lastly, we jump for free to $z^{(t+1)} \in \mathrm{CP}$ satisfying $\overline{\mu}(z^{t+1}) = \overline{\mu}(z^{(t)} + \alpha^{(t)} \Delta z^{(t)})$.

Given this setup, our goal is to compute movement directions, such that after $k = O(n^{1.5} \log(n/\gamma))$ iterations, we have $\overline{\mu}(z^{(k)}) \le \mu_1$, i.e., that we have crossed the segment. We would like to emphasize that our algorithm will in fact compute the movement direction $\Delta z^{(t)}$ using only *local information* at $z^{(t)}$, without any explicit knowledge of the polarized segment.

A natural movement direction is *affine scaling* used in predictor-corrector methods, see Section 4.3.2. This direction guarantees $1 - \Omega(1/\sqrt{n})$ decrease in normalized gap per step. Hence, if $\mu_0/\mu_1 \le \mathrm{poly}(n, 1/\gamma)$, then simply using $\sqrt{n} \ln(\mu_0/\mu_1)$ affine scaling iterations is sufficient for our purposes.

Thus, we may assume that $\mu_0/\mu_1 \gg \mathrm{poly}(n, 1/\gamma)$. In this case, we show that computing an affine scaling direction $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ at the current iterate $(x^{(t)}, s^{(t)})$ reveals the correct partition $B \cup N = [n]$ of the current polarized segment. This is because the standard affine scaling step itself exhibits a polarized behaviour: we can simply select B as the set of coordinates i where $|\Delta x^{\mathrm{a}}_i / x^{(t)}_i| < |\Delta s^{\mathrm{a}}_i / s^{(t)}_i|$, i.e., the relative primal movement is smaller than the relative dual movement (see Definition 5.4.4).

Trust Region Programs and Subspace LLS. The trust region programs by Lan, Monteiro, Tsuchiya [LMT09] provide a good starting point for defining our movement direction $\Delta z^{(t)} = (\Delta x^{(t)}, \Delta s^{(t)}) \in W \times W^\perp$ from an iterate $z^{(t)} = (x^{(t)}, s^{(t)}) \in \mathrm{CP}[\mu_1, \mu_0]$ and a given a partition $[n] = B \cup N$:

$$\begin{aligned} &\min_{\Delta x \in W} \left\{ \|(x^{(t)}_N + \Delta x_N)/x^{(t)}_N\| : \|\Delta x_B / x^{(t)}_B\| \le \varrho \right\} \\ &\min_{\Delta s \in W^\perp} \left\{ \|(s^{(t)}_B + \Delta s_B)/s^{(t)}_B\| : \|\Delta s_N / s^{(t)}_N\| \le \varrho \right\} \end{aligned} \tag{5.6}$$

where $\varrho = 1/100$ is sufficient for our purposes. We use the notation $\Delta x / x^{(t)} := (\Delta x_1 / x^t_1, \ldots, \Delta x_n / x^{(t)}_n)$ and similarly for $\Delta s / s^{(i)}$. The norms $\|x/x^{(t)}\|$ and $\|s/s^{(t)}\|$ are the so-called primal and dual *local norms* at $x^{(t)}$ and $s^{(t)}$. By definition, the optimal primal trust region direction Δx^* achieves a

maximal multiplicative decrease on the coordinates in N while 'barely moving' the coordinates in B as measured in the local norm. The optimal dual direction Δs^* achieves the same on the dual side with the role of N and B swapped.

Note that these directions mesh very well with polarization of the segment CP$[\mu_1, \mu_0]$. In particular, they reflect the idea that the coordinates of $x(\mu)$ in N should be linearly scaling down while those in B are staying mostly fixed, and vice versa for $s(\mu)$. As shown in [LMT09], moving in any direction $\Delta z^{(t)} = (\Delta x^{(t)}, \Delta s^{(t)})$ corresponding to feasible solutions to (5.6), the normalized gap can be reduced as

$$\frac{\overline{\mu}(z^{(t+1)})}{\overline{\mu}(z^{(t)})} \leq \|(x_N^{(t)} + \Delta x_N^{(t)})/x_N^{(t)}\| + \|(s_B^{(t)} + \Delta s_B^{(t)})/s_B^{(t)}\|. \tag{5.7}$$

That is, we can achieve a drop that corresponds to the sum of primal and dual objective values.

In many ways, the trust region direction can be seen as the 'optimal' movement direction. However, [LMT09] solves the quadratic convex programs in (5.6) in weakly polynomial time with dependence on the vectors b and c in LP. It is not known whether a strongly polynomial algorithm (with dependence only on n) exists. Further, the analysis in [LMT09] relies on combinatorial progress measures adapted from the LLS analyses. It remains unclear how to analyze the convergence of the trust region steps by only using only the fact that they are maximally long.

Instead of optimally solving (5.6), we introduce what we call *subspace LLS steps* that yield 'good enough' approximate solutions for our purposes. We restrict the set of primal and dual directions to come from carefully selected subspaces $V^{(t)} \subseteq W$ and $U^{(t)} \subseteq W^\perp$ satisfying:

$$\|\Delta x_B / x_B^{(t)}\| \leq \tau \|\Delta x_N / x_N^{(t)}\|, \quad \forall \Delta x \in V^{(t)} \tag{5.8}$$

$$\|\Delta s_N / s_N^{(t)}\| \leq \tau \|\Delta s_B / s_B^{(t)}\|, \quad \forall \Delta s \in U^{(t)}, \tag{5.9}$$

where we set $\tau = \varrho/(16\sqrt{n})$. We call any such subspaces $U^{(t)}, V^{(t)}$ *cheap lift subspaces*. Note that every such solution automatically satisfies the constraints in program (5.6). Hence, the optimal solutions can be computed by solving systems of linear equations that correspond to minimum-norm points in the local norms.

In terms of the choice of subspaces, there is quite a lot of flexibility. A canonical choice, which we use for simplicity in the analysis, consists of choosing a space spanned by the singular vectors of a certain 'lifting map' whose corresponding singular values are at most τ. While singular values (and their corresponding spaces) are not computable in strongly polynomial time, one can indeed compute sufficiently good approximations for our purposes (see Section 5.4.1).

Analyzing Subspace LLS. At each iteration, our algorithm computes the affine scaling steps and the subspace LLS steps as above, and uses the one that enables more progress by chosing the largest possible step-length. We use the subspaces $U^{(t)}, V^{(t)}$ to compute the subspace LLS direction $\Delta z^{(t)}$ as above.

Let us now explain the key idea in showing that subspace LLS steps can reach the end of the current γ-polarized segment CP$[\mu_1, \mu_0]$ in $k = O(n^{1.5} \ln(n/\gamma))$ iterations. Let $k = \Omega(\sqrt{n} \ln(n/\gamma))$. Given any iterate $z^{(t)} \in$ CP$[\mu_1, \mu_0]$, if $\overline{\mu}(z^{(t+k)}) > \mu_1$—i.e., we have not reached the end of the segment—then we show that both $\dim(U^{(t+k)}) > \dim(U^{(t)})$ and $\dim(V^{(t+k)}) > \dim(V^{(t)})$. The overall bound follows since this can occur at most n times.

To get this result, we analyze the evolution of what we call the 'empirical gradient' at $z^{(t)}$, which we define to be $\Delta \tilde{z}^{(t)} := z(\mu_1) - z^{(t)}$, i.e., the difference between the current iterate and the end of the segment. A crucial observation is that if $\Delta \tilde{z}^{(t)}$ were a feasible solution to (5.6), then following this direction would get to within a poly(n/γ) factor for the end of the segment in one step (though

we do not know how to compute it). Furthermore, the empirical gradient is never far from being feasible, in particular, it is feasible if the bound of β is replaced by $O(n)$. We show the following dichotomy. Given an iteration $z^{(t)}$, either the empirical gradient $\Delta\tilde{z}^{(t)}$ is mostly "aligned" with the LLS subspaces $U^{(t)} \times V^{(t)}$, in which case we get close to the end of the segment in one step, or we can extract from $\Delta\tilde{z}^{(t)}$ an additional "cheap lift" dimension in the next $O(\sqrt{n}\ln(n/\gamma))$ iterations. In the latter case, we use the polarization property to analyze the evolution of the singular values of lifting maps.

This concludes our overview of the proof of Theorem 5.1.1. For more details, see Section 5.4.

5.1.3 Related Work

Interior points methods have been a tremendously active and fruitful research area since the seminal works of Karmarkar [Kar84] and Renegar [Ren88] in the 80's. Remarkable advances have been made both in speed as well as applicability of IPMs. We first briefly review works that—unlike the present chapter—aim for ε-approximate solutions. A key ingredient has been using different, self-concordant barrier functions. We refer to [NN94] for an introduction into self-concordant functions. Like the logarithmic barrier, every such function gives rise to a notion of central path. In the general setting, the iteration complexity to get an ε-approximation of the optimal value is bounded by $O(\vartheta^{1/2}\log\varepsilon^{-1})$, where ϑ is a complexity parameter specific to the barrier function. General bounds on self-concordant barriers were given by Nesterov and Nemirovski [NN94], improved recently by Lee and Yue [LY21]. Specific barrier functions include Vaidya's volumetric barrier [Vai89], the entropic barrier by Bubeck and Eldan [BE15], and the weighted log-barrier by Lee and Sidford [LS14; LS19].

Recent improvements make use of efficient data structures to amortize the cost of the iterative updates, and work with approximate computations, see Cohen, Lee and Song [CLS19], van den Brand [Bra20], and van den Brand, Lee, Sidford, and Song [BTSS20]. For special classes of LP such as network flow and matching problems, even faster algorithms have been obtained using, among other techniques, fast Laplacian solvers [ST04a], see e.g. [AMV22; Bra+20; Bra+21; DS08; GLP22; Mad13], culminating in the very recent near-linear time minimum-cost flow algorithm [Che+22].

Layered least squares IPMs, initiated by Vavasis and Ye [VY96] find exact optimal solutions and their running time bound is independent of b and c. Improved LLS algorithms were given by Megiddo, Mizuno, and Tsuchiya [MMT98] and Monteiro and Tsuchiya [MT03; MT05]. As discussed previously, scaling invariant algorithms with a $\bar{\chi}^*_{\mathbf{A}}$ dependence are the Trust Region algorithm by Lan, Monteiro, and Tsuchiya [LMT09], and the LLS algorithm [DHNV20] that relies on approximating circuit imbalances.

There is an interesting connection between IPMs and differential geometry. Sonnevend, Stoer, and Zhao [SSZ91] introduced a primal-dual curvature concept for the central path, and related the curvature integral to the iteration complexity of IPMs. Monteiro and Tsuchiya [MT08] showed that a curvature integral is bounded by $O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}} + n))$. This has been extended to SDP and symmetric cone programming [KOT14], and also studied in the context of information geometry [KOT13].

Relating the central path with a simplex path has been mainly used to build LP with pathological properties. On top of the construction of [AGV22] that we already discussed, Deza, Nematollahi and Terlaky [DNT08] built a Klee–Minty cube with exponentially many redundant inequalities where the central path is distorted into the neighbhorhood of the simplex path that visits the 2^n vertices.

The max central path studied in this chapter is related to the tropical central path in [ABGJ18; ABGJ21; AGV22]. The latter arises when studying parametric families of LP where the input $(\mathbf{A}, b, c)$ depend on a parameter $t > 1$. The tropical central path is defined as the log-limit, i.e., the limit as $t \to \infty$ of the image under the map $z \mapsto \log_t z = \frac{\log z}{\log t}$, of the central path of these LP. In [ABGJ18; ABGJ21; AGV22], it was shown that the tropical central path corresponds to the greatest point (entrywise) of the log-limit of the feasible sets of (5.4). This turns to be precisely the log-limit of the max central path.

5.1.4 Organization of the chapter

In Section 5.2, we recall some basic properties related to the central path and its neighborhoods (Section 5.2.1). We also discuss the affine scaling steps used in predictor-correct methods (Section 4.3.2). We finally introduce the lifting maps used in the subspace LLS step (Section 5.2.3). Section 5.3 deals with the polarized segments of the central path and their connection with linear segments in the wide neighborhood. Section 5.4 provides the complexity analysis of the algorithm IPM with subspace LLS for traversing polarized segments. Section 5.5 studies the max central path. We give a direct proof of the polarization of the central path along the max central path (Section 5.5.2), and an alternative proof via a piecewise linear curve in the wide neighborhood induced by the max central path (Section 5.5.3). Omitted proofs are deferred to the Appendix.

5.2 Preliminaries

In this section we reintroduce preliminaries to IPM. Many concepts will be familiar from Chapter 4, but the notation deviates.

5.2.1 Preliminaries on Interior-Point Methods

In this section, we recall standard properties of the central path and IPM that will be required for our algorithm. To ensure that the central path is well-defined, we assume that $\mathcal{P}$ and $\mathcal{D}$ admit strictly feasible solutions, i.e., the sets

$$\mathcal{P}_{++} := \{x \in \mathcal{P} : x > 0\}, \quad \mathcal{D}_{++} := \{s \in \mathcal{D} : s > 0\}$$

are both nonempty. We use $z^{\mathrm{cp}}(\mu) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$ to denote the central path point at μ rather than $z(\mu) = (x(\mu), s(\mu))$ used in the Introduction.

Given $z = (x, s) \in \mathcal{P} \times \mathcal{D}$, we recall that the normalized duality gap is defined as $\overline{\mu}(z) = \frac{\langle x, s \rangle}{n}$. The following identity is useful in comparing duality gaps:

Proposition 5.2.1. *Given $x, x' \in W + d$, $s, s' \in W^\perp + c$, we have that*

$$\langle x, s \rangle + \langle x', s' \rangle = \langle x, s' \rangle + \langle x', s \rangle.$$

In particular, if $\langle x', s' \rangle = 0$, then

$$\langle x, s \rangle = \langle x, s' \rangle + \langle x', s \rangle.$$

The next proposition shows that the normalized duality gap is a linear function for convex combinations of points.

Proposition 5.2.2 (Linearity duality gap). *Given $x^{(1)}, \dots, x^{(k)} \in W + d, s^{(1)}, \dots, s^{(k)} \in W^\perp + c$ forming the sequence $z^{(1)} = (x^{(1)}, s^{(1)}), \dots, z^{(k)} = (x^{(k)}, s^{(k)})$ and $\lambda \in \mathbb{R}^k$ such that $\sum_{i=1}^k \lambda_i = 1$, we have that*

$$\overline{\mu}\left(\sum_{i=1}^k \lambda_i z^{(i)}\right) = \sum_{i=1}^k \lambda_i \overline{\mu}(z^{(i)})\,.$$

A key property of the central path is *'near monotonicity'*, formulated in the following lemma, see [VY96, Lemma 16].

Lemma 5.2.3. *For the central path points at $0 \le \mu' \le \mu$, we have*

$$\left\| \frac{x^{\mathrm{cp}}(\mu')}{x^{\mathrm{cp}}(\mu)} + \frac{s^{\mathrm{cp}}(\mu')}{s^{\mathrm{cp}}(\mu)} \right\|_\infty \le n\,.$$

We will also require following lemma which was implicit in the proof of [VY96, Lemma 16].

Lemma 5.2.4. *For the central path points at $0 \le \mu' \le \mu$, we have*

$$\left\| \frac{x^{\mathrm{cp}}(\mu')}{x^{\mathrm{cp}}(\mu)} + \frac{s^{\mathrm{cp}}(\mu')}{s^{\mathrm{cp}}(\mu)} \right\|_1 \le 2n\,.$$

Proof. There is nothing to prove if $\mu = \mu' = 0$; let us assume $\mu > 0$, and denote $(x, s) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$, $(x', s') = (x^{\mathrm{cp}}(\mu'), s^{\mathrm{cp}}(\mu'))$. Using Proposition 5.2.1,

$$\begin{aligned} \|x'/x + s'/s\|_1 &= \mu^{-1}(\langle s, x'\rangle + \langle x, s'\rangle) = \mu^{-1}(\langle x, s\rangle + \langle x', s'\rangle) \\ &= \mu^{-1}(n\mu + n\mu' + 0) = n(1 + \mu^{-1}\mu') \le 2n. \end{aligned}$$

□

Central path neighborhoods. The neighborhoods $\mathcal{N}(\beta)$ and $\mathcal{N}^{-\infty}(\theta)$ introduced in (5.3) and (5.2) comprise the points $z = (x, s) \in \mathcal{P} \times \mathcal{D}$ such that the *centrality error*, i.e., the norm of the vector $\frac{xs}{\overline{\mu}(z)} - \mathbf{1}$, is bounded. They use of the ℓ_2-norm and the ℓ_∞-seminorm $\|u\|_{-\infty} := \max_{1 \le i \le n} \max(0, -u_i)$, respectively.

The following proposition gives a bound on the distance between a point $z \in \mathcal{N}(\beta)$ in the β-neighbourhood and the corresponding central path point with the same normalized duality gap $z(\mu)$ for $\mu = \overline{\mu}(z)$. See e.g. [Gon92, Lemma 5.4], [MT03, Proposition 2.1].

Proposition 5.2.5. *Let $z = (x, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$ and $\mu = \overline{\mu}(z)$, and consider the central path point $z^{\mathrm{cp}}(\mu) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$. For each $i \in [n]$,*

$$\begin{aligned} \frac{x_i}{1+2\beta} \le \frac{1-2\beta}{1-\beta}\, x_i \le x_i^{\mathrm{cp}}(\mu) \le \frac{x_i}{1-\beta}\,, \quad \text{and} \\ \frac{s_i}{1+2\beta} \le \frac{1-2\beta}{1-\beta}\, s_i \le s_i^{\mathrm{cp}}(\mu) \le \frac{s_i}{1-\beta}\,. \end{aligned}$$

We will often use the following proposition which is immediate from the definition of $\mathcal{N}(\beta)$.

Proposition 5.2.6. *Let $z = (x, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, and $\mu = \overline{\mu}(z)$. Then for each $i \in [n]$*

$$(1-\beta)\mu \le s_i x_i \le (1+\beta)\mu\,.$$

We will need the following lemma regarding the near-optimality of the choice $\overline{\mu}(z)$ as $\langle x, s\rangle/n$ for a point $z = (x, s)$ with respect to minimizing centrality error.

Lemma 5.2.7 ([MT05, Lemma 4.4]). *For $\beta \in (0, 1/4]$, let $z = (x, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$ and $\mu' > 0$ satisfy $\|xs/\mu' - \mathbf{1}\|_2 \le \beta$. Then,*

$$(1 - \beta/\sqrt{n})\mu' \le \overline{\mu}(z) \le (1 + \beta/\sqrt{n})\mu \quad \text{and} \quad z \in \mathcal{N}(\beta/(1-\beta)).$$

The next lemma relates a point in the wide neighborhood to the corresponding central path point.

Lemma 5.2.8. *Let $z = (x', s') \in \mathcal{N}^{-\infty}(\theta)$, $\theta \in [0, 1)$. Then for $\mu = \overline{\mu}(z)$, and the corresponding central path point $z^{\mathrm{cp}}(\mu) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$, we have that*

(1) $\frac{1}{2n}x' \le x^{\mathrm{cp}}(\mu) \le \frac{2n}{1-\theta}x'$.

(2) $\frac{1}{2n}s' \le s^{\mathrm{cp}}(\mu) \le \frac{2n}{1-\theta}s'$.

Proof. We only prove (1), as the proof of (2) is symmetric. Let $(x, s) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$. Using Proposition 5.2.1, for $i \in [n]$ we have that

$$\frac{x'_i}{x_i} = \frac{x'_i s_i}{\mu} \le \frac{1}{\mu}(\langle x', s\rangle + \langle x, s'\rangle) = \frac{1}{\mu}(\langle x', s'\rangle + \langle x, s\rangle) = 2n.$$

This proves the first inequality of (1); note that this part does not use $z \in \mathcal{N}^{-\infty}(\theta)$, but only that $z \in \mathcal{P}_{++} \times \mathcal{D}_{++}$. For the second inequality, $z \in \mathcal{N}^{-\infty}(\theta)$ by definition implies

$$\frac{x_i}{x'_i} \le \frac{x_i s'_i}{\mu(1-\theta)} \le \frac{1}{\mu(1-\theta)}(\langle x', s'\rangle + \langle x, s\rangle) = \frac{2n}{1-\theta},$$

as needed. □

5.2.2 Predictor-Corrector Methods

Given $z = (x, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$, the search directions commonly used in interior-point methods are obtained as the solution $(\Delta x, \Delta s)$ to the following linear system for some $\nu \in [0, 1]$.

$$\Delta x \in W \tag{5.10}$$

$$\Delta s \in W^{\perp} \tag{5.11}$$

$$s\Delta x + x\Delta s = \nu\mu\mathbf{1} - xs \tag{5.12}$$

Predictor-corrector methods, such as the Mizuno–Todd–Ye Predictor-Corrector (MTY P-C) algorithm [MTY93], alternate between two types of steps. In *predictor steps*, we use $\nu = 0$. This direction is also called the *affine scaling direction*, and will be denoted as $\Delta z^{\mathrm{a}} = (\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ throughout. In *corrector steps*, we use $\nu = 1$. This gives the *centrality direction*, denoted as $\Delta z^{\mathrm{c}} = (\Delta x^{\mathrm{c}}, \Delta s^{\mathrm{c}})$.

In the predictor steps, we make progress along the central path. Given the search direction on the current iterate $z = (x, s) \in \mathcal{N}(\beta)$, the step-length is chosen maximal such that we remain in $\mathcal{N}(2\beta)$, i.e.,

$$\alpha^{\mathrm{a}} := \sup\{\alpha \in [0, 1] : \forall \alpha' \in [0, \alpha] : z + \alpha'\Delta z^{\mathrm{a}} \in \mathcal{N}(2\beta)\}.$$

Thus, we obtain a point $z^+ = z + \alpha^{\mathrm{a}}\Delta w^{\mathrm{a}} \in \mathcal{N}(2\beta)$. The corrector step finds a next iterate $z^c = z^+ + \Delta z^{\mathrm{c}}$, where Δz^{c} is the centrality direction computed at z^+. The next proposition summarizes well-known properties, see e.g., [Ye97, Section 4.5.1].

Proposition 5.2.9. *Let $z = (x, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$.*

(i) For the affine scaling step, we have $\overline{\mu}(z^+) = (1 - \alpha^a)\overline{\mu}(z)$.

(ii) The affine scaling step-length is

$$\alpha^a \geq \max\left\{\frac{\beta}{\sqrt{n}}, 1 - \frac{\|\Delta x^a \Delta s^a\|}{\beta\overline{\mu}(z)}\right\}.$$

(iii) For $z^+ \in \mathcal{N}(2\beta)$, and $z^c = z^+ + \Delta w^c$, we have $\overline{\mu}(z^c) = \overline{\mu}(z^+)$ and $z^c \in \mathcal{N}(\beta)$.

(iv) After a sequence of $O(\sqrt{n}t)$ predictor and corrector steps, we obtain an iterate $z' = (x', s') \in \mathcal{N}(\beta)$ such that $\overline{\mu}(z') \leq \overline{\mu}(z)/2^t$.

Minimum-norm viewpoint We introduce some useful notation for the algorithm, and derive the minimum-norm interpretation of the affine scaling steps. For $z = (x, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$, we let

$$\xi(z) = \frac{x^{1/2}s^{1/2}}{\overline{\mu}(z)^{1/2}} \in \mathbb{R}^n\,, \quad \hat{x} = x\xi^{-1}(z) = x^{1/2}s^{-1/2}\overline{\mu}(z)^{1/2} \in \mathbb{R}^n\,, \quad \hat{s} = s\xi^{-1}(z) = s^{1/2}x^{-1/2}\overline{\mu}(z)^{1/2} \in \mathbb{R}^n\,. \tag{5.13}$$

If clear from the context, we simply use ξ. If $z = (x, s)$ falls on the central path, that is, $xs = \overline{\mu}(z)\mathbf{1}$, then $\xi(z) = \mathbf{1}$, $\hat{x} = x$ and $\hat{s} = s$. The variables $\hat{x}$ and $\hat{s}$ represent natural adjustments for points off the central path. The following is a simple corollary of Proposition 5.2.6.

Proposition 5.2.10. *For $z = (x, s) \in \mathcal{N}(\beta)$ for $\beta \in (0, 1/4]$, we have $\|\xi\| = \sqrt{n}$. Moreover,*

$$\sqrt{1-\beta}\mathbf{1} \leq \xi \leq \sqrt{1+\beta}\mathbf{1}\,, \quad \sqrt{1-\beta}x \leq \hat{x} \leq \sqrt{1+\beta}x \quad \text{and} \quad \sqrt{1-\beta}s \leq \hat{s} \leq \sqrt{1+\beta}s\,.$$

We will frequently use the rescaled subspaces $\hat{x}^{-1}W$ and $\hat{s}^{-1}W^\perp$ that correspond to using the local geometry at the point $z = (x, s)$. Throughout, we will refer to $\|w/\hat{x}\|$ and $\|w/\hat{s}\|$ as the *primal and dual local norms* of the vector $w \in \mathbb{R}^n$ at the point $z = (x, s) \in \mathcal{P}_{++} \times \mathcal{D}_{++}$. The following statement is immediate from the definitions using $\hat{x}\hat{s} = \overline{\mu}(z)\mathbf{1}$.

Proposition 5.2.11. *The subspaces $\hat{x}^{-1}W$ and $\hat{s}^{-1}W^\perp$ are orthogonal.*

Equation (5.12) for the predictor step ($\nu = 0$) with update direction $(\Delta x^a, \Delta s^a)$ can be written as

$$x^{-1}\Delta x^a + x^{-1}\Delta s^a = -\mathbf{1}\,, \tag{5.14}$$

or equivalently,

$$\hat{x}^{-1}\Delta x^a + \hat{s}^{-1}\Delta s^a = -\xi, \tag{5.15}$$

which serves the purpose that now $\hat{x}^{-1}\Delta x^a \in \hat{x}^{-1}W$ and $\hat{s}^{-1}\Delta s^a \in \hat{s}^{-1}W^\perp$ are orthogonal vectors (Proposition 5.2.11). Thus, $\hat{x}^{-1}\Delta x^a$ and $\hat{s}^{-1}\Delta s^a$ give an orthogonal decomposition of $-\xi$. This leads to the following formulas:

$$\begin{aligned} \Delta x^a &= -\hat{x}\Pi_{\hat{x}^{-1}W}(\xi)\,, \\ \Delta s^a &= -\hat{s}\Pi_{\hat{s}^{-1}W^\perp}(\xi)\,. \end{aligned} \tag{5.16}$$

Equivalently, we can see $\Delta z^a = (\Delta x^a, \Delta s^a)$ as the optimal solutions of the following minimum-norm problems:

$$\begin{aligned} \Delta x^a &= \arg\min_{\Delta x \in W} \|\hat{x}^{-1}(x + \Delta x)\|\,, \\ \Delta s^a &= \arg\min_{\Delta s \in W^\perp} \|\hat{s}^{-1}(s + \Delta s)\|\,. \end{aligned} \tag{5.17}$$

A further equivalent way to express these movement directions is by projections in the rescaled subspaces $\hat{x}^{-1}W$ and $\hat{s}^{-1}W^\perp$; this viewpoint will be used in Section 5.4.

$$\begin{aligned} \Delta x^{\mathrm{a}} &= \hat{x} \operatorname*{arg\,min}_{\delta \in \hat{x}^{-1}W} \|\xi + \delta\|, \\ \Delta s^{\mathrm{a}} &= \hat{s} \operatorname*{arg\,min}_{\delta \in \hat{s}^{-1}W^\perp} \|\xi + \delta\|. \end{aligned} \tag{5.18}$$

The equivalence of the two forms follows by noting that $\hat{x}^{-1}x = \hat{s}^{-1}s = \xi$.

Step-length estimates We will also need good estimates on the size on predictor steps beyond affine scaling. Our main estimate in this regard is given below.

Lemma 5.2.12 (Step-length estimate for general directions)**.** *Let $z = (x, s) \in \mathcal{N}(\beta)$, $\beta \in (0, 1/6]$. Consider directions $\Delta x \in W$, $\Delta s \in W^\perp$ that satisfy $\|\Delta x \Delta s\| \le \beta\mu/4$. Let*

$$\gamma = \frac{\|(x + \Delta x)(s + \Delta s)\|}{\mu}.$$

Then $(x+\alpha\Delta x, s+\alpha\Delta s) \in \mathcal{N}(2\beta)$ and $\overline{\mu}(x+\alpha\Delta x, s+\alpha\Delta s) \le (1+\frac{3}{2}\beta/\sqrt{n})(1-\alpha)\mu$, for all $0 \le \alpha \le 1-\frac{4\gamma}{\beta}$.

Proof. Let $z_\alpha := (x + \alpha\Delta x, s + \alpha\Delta s)$. We first bound the centrality error using the estimate $(1-\alpha)\mu$ for $\mu(z_\alpha)$ as follows:

$$\begin{aligned} \left\| \frac{(x + \alpha\Delta x)(s + \alpha\Delta s)}{(1-\alpha)\mu} - \mathbf{1} \right\| &= \left\| \frac{(1-\alpha)xs + \alpha(x + \Delta x)(s + \Delta s) - \alpha(1-\alpha)\Delta x \Delta s}{(1-\alpha)\mu} - \mathbf{1} \right\| \\ &\le \left\| \frac{xs}{\mu} - \mathbf{1} \right\| + \frac{\alpha}{1-\alpha} \left\| \frac{(x + \Delta x)(s + \Delta s)}{\mu} \right\| + \alpha \left\| \frac{\Delta x \Delta s}{\mu} \right\| \\ &\le \beta + \frac{\alpha}{1-\alpha}\gamma + \alpha\frac{\beta}{4} \le \frac{3}{2}\beta, \end{aligned} \tag{5.19}$$

where the last inequality follows since $\frac{\alpha}{1-\alpha}\gamma \le \beta/4$ for $0 \le \alpha \le 1 - 4\gamma/\beta$.

Given (5.19), by Lemma 5.2.7, we get that $\mu(z_\alpha) \le (1+\frac{3}{2}\beta/\sqrt{n})(1-\alpha)\mu$ and $z_\alpha \in \mathcal{N}\left(\frac{\frac{3}{2}\beta}{1-\frac{3}{2}\beta}\right) \subseteq \mathcal{N}(2\beta)$, for $\beta \in (0, 1/6]$. □

5.2.3 Lifting Maps

Our algorithm in Section 5.4 uses a layered least squares step that consists of solving a minimum-norm point in a smaller subspace first, and then extending it to the entire space. A crucial operation in both computing the layered step as well as in identifying the appropriate subspaces is the lifting map defined next.

Definition 5.2.13. *Given a partition $I \cup J = [n]$ and a subspace $W \subseteq \mathbb{R}^n$, we define the lifting map $L_I^W : \mathbb{R}^I \to W \subseteq \mathbb{R}^n$ as follows:*

$$L_I^W(x) := \arg\min\{ \|w\| : w \in W, w_I = \Pi_{\pi_I(W)}(x) \}. \tag{5.20}$$

We further define $\ell_I^W : \mathbb{R}^I \to \mathbb{R}^J$ by

$$\ell_I^W(x) := (L_I^W(x))_J = \pi_J\left(L_I^W(x)\right). \tag{5.21}$$

Note that if $x \in \pi_I(W)$, $w = L_I^W(x)$ is the minimum-norm point in W with $w_I = x$. The following lists the key properties of the lifting map.

Lemma 5.2.14. *For a partition $I \cup J = [n]$ and a linear subspace W, $L_I^W : \mathbb{R}^I \to W$ and $\ell_I^W : \mathbb{R}^I \to \mathbb{R}^J$ are linear maps. Moreover, for $x \in \mathbb{R}^I$, $w = L_I^W(x)$ is the unique solution to the following linear system:*

$$\begin{aligned} & w \in W\,, \\ & w_I \in \pi_I(W)^\perp + x\,, \\ & w_J \in \pi_J(W^\perp)\,. \end{aligned} \tag{5.22}$$

We now give the fundamental duality relation between lifting maps. For this purpose, we define $\ell_I^{W*} : \mathbb{R}^J \to \mathbb{R}^I$ to denote the adjoint of ℓ_I^W, namely, the map satisfying $\langle \ell_I^{W*}(y), x \rangle = \langle y, \ell_I(x) \rangle$, $\forall y \in \mathbb{R}^J, x \in \mathbb{R}^I$. If we are expressing ℓ_I^W as matrix $M \in \mathbb{R}^{J \times I}$, then ℓ_I^{W*} is represented by $M^\top$. That is, if $\ell_I^W(x) = Mx$, then $\ell_J^{W*}(y) = M^\top y$.

Lemma 5.2.15. *For a partition $I \cup J = [n]$ and a linear subspace W, we have that $\ell_I^W = \ell_J^{W^\perp *}$. In particular, the non-zero singular values of ℓ_I^W and $\ell_J^{W^\perp *}$ are the same.*

5.3 Polarization of the Central Path

We now introduce the notion of polarized segments of the central path. For $0 \le \mu_1 \le \mu_0$, the central path segment between these values is denoted by

$$\mathrm{CP}[\mu_1, \mu_0] \coloneqq \{\, z^{\mathrm{cp}}(\mu) : \mu_1 \le \mu \le \mu_0 \,\}\,. \tag{5.23}$$

Definition 5.3.1 (Polarization)**.** *For $\gamma \in [0,1]$ and $\mu_0 \ge \mu_1 \ge 0$, we say that the segment $\mathrm{CP}[\mu_1, \mu_0]$ is γ-polarized if there exists a partition $B \cup N = [n]$ such that for all $\mu \in [\mu_1, \mu_0]$:*

$$\begin{aligned} x_i^{\mathrm{cp}}(\mu) &\ge \gamma x_i^{\mathrm{cp}}(\mu_0)\,, \quad \forall i \in B\,, \\ s_i^{\mathrm{cp}}(\mu) &\ge \gamma s_i^{\mathrm{cp}}(\mu_0)\,, \quad \forall i \in N\,. \end{aligned}$$

Remark 5.3.2. As stated, the notion of polarization requires an inequality to hold for all $\mu \in [\mu_1, \mu_0]$. At the cost of losing a factor n however, it is in fact sufficient to check the polarization condition only at $\mu = \mu_1$. This follows by the near-monotonicity of the central path (Lemma 5.2.3):

$$\frac{x_i^{\mathrm{cp}}(\mu)}{x_i^{\mathrm{cp}}(\mu_0)} = \frac{x_i^{\mathrm{cp}}(\mu)}{x_i^{\mathrm{cp}}(\mu_1)} \cdot \frac{x_i^{\mathrm{cp}}(\mu_1)}{x_i^{\mathrm{cp}}(\mu_0)} \ge \frac{1}{n} \cdot \frac{x_i^{\mathrm{cp}}(\mu_1)}{x_i^{\mathrm{cp}}(\mu_0)}\,, \quad \forall i \in [n]\,,$$

The same is true for $s(\mu)$ by a symmetric argument.

As a direct consequence of the definition together with near-monotonicity, we deduce the following crucial corollary:

Corollary 5.3.3. *Let $\mathrm{CP}[\mu_1, \mu_0]$, $0 \le \mu_1 \le \mu_0$, be γ-polarized with respect to the partition $B \cup N = [n]$. Then, for all $\mu \in [\mu_1, \mu_0]$, the following holds:*

(1) $\gamma x_i(\mu_0) \le x_i(\mu) \le n x_i(\mu_0)$, $i \in B$.

(2) $\gamma s_i(\mu_0) \le s_i(\mu) \le n s_i(\mu_0)$, $i \in N$.

(3) $\frac{\mu}{n\mu_0} x_i(\mu_0) \le x_i(\mu) \le \frac{\mu}{\gamma\mu_0} x_i(\mu_0)$, $i \in N$.

(4) $\frac{\mu}{n\mu_0} s_i(\mu_0) \le s_i(\mu) \le \frac{\mu}{\gamma\mu_0} s_i(\mu_0)$, $i \in B$.

Proof. The first inequalities in (1) and (2) are the definition of γ-polarization and the second inequalities are from Lemma 5.2.3. (3) and (4) are equivalent to (1) and (2) using the central path relations $x(\mu_0)s(\mu_0) = \mu_0\mathbf{1}$ and $x(\mu)s(\mu) = \mu\mathbf{1}$. □

Section 5.4 introduces the algorithm IPM with subspace LLS that can traverse γ-polarized segments in $O(n^{1.5}\log(n/\gamma))$ iterations. Theorem 5.1.1 follows by combining this algorithm with the following decomposition result that is the main result of this section.

Theorem 5.3.4. *Let $\Gamma\colon (\mu_1, \mu_0) \to \mathcal{N}^{-\infty}(\theta)$, $\theta \in (0,1)$, $0 \le \mu_1 \le \mu_0 \le \infty$, be a piecewise linear curve satisfying $\overline{\mu}(\Gamma(\mu)) = \mu$, $\forall \mu \in (\mu_1, \mu_0)$ consisting of T linear segments. Then, $\mathrm{CP}[\mu_1, \mu_0]$ can be decomposed into T segments that are $\frac{(1-\theta)^2}{16n^3}$-polarized.*

Theorem 5.1.2 showing the combinatorial bound on the number of iterations of algorithm IPM with subspace LLS follows from the existence of a piecewise linear curve traversing the wide neighborhood with at most $O(2^n)$ pieces. This will be shown in Section 5.5, where we extract such a path from the max central path. We will also give a direct proof that the central path can be decomposed into $O(2^n)$ polarized segments, by showing that linear segments of the max-central path correspond to polarized segments of the central path. Theorem 5.3.4 is a direct consequence of the following key lemma.

Lemma 5.3.5. *For $\theta \in (0,1)$, let $[z^{(0)}, z^{(1)}] \subseteq \mathcal{N}^{-\infty}(\theta)$, $\overline{\mu}(z^{(0)}) > \overline{\mu}(z^{(1)})$. Then, $\mathrm{CP}[\overline{\mu}(z^{(1)}), \overline{\mu}(z^{(0)})]$ is $\frac{(1-\theta)^2}{16n^3}$-polarized.*

The proof requires the following simple technical lemma that allows us to relate approximate centrality along lines to polarization.

Lemma 5.3.6. *For any $u, v > 0$,*

$$\min_{\alpha\in[0,1]} \frac{(1-\alpha+\alpha u)(1-\alpha+\alpha v)}{1-\alpha+\alpha uv} = \min\left\{1, \left(\frac{\sqrt{u}+\sqrt{v}}{1+\sqrt{uv}}\right)^2\right\} \le 2(u+v)\,. \tag{5.24}$$

Proof. To show the equality, let $\mu := uv$. Note that

$$\begin{aligned}\min_{\alpha\in[0,1]} \frac{(1-\alpha+\alpha u)(1-\alpha+\alpha v)}{1-\alpha+\alpha uv} &= \min_{\alpha\in[0,1]} \frac{(1-\alpha)^2+\alpha^2\mu+\alpha(1-\alpha)(u+v)}{1-\alpha+\alpha\mu} \\ &= 1 + \min_{\alpha\in[0,1]} (u+v-(1+\mu))\frac{\alpha(1-\alpha)}{1-\alpha+\alpha\mu}\,.\end{aligned} \tag{5.25}$$

Case I: $\left(\frac{\sqrt{u}+\sqrt{v}}{1+\sqrt{uv}}\right)^2 \ge 1$**.** In this case, we need to show that the minimum of the expression is 1. It is easy to see that the condition equivalent to $u+v \ge 1+uv = 1+\mu$. Thus, the minimum value of (5.25) is 1, attained at $\alpha \in \{0,1\}$.

Case II: $\left(\frac{\sqrt{u}+\sqrt{v}}{1+\sqrt{uv}}\right)^2 < 1$**,** or equivalently, $u+v < 1+\mu$. In this case, the minimizer of (5.25) corresponds to the maximizer of $\frac{\alpha(1-\alpha)}{1-\alpha+\alpha\mu}$. This function takes value 0 at $\alpha \in \{0,1\}$ and is strictly positive for $0 < \alpha < 1$. Furthermore, the unique critical point in the interval $[0,1]$ occurs at $\alpha^* = \frac{1}{1+\sqrt{\mu}}$, which is thus the maximizer. The minimum value of (5.25) is therefore

$$1 + (u+v-(1+\mu))\frac{\alpha^*(1-\alpha^*)}{1-\alpha^*+\alpha^*\mu} = \frac{u+v+2\sqrt{\mu}}{(1+\sqrt{\mu})^2}\,,$$

as required. The inequality in the statement follows easily as

$$\left(\frac{\sqrt{u}+\sqrt{v}}{1+\sqrt{uv}}\right)^2 \le \left(\sqrt{u}+\sqrt{v}\right)^2 = 2(u+v) - (\sqrt{u}-\sqrt{v})^2 .$$

□

Proof of Lemma 5.3.5. For $\alpha \in [0,1]$, let $z^{(\alpha)} := (x^{(\alpha)}, s^{(\alpha)}) := (1-\alpha)z^{(0)} + \alpha z^{(1)}$. By Proposition 5.2.2, we first note that the normalized gap function $\overline{\mu}(z)$ is in fact linear on $[z^{(0)}, z^{(1)}]$. That is,

$$\mu_\alpha = \overline{\mu}(z^{(\alpha)}) = (1-\alpha)\overline{\mu}(z^{(0)}) + \alpha\overline{\mu}(z^{(1)}) .$$

For any $i \in [n]$, $z^{(\alpha)} \in \mathcal{N}^{-\infty}(\theta)$ implies

$$\frac{x_i^{(\alpha)} s_i^{(\alpha)}}{(1-\alpha)x_i^{(0)}s_i^{(0)} + \alpha x_i^{(1)} s_i^{(1)}} \ge \frac{(1-\theta)\mu_\alpha}{(1-\alpha)x_i^{(0)}s_i^{(0)} + \alpha x_i^{(1)} s_i^{(1)}} \ge \frac{(1-\theta)\mu_\alpha}{n((1-\alpha)\overline{\mu}(z^{(0)}) + \alpha\overline{\mu}(z^{(1)}))} \ge \frac{1-\theta}{n} .$$

From Lemma 5.3.6 for $u = x_i^{(1)}/x_i^{(0)}$, $v = s_i^{(1)}/s_i^{(0)}$ we get that

$$\frac{1-\theta}{n} \le 2\left(\frac{x_i^{(1)}}{x_i^{(0)}} + \frac{s_i^{(1)}}{s_i^{(0)}}\right).$$

Let

$$B := \left\{ i \in N : \frac{x_i^{(1)}}{x_i^{(0)}} \ge \frac{s_i^{(1)}}{s_i^{(0)}} \right\}, \quad N := [n] \setminus B .$$

Then, $x_i^{(1)}/x_i^{(0)} \ge \frac{1-\theta}{4n}$, for all $i \in B$, and $s_i^{(1)}/s_i^{(0)} \ge \frac{1-\theta}{4n}$ for all $i \in N$.

For any $\alpha \in [0,1]$ and $i \in B$,

$$\frac{x_i^{(\alpha)}}{x_i^{(0)}} = (1-\alpha) + \alpha\frac{x_i^{(1)}}{x_i^{(0)}} \ge \min\left\{1, \frac{1-\theta}{4n}\right\} = \frac{1-\theta}{4n}.$$

Similarly, for $i \in N$, $s_i^{(\alpha)}/s_i^{(0)} \ge \frac{1-\theta}{4n}$.

For the central path point $z^{\mathrm{cp}}(\mu_\alpha) = (x^{\mathrm{cp}}(\mu_\alpha), s^{\mathrm{cp}}(\mu_\alpha))$ at μ_α, the bounds in Lemma 5.2.8 relating points in a neighborhood with central path points give

$$\frac{x_i^{\mathrm{cp}}(\mu_\alpha)}{x_i^{\mathrm{cp}}(\mu_0)} \ge \frac{x_i^{(\alpha)}/(2n)}{\frac{2n}{1-\theta}x_i^{(0)}} \ge \frac{(1-\theta)^2}{16n^3}, \quad \forall i \in B .$$

By a similar argument, we also have $s_i^{\mathrm{cp}}(\mu_\alpha)/s_i^{\mathrm{cp}}(\mu_0) \ge \frac{(1-\theta)^2}{16n^3}$, $\forall i \in N$. Thus, $\mathrm{CP}[\mu_1, \mu_0]$ is $\frac{(1-\theta)^2}{16n^3}$ polarized. □

To conclude this section, we show that linearity of the central path can be equivalently restated in terms of polarization. While not needed in the sequel, we believe it to be of independent interest. The proof can be found in the Appendix.

Lemma 5.3.7. *For $\mu_0 > \mu_1 \ge 0$, let $\mathrm{CP}[\mu_1, \mu_0]$ be γ-polarized. Then $\gamma \in [0,1]$. Furthermore, $\gamma = 1$ if and only if $\mathrm{CP}[\mu_1, \mu_0]$ is linear.*

5.4 The Subspace Layered Least Squares IPM

In this section, we introduce the algorithm IPM with subspace LLS (Algorithm 5.1) and prove the following result.

Theorem 5.4.1 (Proof on 130). *Let CP$[\mu_1, \mu_0]$ be γ-polarized. Then, given an iterate $z \in \mathcal{N}(\beta)$ with parameter $\overline{\mu}(z) \in (\mu_1, \mu_0)$, the algorithm IPM with subspace LLS (Algorithm 5.1) takes $O\left(n^{1.5} \log(n/\gamma)\right)$ many iterations to find $z' \in \mathcal{N}(\beta)$ such that $\overline{\mu}(z') \leq \mu_1$.*

Recall from Theorem 5.3.4 that the existence of a line segment in the wide neighborhood of the central path implies polarization of the corresponding central path segment. Hence, Theorem 5.1.1 directly follows from Theorem 5.3.4 and Theorem 5.4.1. Recall from the discussion after the statement of Theorem 5.3.4 in Section 5.3 how Theorem 5.1.2 and Theorem 5.1.3 can also be derived. We stress that IPM with subspace LLS does not have any information about the polarization of CP$[\mu_1, \mu_0]$, but only defines the steps using local information.

The subspace LLS direction We now introduce a new update direction, called the *subspace layered least squares update direction*. At a given point $z \in \mathcal{N}(\beta)$, this step direction is specified by a partition $B \cup N = [n]$ and two subspaces $V \subseteq \pi_N(\hat{x}^{-1}W)$ and $U \subseteq \pi_B(\hat{s}^{-1}W^\perp)$.

Recall the notation ξ for the local error scaling and the thereby adjusted versions $\hat{x}$ and $\hat{s}$ of x and s defined in (5.13). For a given partition $B \cup N = [n]$ such that $B, N \neq \emptyset$, recall the lifting maps introduced in Section 5.2.3. We use the following shorthand notation

$$\ell_{z,N} := \ell_N^{\hat{x}^{-1}W} \quad \text{and} \quad \ell_{z,B}^\perp := \ell_B^{\hat{s}^{-1}W^\perp}.$$

Thus, the linear map $\ell_{z,N} \colon \mathbb{R}^N \to \mathbb{R}^B$ computes a minimum-norm lift in the rescaling $\hat{x}^{-1}W$ corresponding to the local geometry at the point z. Recall from Lemma 5.2.15 that the linear map $\ell_{z,B}^\perp \colon \mathbb{R}^B \to \mathbb{R}^N$ is the adjoint of $\ell_{z,N}$, thus, they are represented by matrices that are transposed of each other.

Also recall the affine scaling direction $(\Delta x^a, \Delta s^a)$; the most convenient formulation in the context of the next definition is (5.18).

Definition 5.4.2 (Subspace LLS direction). *Let $z \in \mathcal{N}(\beta)$, and let ξ, $\hat{x}$, $\hat{s}$ be defined as in (5.13). Assume we are given a partition $B \cup N = [n]$ and two subspaces $V \subseteq \pi_N(\hat{x}^{-1}W)$ and $U \subseteq \pi_B(\hat{s}^{-1}W^\perp)$. The* Subspace LLS *update direction $(\Delta x^\ell, \Delta s^\ell) \in W \times W^\perp$ at z with respect to (B, N, U, V) is defined as follows. If $B, N \neq \emptyset$, then*

$$\begin{aligned}
\delta^V &:= \arg\min_{\delta \in V} \|\xi_N + \delta\|, & \delta^U &:= \arg\min_{\delta \in U} \|\xi_B + \delta\|, \\
\delta^x &:= \left(\ell_{z,N}(\delta^V), \delta^V\right), & \delta^s &:= \left(\delta^U, \ell_{z,B}^\perp(\delta^U)\right), \\
\Delta x^\ell &:= \hat{x}\delta^x, & \Delta s^\ell &:= \hat{s}\delta^s.
\end{aligned}$$

If $N = \emptyset$, then we let $(\Delta x^\ell, \Delta s^\ell) = (0, \Delta s^a)$ and if $B = \emptyset$, then we let $(\Delta x^\ell, \Delta s^\ell) = (\Delta x^a, 0)$.

The formula defining δ^V is similar to the definition (5.18) of the affine scaling direction. However, when defining δ^V, we restrict ourselves to norm minimization in $\pi_N(\hat{x}^{-1}W)$, and within that, we require $\delta^V \in V$. This step is then extended to the coordinates in B using a minimum norm lift in the rescaled subspace $\hat{x}^{-1}W$.

Note that we can equivalently write $\delta^V = -\Pi_V(\xi_N)$ and $\delta^U = -\Pi_U(\xi_B)$. An equivalent definition of Δx^ℓ, using the primal local norm in the original space W—similarly to (5.17)—is the following:

$$\begin{aligned} \Delta v^\ell &:= \arg\min\{ \left\|\hat{x}_N^{-1}(x_N + \Delta v)\right\| : \Delta v \in \hat{x}V \}\,, \\ \Delta x^\ell &:= \arg\min\{ \left\|\hat{x}^{-1}\Delta x\right\| : \Delta x_N = \Delta v^\ell\,, \Delta x \in W \}\,. \end{aligned} \tag{5.26}$$

Analogous formulas can be given for Δs^ℓ.

For the subspace LLS direction as in Definition 5.4.2, let us define the residuals

$$\varrho^V := \xi + \delta^x = \frac{x + \Delta x^\ell}{\hat{x}}\,, \quad \varrho^U := \xi + \delta^s = \frac{s + \Delta s^\ell}{\hat{s}}\,. \tag{5.27}$$

Cheap lifts and singular values To argue for the usefulness of the above defined step direction, and to select suitable subspaces V and U for a given partition (B, N), we recall the discussion of the trust region step from the Introduction (Section 5.1). As long as the step primal and dual directions $(\Delta x, \Delta s)$ are feasible to the systems (5.6) for a suitably small threshold ϱ, we are guaranteed to make progress as measured by the primal and dual objective values as in (5.7).

Simply selecting $V = \pi_N(\hat{x}^{-1}W)$ and $U = \pi_B(\hat{s}^{-1}W^\perp)$ would attain the smallest possible objective values; however, the constraints bounding the local norms of Δx_B^ℓ and Δs_N^ℓ in (5.6) could be arbitrarily violated. Therefore, we select the subspaces V and U such that no matter how δ^V and δ^U are selected, the lifts $\Delta x_B^\ell = \hat{x}_N \ell_{z,N}(\delta^V)$ and $\Delta s_N^\ell = \hat{s}_N \ell_{z,B}^\perp(\delta^U)$ will have small local norms; that is, $\|\ell_{z,N}(\delta^V)\|$ and $\|\ell_{z,B}^\perp(\delta^U)\|$ are suitably bounded.

Using that $\|\delta^V\| \le \|\xi_N\|$ as δ^V is the projection of ξ_N, and Proposition 5.2.10, we get

$$\|\delta^V\| \le \|\xi_N\| \le \sqrt{n}\,.$$

Hence, if we can guarantee that the lifting map satisfies $\|\ell_{z,N}(\delta)\| \le \tau\|\delta\|$ for all $\delta \in V$ for a suitably small threshold τ, we can guarantee Δx^ℓ to be feasible to the primal trust region program (5.6); analogous arguments can be made for the dual direction. Naturally, we would like to select the largest subspaces V and U with this property.

Let us define the threshold

$$\tau := \frac{\beta}{16\sqrt{n}}\,. \tag{5.28}$$

Our goal is to ensure that

$$\|\ell_{z,N}(\delta)\| \le \tau\|\delta\|\,, \quad \forall \delta \in V \quad \text{and} \quad \|\ell_{z,B}^\perp(\delta)\| \le \tau\|\delta\|\,, \quad \forall \delta \in U\,. \tag{5.29}$$

This holds if V is the subspace spanned by the singular vectors of the map $\ell_{z,N}$ corresponding to the singular values $\le \tau$ in absolute value; analogously for $\ell_{z,B}^\perp$ and U. In the following definition, we use a full Singular Value Decomposition (SVD) of the matrix $\mathbf{M} \in \mathbb{R}^{|B|\times|N|}$ representing the map $\ell_{z,N}$. The Singular Value Decomposition (SVD) gives $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ where $\mathbf{U} \in \mathbb{R}^{|B|\times|B|}$ and $\mathbf{V} \in \mathbb{R}^{|N|\times|N|}$ are orthogonal matrices, and $\mathbf{\Sigma} \in \mathbb{R}^{|B|\times|N|}$ is a rectangular diagonal matrix also including the zero singular values.

Definition 5.4.3 (Cheap Subspaces). *For a partition $B \cup N = [n]$ with $B, N \neq \emptyset$ and $z \in \mathcal{N}(\beta)$, consider an SVD decomposition*

$$\ell_{z,N} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top\,, \quad \ell_{z,B}^\perp = \mathbf{V}\mathbf{\Sigma}^\top\mathbf{U}^\top\,.$$

Depending on a threshold parameter $t \in \mathbb{R}_{>0}$, we set

$$V_{z,N}(t) := \mathrm{im}(\mathbf{V}_S) \cap \pi_N(\hat{x}^{-1}W) \quad \text{for} \quad S := \{\, i : \|\mathbf{\Sigma}_{\cdot,i}\| \le t \,\},$$
$$U_{z,B}(t) := \mathrm{im}(\mathbf{U}_T) \cap \pi_B(\hat{s}^{-1}W^\perp) \quad \text{for} \quad T := \{\, i : \|\mathbf{\Sigma}_{i,\cdot}\| \le t \,\}.$$

Further, we let $\overline{V}_{z,N}(t)$ be the orthogonal complement of $V_{z,N}(t)$ in $\pi_N(\hat{x}^{-1}W)$, that is, $V_{z,N}(t) \oplus \overline{V}_{z,N}(t) = \pi_N(\hat{x}^{-1}W)$ and $V_{z,N}(t) \perp \overline{V}_{z,N}(t)$. Analogously we define $\overline{U}_{z,B}(t)$ such that $U_{z,B}(t) \oplus \overline{U}_{z,B}(t) = \pi_B(\hat{s}^{-1}W^\perp)$ and $U_{z,B}(t) \perp \overline{U}_{z,B}(t)$.

Note that $\|\mathbf{\Sigma}_{\cdot,i}\|$ and $\|\mathbf{\Sigma}_{\cdot,i}\|$ are the absolute values of the singular values corresponding to the i-th columns of $\mathbf{V}$ and $\mathbf{U}$, respectively. It is clear from the definition that $V = V_{z,N}$ and $U = U_{z,B}$ satisfy (5.29).

Definition 5.4.3 gives a natural way of defining $V = V_{z,N}(\tau)$ and $U = U_{z,B}(\tau)$ for our algorithm. However, this requires an exact SVD decomposition of the matrix $\mathbf{M}$ representing the lifting map $\ell_{z,N}$. It is important to note that our arithmetic model does not allow for computing an exact decomposition. Since we would like to implement each iteration of the algorithm in strongly polynomial time, we cannot use a numerical approximation for the SVD , as such approximations would depend on the norm of the matrix. However, a weaker property suffices for our analysis. Namely, we need that the subspace $V \subseteq \pi_{\widetilde{N}}(\hat{x}^{-1}W)$ has singular values at most τ—and thus satisfies (5.29)—and further it includes the subspace of right singular vectors corresponding to the singular values $\le \tau/n^c$ for some constant $c > 0$; analogously for U. A strongly polynomial subroutine Approx-SVD finding such a subspace is given in Section 5.4.1.

The associated partition Definition 5.4.2 and Definition 5.4.3 are applicable for any partition $B \cup N = [n]$, $B, N \neq \emptyset$ and $z \in \mathcal{N}(\beta)$. Our algorithm chooses a natural partition derived from the relative step lengths in the affine scaling step:

Definition 5.4.4 (associated partition)**.** *For $z = (x, s) \in \mathcal{N}(\beta)$, let $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ be the affine scaling step as in (5.16). Let us define the associated partition $\widetilde{B}_z \cup \widetilde{N}_z = [n]$ as*

$$\widetilde{B}_z := \left\{ i : \left|\frac{\Delta x^{\mathrm{a}}_i}{x_i}\right| < \left|\frac{\Delta s^{\mathrm{a}}_i}{s_i}\right| \right\} \quad \widetilde{N}_z := [n] \setminus \widetilde{B}_z\,.$$

Further, let $\widetilde{\ell}_z := \ell_{z,\widetilde{N}_z}$ and $\widetilde{\ell}^\perp_z := \ell^\perp_{z,\widetilde{B}_z}$ denote the lifting maps corresponding to this partition, assuming $\widetilde{B}_z, \widetilde{N}_z \neq \emptyset$.

The affine scaling step is the canonical candidate for an improving direction. Namely, for each $i \in \widetilde{B}_z$ the variable s_i decreases at a faster rate than x_i, and vice versa for $i \in \widetilde{N}_z$. As shown in the analysis (Lemma 5.4.11), for a sufficiently long polarized segment, $(\widetilde{B}_z, \widetilde{N}_z)$ reveals the polarizing partition.

Description of the algorithm We are ready to describe the predictor-corrector algorithm IPM with subspace LLS, shown in Algorithm 5.1. We are given a starting point $(x^0, s^0) \in \mathcal{N}(\beta)$. In each iteration, we compute the affine scaling direction $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ and identify the associated partition $(\widetilde{B}, \widetilde{N})$. Using this partition, we approximate the cheap subspaces U and V using Approx-SVD (Algorithm 5.2). We then compute the subspace LLS direction $(\Delta x^\ell, \Delta s^\ell)$ for $(\widetilde{B}, \widetilde{N}, V, U)$. For both directions, compute the feasible step-lengths according to the bounds in Proposition 5.2.9 and Lemma 5.2.12. We use the better of these two possible steps, and obtain the next iterate after a corrector step.

Algorithm 5.1: IPM with subspace LLS

Input : Instance of LP and initial $(x^0, s^0) \in \mathcal{N}(\beta)$, $\beta \in (0, 1/6]$.
Output : Optimal solution $(x^\star, s^\star)$ to LP.

$x \leftarrow x^0, s \leftarrow s^0$
while $\langle x, s\rangle > 0$ **do**
  Compute affine scaling direction $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$
  Set α^{a} for $(\Delta x^{\mathrm{a}}, \Delta s^{\mathrm{a}})$ according to Proposition 5.2.9(ii)
  $(\hat{x}^{\mathrm{a}}, \hat{s}^{\mathrm{a}}) \leftarrow (x + \alpha^{\mathrm{a}}\Delta x^{\mathrm{a}}, s + \alpha^{\mathrm{a}}\Delta s^{\mathrm{a}})$
  $\widetilde{B} \leftarrow \left\{ i : \left|\frac{\Delta x_i^{\mathrm{a}}}{x_i}\right| < \left|\frac{\Delta s_i^{\mathrm{a}}}{s_i}\right| \right\}, \widetilde{N} \leftarrow [n] \setminus \widetilde{B}$
  $\mathbf{V} \leftarrow$ Approx-SVD$(\ell_{(x,s),\widetilde{N}})$
  $i \leftarrow \max\{j : \frac{\|\ell_{(x,s),\widetilde{N}}(\mathbf{V}_j)\|}{\|\mathbf{V}_j\|} \leq \frac{\tau}{n}\}$
  $V \leftarrow \mathrm{im}(\mathbf{V}_{\leq i}) \cap \pi_{\widetilde{N}}(\hat{x}^{-1}W)$
  $\mathbf{U} \leftarrow$ Approx-SVD$(\ell^{\perp}_{(x,s),\widetilde{B}})$
  $i \leftarrow \max\left\{ j : \frac{\|\ell^{\perp}_{(x,s),\widetilde{B}}(\mathbf{U}_j)\|}{\|\mathbf{U}_j\|} \leq \frac{\tau}{n} \right\}$
  $U \leftarrow \mathrm{im}(\mathbf{U}_{\leq i}) \cap \pi_{\widetilde{B}}(\hat{s}^{-1}W^{\perp})$
  Find subspace LLS direction $(\Delta x^\ell, \Delta s^\ell)$ according to Definition 5.4.2 for V and U
  Set α^ℓ for $(\Delta x^\ell, \Delta s^\ell)$ according to Lemma 5.2.12
  $(\hat{x}^\ell, \hat{s}^\ell) \leftarrow (x + \alpha^\ell \Delta x^\ell, s + \alpha^\ell \Delta s^\ell)$
  if $\overline{\mu}(\hat{x}^{\mathrm{a}}, \hat{s}^{\mathrm{a}}) \leq \overline{\mu}(\hat{x}^\ell, \hat{s}^\ell)$ **then**
    $(x, s) \leftarrow (\hat{x}^{\mathrm{a}}, \hat{s}^{\mathrm{a}})$
  else
    $(x, s) \leftarrow (\hat{x}^\ell, \hat{s}^\ell)$
  $(\Delta x^c, \Delta s^c) =$ Corrector(x, s)
  $x \leftarrow x + \Delta x^c, s \leftarrow s + \Delta s^c$
return (x, s)

The main potential in the analysis is the $\sigma(z)$, the non-decreasing vector of non-zero singular values of ℓ_z and $\ell_z^\perp$, which are the same by Lemma 5.2.15.

In particular, we have

$$\sigma(z)_i = \sigma_{i+\dim(\ker(\ell_z))}(\ell_z) = \sigma_{i+\dim(\ker(\ell_z^\perp))}(\ell_z^\perp)\,. \tag{5.30}$$

for all $i \in [\dim(\mathrm{im}(\ell_z))]$.

Throughout this section, we will analyze the behaviour of IPM with subspace LLS on a fixed γ-polarized segment CP$[\mu_1, \mu_0]$ with respect to the polarizing partition $B \cup N = [n]$. This partition will be fixed throughout, and for $z \in \mathcal{N}(\beta)$ with $\overline{\mu}(z) \in [\mu_1, \mu_0]$, we use the shorthands

$$\ell_z := \ell_{z,N} = \ell_N^{\hat{x}^{-1}W}\,, \quad \ell_z^\perp := \ell_{z,B}^\perp = \ell_B^{\hat{s}^{-1}W^\perp}\,.$$

5.4.1 Algorithmic Tools

Polynomial approximation of singular values

To compute the subspace LLS direction, we need to identify the linear spaces U and V that are obtained from an exact SVD of the matrix M representing the lifting map $\widetilde{\ell_z}$. It is important to note that our arithmetic model does not allow for computing an exact decomposition. Since our goal is to be able to implement each iteration of the algorithm in strongly polynomial time, we cannot use a numerical approximation for the SVD, as such approximations would depend on the norm of the

matrix. However, the analysis of the main algorithm (Algorithm 5.1) is robust, and it suffices to identify a subspace $V \subseteq \pi_{\tilde{N}}(\hat{x}^{-1}W)$ such that the lifting map from V has singular values at most τ and includes the subspace of right singular vectors corresponding to the singular values $\leq \tau/n^c$ for some threshold τ and $c > 0$. In the following we provide such an approximation for $c = 1/2$.

Algorithm 5.2: Approx-SVD

Input : A matrix $\mathbf{M} \in \mathbb{R}^{m\times n}$.
Output : A matrix $\mathbf{V} \in \mathbb{R}^{n\times n}$.

1 $\mathbf{V} \leftarrow \mathbf{I}_n$
2 **for** $i = 1, \ldots, n$ **do**
3 $\quad \mathbf{V} \leftarrow$ ORTHOGONALIZE$(\mathbf{I}_n, \mathbf{V})$
4 $\quad \mathbf{V}_{\geq i} \leftarrow$ ORTHOGONALIZE$(\mathbf{M}, \mathbf{V}_{\geq i})$
5 $\quad c \leftarrow \arg\min_{i\leq j\leq n} \frac{\|\mathbf{M}\mathbf{V}_j\|}{\|\mathbf{V}_j\|}$
6 $\quad \mathbf{V} \leftarrow \mathbf{V}$ with column i swapped with column c
7 $\quad \mathbf{V}^{(i)} \leftarrow \mathbf{V}$ // Only needed for the analysis
8 **return** $\mathbf{V}$
9 **procedure** *Orthogonalize*$(\mathbf{N}, \mathbf{V})$:
$\quad$ **Input** : Matrices $\mathbf{N} \in \mathbb{R}^{m\times n}$ and $\mathbf{V} \in \mathbb{R}^{n\times k}$
$\quad$ **Output** : A matrix $\widehat{\mathbf{V}} \in \mathbb{R}^{n\times k}$ with $\mathrm{im}(\mathbf{V}_{\leq i}) = \mathrm{im}(\widehat{\mathbf{V}}_{\leq i})$ for all $i \in [k]$ and $\mathbf{N}$-orthogonal columns, i.e., $\widehat{\mathbf{V}}^\top\mathbf{N}^\top\mathbf{N}\widehat{\mathbf{V}}$ is diagonal. // If $m = 0$ return V.
10 $\quad$ Perform standard Gram-Schmidt orthogonalization with the inner product induced by $\mathbf{N}$

We denote the set of i-dimensional linear subspaces of $\mathbb{R}^n$ by $\mathcal{S}(i)$. Recall the min-max principle for singular values:

$$\sigma_i(\mathbf{M}) = \min_{S\in\mathcal{S}(i)} \max_{u\in S\setminus\{0\}} \frac{\|\mathbf{M}u\|}{\|u\|} = \max_{S\in\mathcal{S}(n-i+1)} \min_{u\in S\setminus\{0\}} \frac{\|\mathbf{M}u\|}{\|u\|}. \tag{5.31}$$

The next statement gives an approximate version based on an the approximate representatives for i-dimensional linear subspaces of $\mathbb{R}^n$.

Lemma 5.4.5. *Algorithm 5.2 returns a matrix* $\mathbf{V}$ *with orthogonal columns such that*

$$\frac{\sigma_i(\mathbf{M})}{n} \leq \frac{1}{n} \max_{v\in\mathrm{im}(\mathbf{V}_{\leq i})\setminus\{0\}} \frac{\|\mathbf{M}v\|}{\|v\|} \leq \frac{\|\mathbf{M}\mathbf{V}_i\|}{\|\mathbf{V}_i\|} \leq \sqrt{n} \min_{v\in\mathrm{im}(\mathbf{V}_{\geq i})\setminus\{0\}} \frac{\|\mathbf{M}v\|}{\|v\|} \leq \sqrt{n}\sigma_i(\mathbf{M}), \tag{5.32}$$

for all $i \in [n]$.

Proof. Note that the first and last inequality follow from (5.31). We prove the second and the third inequality. Note that by choice of j in line 5 we have for all j with $i \leq j \leq n$ that

$$\frac{\|\mathbf{M}\mathbf{V}_i^{(i)}\|}{\|\mathbf{V}_i^{(i)}\|} \leq \frac{\|\mathbf{M}\mathbf{V}_j^{(i)}\|}{\|\mathbf{V}_j^{(i)}\|}. \tag{5.33}$$

Therefore, by $\mathbf{M}$-orthogonality of $\mathbf{V}_{\geq i}^{(i)}$ we have

$$\begin{aligned}\frac{\|\mathbf{M}\mathbf{V}_{\geq i}^{(i)}u\|}{\|\mathbf{V}_{\geq i}^{(i)}u\|} = \frac{\sqrt{\sum_{j\geq i} u_j^2\|\mathbf{M}\mathbf{V}_j^{(i)}\|^2}}{\|\mathbf{V}_{\geq i}^{(i)}u\|} &\overset{(5.33)}{\geq} \frac{\|\mathbf{M}\mathbf{V}_i^{(i)}\|}{\|\mathbf{V}_i^{(i)}\|}\frac{\sqrt{\sum_{j\geq i} u_j^2\|\mathbf{V}_j^{(i)}\|^2}}{\|\mathbf{V}_{\geq i}^{(i)}u\|} \geq \frac{\|\mathbf{M}\mathbf{V}_i^{(i)}\|}{\|\mathbf{V}_i^{(i)}\|}\frac{\sqrt{\sum_{j\geq i} u_j^2\|\mathbf{V}_j^{(i)}\|^2}}{\left(\sum_{j\geq i} u_j\|\mathbf{V}_j^{(i)}\|\right)} \\ &\geq \frac{1}{\sqrt{n}}\frac{\|\mathbf{M}\mathbf{V}_i^{(i)}\|}{\|\mathbf{V}_i^{(i)}\|}.\end{aligned} \tag{5.34}$$

Note that $\mathrm{im}(\mathbf{V}_{\geq i}^{(i)}) = \mathrm{im}(\mathbf{V}_{\geq i})$ for and $\mathbf{V}_i^{(i)} = \mathbf{V}_i$ for all $i \in [n]$.
Therefore,

$$\min_{v\in\mathrm{im}(\mathbf{V}_{\geq i})}\frac{\|\mathbf{M}v\|}{\|v\|}=\min_{v\in\mathrm{im}(\mathbf{V}^{(i)}_{\geq i})}\frac{\|\mathbf{M}v\|}{\|v\|}\overset{(5.34)}{\geq}\frac{1}{\sqrt{n}}\frac{\|\mathbf{M}\mathbf{V}^{(i)}_i\|}{\|\mathbf{V}^{(i)}_i\|}=\frac{1}{\sqrt{n}}\frac{\|\mathbf{M}\mathbf{V}_i\|}{\|\mathbf{V}_i\|}\,, \tag{5.35}$$

which proves the third inequality of the lemma.

It remains to prove the second inequality of the lemma. Note that by the $\mathbf{I}_n$-orthogonality of the columns of $\mathbf{V}$ we have for all $i\in[n]$ that

$$\begin{aligned}\max_{v\in\mathrm{im}(\mathbf{V}_{\leq i})\setminus\{0\}}\frac{\|\mathbf{M}v\|}{\|v\|}&=\max_{u\in\mathbb{R}^i\setminus\{0\}}\frac{\|\mathbf{M}\mathbf{V}_{\leq i}u\|}{\|\mathbf{V}_{\leq i}u\|}\overset{\text{orthogonality}}{=}\max_{u\in\mathbb{R}^i\setminus\{0\}}\frac{\|\mathbf{M}\mathbf{V}_{\leq i}u\|}{\sqrt{\sum_{j\leq i}\|\mathbf{V}_j\|^2u_j^2}}\\&\leq\max_{u\in\mathbb{R}^i\setminus\{0\}}\frac{\sum_{j\leq i}\|\mathbf{M}\mathbf{V}_j\|\|u_j|}{\sqrt{\sum_{j\leq i}\|\mathbf{V}_j\|^2u_j^2}}\overset{\text{AM-QM}}{\leq}\max_{u\in\mathbb{R}^i\setminus\{0\}}\sqrt{n}\frac{\sum_{j\leq i}\|\mathbf{M}\mathbf{V}_j\|\|u_j|}{\sum_{j\leq i}\|\mathbf{V}_j\|\|u_j|}\\&\overset{\text{mediant}}{\leq}\sqrt{n}\max_{j\leq i}\frac{\|\mathbf{M}\mathbf{V}_j\|}{\|\mathbf{V}_j\|}\,.\end{aligned} \tag{5.36}$$

Further, we have for all $j\leq i$ that $\mathbf{V}_i\in\mathrm{im}(\mathbf{V}^{(j)}_{\geq j})=\mathrm{im}(\mathbf{V}_{\geq j})$. Therefore,

$$\frac{\|\mathbf{M}\mathbf{V}_i\|}{\|\mathbf{V}_i\|}\geq\min_{v\in\mathrm{im}(\mathbf{V}_{\geq j})}\frac{\|\mathbf{M}v\|}{\|v\|}\overset{(5.35)}{\geq}\frac{1}{\sqrt{n}}\frac{\|\mathbf{M}\mathbf{V}_j\|}{\|\mathbf{V}_j\|}\,. \tag{5.37}$$

Combining (5.37) with (5.36) gives the second inequality of the lemma. □

Stability of singular values on polarized segments

We now present two statements that describe the stability and evolution of singular values of the map ℓ_z on polarized segments of the central path.

Lemma 5.4.6 (Stability of singular values for multiplicative perturbation)**.** *Let $y\in\mathbb{R}^n_{>0}$, $W\subseteq\mathbb{R}^n$ a subspace and let $\ell_N^{y^{-1}W}\colon\mathbb{R}^N\to\pi_B(y^{-1}W)$ and $\ell_N^W\colon\mathbb{R}^N\to\pi_B(W)$ be defined according to Definition 5.2.13. Let $\sigma_1\leq\cdots\leq\sigma_N$ and $\widetilde{\sigma}_1\leq\cdots\leq\widetilde{\sigma}_N$ their respective singular values. Then, for all $k\in[N]$, we have*

$$\frac{1}{\|y_B^{-1}\|_\infty\|y_N\|_\infty}\sigma_k\leq\widetilde{\sigma}_k\leq\|y_B\|_\infty\|y_N^{-1}\|_\infty\sigma_k\,. \tag{5.38}$$

Proof. We denote $\ell:=\ell_N^{y^{-1}W}$ and $\widetilde{\ell}:=\ell_N^W$.

We prove the second inequality, the first inequality follows by replacing y with y^{-1} and σ with $\tilde{\sigma}$.

Note that for any $x\in\pi_N(y^{-1}W)$, we have $y_Nx=(yL_N^{y^{-1}W}(x))_N\in\pi_N(W)$ and $(yL_N^{y^{-1}W}(x))_B=y_B\ell(x)\in\pi_B(W)$. By definition of $\widetilde{\ell}$, we deduce that

$$\|\widetilde{\ell}(y_Nx)\|\leq\|y_B\ell(x)\|\leq\|y_B\|_\infty\|\ell(x)\|\,. \tag{5.39}$$

Consider the singular value decomposition $\ell=\mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top$ such that the diagonal of $\mathbf{\Sigma}$ is non-decreasing. For $k\in|N|$, let $V^{(k)}=\mathrm{im}((\mathbf{V}^\top)_{\leq k})\subseteq\pi_N(y^{-1}W)$ be the column span of the first k columns of $\mathbf{V}^\top$. This is the subspace corresponding to the smallest k singular values of ℓ. Let further $\widetilde{V}^{(k)}:=y_NV^{(k)}$ be the rescaled subspace in $\pi_N(W)$. Now, for any $\widetilde{v}\in\widetilde{V}^{(k)}$, we have for $v:=y_N^{-1}\widetilde{v}\in V^{(k)}$ that $\|\ell(v)\|\leq\sigma_k\|v\|$ and so

$$\|\widetilde{\ell}(\widetilde{v})\|=\|\widetilde{\ell}(y_Nv)\|\overset{(5.39)}{\leq}\|y_B\|_\infty\|\ell(v)\|\leq\|y_B\|_\infty\sigma_k\|v\|\leq\sigma_k\|y_B\|_\infty\|y_N^{-1}\|_\infty\|\widetilde{v}\|\,. \tag{5.40}$$

Hence, by (5.31) we get

$$\widetilde{\sigma}_k \le \max_{\widetilde{v}\in\widetilde{V}^{(k)}} \frac{\|\widetilde{\ell}(\widetilde{v})\|}{\|\widetilde{v}\|} \overset{(5.40)}{\le} \|y_B\|_\infty \|y_N^{-1}\|_\infty \sigma_k \,. \qquad \square$$

Lemma 5.4.7 (Stability of singular values on polarized segments). *Let* $\mathrm{CP}[\mu_1,\mu_0]$ *be a* γ*-polarized segment of the central path with partition* $B\cup N=[n]$*. Let* $z,\widetilde{z}\in\mathcal{N}(\beta)$ *for* $\beta\in(0,1/6)$*, such that* $\mu := \overline{\mu}(z)$ *and* $\widetilde{\mu} := \overline{\mu}(\widetilde{z})$ *satisfy* $\mu_0\ge\mu\ge\widetilde{\mu}\ge\mu_1$*. Then we have:*

$$\frac{\gamma^2}{4n^2}\cdot\frac{\widetilde{\mu}}{\mu}\sigma(z) \le \sigma(\widetilde{z}) \le \frac{4n^2}{\gamma^2}\cdot\frac{\mu}{\widetilde{\mu}}\sigma(z). \tag{5.41}$$

Proof. Let $z := (x,s)$ and $\widetilde{z} := (\widetilde{x},\widetilde{s})$. We denote $\xi(z) = x^{1/2}s^{1/2}\mu^{-1/2}$ and $\xi(\widetilde{z}) = \widetilde{x}^{1/2}\widetilde{s}^{1/2}\widetilde{\mu}^{-1/2}$ by ξ and $\widetilde{\xi}$, respectively. We want to apply Lemma 5.4.6 with $y = x\xi^{-1}\widetilde{x}^{-1}\widetilde{\xi}$. By Proposition 5.2.10 we have

$$\sqrt{1-\beta}\mathbf{1} \le \xi,\widetilde{\xi} \le \sqrt{1+\beta}\mathbf{1}\ . \tag{5.42}$$

With Proposition 5.2.5 and Corollary 5.3.3 we get

$$\begin{aligned}
\|y_B\|_\infty = \|x_B\widetilde{x}_B^{-1}\xi_B^{-1}\widetilde{\xi}_B\|_\infty &\le \frac{(1+2\beta)(1+\beta)^{1/2}}{(1-\beta)^{3/2}}\left\|\frac{x(\mu)_B}{x(\widetilde{\mu})_B}\right\|_\infty \le \frac{2n}{\gamma}\\
\|y_N^{-1}\|_\infty = \|\widetilde{x}_N x_N^{-1}\widetilde{\xi}_N^{-1}\xi_N\|_\infty &\le \frac{(1+2\beta)(1+\beta)^{1/2}}{(1-\beta)^{3/2}}\left\|\frac{x(\widetilde{\mu})_N}{x(\mu)_N}\right\|_\infty \le \frac{2n\mu}{\gamma\widetilde{\mu}}
\end{aligned} \tag{5.43}$$

and

$$\begin{aligned}
\|y_B^{-1}\|_\infty = \|x_B^{-1}\widetilde{x}_B\xi_B\widetilde{\xi}_B^{-1}\|_\infty &\le \frac{(1+2\beta)(1+\beta)^{1/2}}{(1-\beta)^{3/2}}\left\|\frac{x(\widetilde{\mu})_B}{x(\mu)_B}\right\|_\infty \le \frac{2n}{\gamma}\\
\|y_N\|_\infty = \|\widetilde{x}_N^{-1}x_N\widetilde{\xi}_N\xi_N^{-1}\|_\infty &\le \frac{(1+2\beta)(1+\beta)^{1/2}}{(1-\beta)^{3/2}}\left\|\frac{x(\mu)_N}{x(\widetilde{\mu})_N}\right\|_\infty \le \frac{2n\mu}{\gamma\widetilde{\mu}}\ .
\end{aligned} \tag{5.44}$$

using $\frac{(1+2\beta)(1+\beta)^{1/2}}{(1-\beta)^{3/2}} \le 2$ for $\beta\in(0,1/6)$ and $\left\|\frac{x(\mu)_B}{x(\widetilde{\mu})_B}\right\|_\infty = \frac{x(\mu)_i}{x(\mu_0)_i}\frac{x(\mu_0)_i}{x(\widetilde{\mu})_i}$ for the maximizing index i (as well as the analogous terms of the other inequalities).

Plugging these estimates into Lemma 5.4.6 yields the result. $\square$

Simple properties of the subspace LLS step

Here and in the rest of the analysis, $\|\cdot\|$ denotes the usual $\ell_2\to\ell_2$ operator norm. By the choice of V and U in the algorithm, and using the second inequality of Lemma 5.4.5 we obtain the following bound.

Lemma 5.4.8. $\left\|\ell_z|_V\right\|, \left\|\ell_z^\perp|_U\right\| \le \tau$.

A simple and useful bound on δ_B^x and δ_N^s is the following.

Lemma 5.4.9. *We have*

$$\|\delta^V\|, \|\delta^U\| \le \sqrt{n}\,,$$

and

$$\|\delta_B^x\| \le \tau\sqrt{n}, \quad \textit{and} \quad \|\delta_N^s\| \le \tau\sqrt{n}\,.$$

Proof. By definition, δ^V is the projection of some coordinates of ξ, and therefore $\|\delta^V\| \le \|\xi\| = \sqrt{n}$ according to Proposition 5.2.10. Further, $\|\delta_B^x\| = \|\ell_z(\delta^V)\|$. Thus, Lemma 5.4.8 yields $\|\delta_B^x\| \le \tau\|\delta^V\| \le \tau\sqrt{n}$. The analogous arguments give the bounds $\|\delta^U\|$ and $\|\delta_N^s\|$. $\square$

The empirical gradient and its projections

For a point $z = (x, s) \in \mathcal{N}(\beta)$ such that $\overline{\mu}(z) \in [\mu_1, \mu_0]$, the *empirical gradient* is the direction that points directly to the endpoint of the polarized segment $z^{\mathrm{cp}}(\mu_1)$, that is,

$$\Delta x^{\mathrm{emp}} := x^{\mathrm{cp}}(\mu_1) - x\,, \quad \text{and} \quad \Delta s^{\mathrm{emp}} := s^{\mathrm{cp}}(\mu_1) - s\,.$$

This would be a desirable direction to directly shoot down along the polarized segment, however, it requires explicit knowledge of the point $z^{\mathrm{cp}}(\mu_1)$. Nevertheless, a key idea of the analysis is to measure the discrepancy between the subspace LLS direction and the empirical gradient. We will analyze the quantities

$$\psi^{\overline{V}} := \|\Pi_{\overline{V}}(\xi_N - \hat{x}_N^{-1} x^{\mathrm{cp}}(\mu_1)_N)\| \quad \text{and} \quad \psi^{\overline{U}} := \|\Pi_{\overline{U}}(\xi_B - \hat{s}_B^{-1} s^{\mathrm{cp}}(\mu_1)_B)\|\,.$$

These are the projections of a rescaling of the empirical gradient onto the respective "non-cheap" subspaces. These quantities are crucial in proving Theorem 5.4.14. Lemma 5.4.11 shows that if both $\psi^{\overline{V}}$ and $\psi^{\overline{U}}$ are small, then the partition associated to the affine scaling direction matches the polarizing partition. Furthermore, if both are small then we make significant progress on the central path (Theorem 5.4.13). Intuitively, in this case 'most of' $\Delta x_N^{\mathrm{emp}}/\hat{x}_N$ is in V and 'most of' $\Delta s_B^{\mathrm{emp}}/\hat{s}_B$ is in U. Since the subspace LLS step is optimized subject to the constraints $\Delta x_N/\hat{x}_N \in V$, $\Delta s_N/\hat{s}_N \in U$, we can move in this direction nearly as far as with the empirical gradient.

On the other hand, if one of $\psi^{\overline{V}}$ and $\psi^{\overline{U}}$ is relatively big, then the fact that the rescaled empirical gradient has lifting costs that are bounded by n shows that one of the non-cheap subspaces $\overline{V}$ or $\overline{U}$ contains a direction that is *not too expensive*.

Partition identification

This section is concerned about the difference between the partition associated to the affine scaling direction and the polarizing partition. We will show that if the residuals ϱ_B^U and ϱ_N^V are small (recall their definition in (5.27)), then both partitions coincide (Lemma 5.4.10). This in turn can be used to show that these two partitions coincide, if the norm of the projections of the empirical gradient $\psi^{\overline{V}}$ and $\psi^{\overline{U}}$ are small (Lemma 5.4.11).

Lemma 5.4.10 (Identify partition). *Let $z = (x, s) \in \mathcal{N}(\beta)$, $\beta \in (0, 1/2]$, $t \in [0, \tau]$ and define $V := V_{z,N}(t)$ and $U := U_{z,B}(t)$. If $\|\varrho_B^U\| + \|\varrho_N^V\| \le 1/16$, then the associated partition (see Definition 5.4.4) $\widetilde{B} = \{ i \in [n] : |\Delta x_i^{\mathrm{a}}/x_i| < |\Delta s_i^{\mathrm{a}}/s_i| \}$, $\widetilde{N} = [n] \setminus \widetilde{B}$ agrees with the polarizing partition, that is $B = \widetilde{B}$, $N = \widetilde{N}$.*

Proof. Since $\hat{x}^{-1}\Delta x^{\mathrm{a}}, \delta^x \in \hat{x}^{-1}W$ and $\hat{s}^{-1}\Delta s^{\mathrm{a}}, \delta^s \in \hat{s}^{-1}W^\perp$ are in orthogonal spaces, we have that

$$\|\delta^x - \hat{x}^{-1}\Delta x^{\mathrm{a}}\|^2 + \|\delta^s - \hat{s}^{-1}\Delta s^{\mathrm{a}}\|^2 = \|\delta^x + \delta^s - (\hat{x}^{-1}\Delta x^{\mathrm{a}} + \hat{s}^{-1}\Delta s^{\mathrm{a}})\|^2 \overset{(5.15)}{=} \|\delta^x + \delta^s + \xi\|^2\,.$$

By the triangle inequality

$$\begin{aligned}\|\delta^x + \delta^s + \xi\| &\le (\|\delta_B^s + \xi_B\| + \|\delta_B^x\|) + (\|\delta_N^x + \xi_N\| + \|\delta_N^s\|) \\ &= \|\varrho_B^U\| + \|\varrho_N^V\| + \|\delta_B^x\| + \|\delta_N^s\| \le \frac{1}{16} + 2\sqrt{n}\tau = \frac{1}{16} + \frac{1}{8}\beta \le \frac{1}{8}\,,\end{aligned}$$

where we used Lemma 5.4.9 to bound $\|\delta_B^x\|$ and $\|\delta_N^s\|$. Recall from Proposition 5.2.10 that $|\hat{x}_i/x_i|, |\hat{s}_i/s_i| \le \sqrt{1+\beta} < 2$ for each $i \in [n]$. Therefore, for $i \in N$, we have

$$\left|\frac{\Delta s_i^a}{s_i}\right| \le 2\left|\frac{\Delta s_i^a}{\hat{s}_i}\right| \le 2\left(|\delta_i^s| + \frac{1}{8}\right) \le \frac{1}{4} + 2\sqrt{n}\tau < \frac{1}{2}, \tag{5.45}$$

and for $i \in B$, we have

$$\left|\frac{\Delta x_i^a}{x_i}\right| \le 2\left|\frac{\Delta x_i^a}{\hat{x}_i}\right| \le 2\left(|\delta_i^x| + \frac{1}{8}\right) \le \frac{1}{4} + 2\sqrt{n}\tau < \frac{1}{2}. \tag{5.46}$$

Since $\Delta x^a/x + \Delta s^a/s = -\mathbf{1}$, we must have $\max\{|\Delta x_i^a/x_i|, |\Delta s_i^a/s_i|\} \ge 1/2$, $\forall i \in [n]$. Combining with (5.45) and (5.46), we have that $|\Delta x_i^a/x_i| < |\Delta s_i^a/s_i|$, $\forall i \in B$, and $|\Delta x_i^a/x_i| > |\Delta s_i^a/s_i|$, $\forall i \in N$, as needed. □

Lemma 5.4.11 (Partition indentification via empirical gradient). *Let $t \in [0, \tau]$ and define $V := V_{z,N}(t)$ and $U := U_{z,B}(t)$. We have*

$$\|\varrho_B^U\| + \|\varrho_N^V\| \le \psi^{\overline{V}} + \psi^{\overline{U}} + 4n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu}.$$

If

$$\psi^{\overline{V}} + \psi^{\overline{U}} + 4n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu} \le \frac{1}{16},$$

then the associated partition $(\widetilde{B}, \widetilde{N})$ coincides with the polarizing partition (B, N), that is $B = \widetilde{B}$, $N = \widetilde{N}$.

Proof. The second part is immediate from the first part and Lemma 5.4.10. We have $\xi_N = \varrho_N^V - \delta^V$. By definition, $\delta^V = -\Pi_V(\xi_N)$. Since V and $\overline{V}$ are orthogonal complements, it follows that $\varrho_N^V = \Pi_{\overline{V}}(\xi_N)$. Hence,

$$\begin{aligned}\psi^{\overline{V}} &= \|\Pi_{\overline{V}}(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\| = \|\varrho_N^V - \Pi_{\overline{V}}(\hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\| \\ &\ge \|\varrho_N^V\| - \|\hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\| \ge \|\varrho_N^V\| - 2n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu}.\end{aligned} \tag{5.47}$$

Analogously $\xi_B = \varrho_B^U = -\delta^U$ and so

$$\psi^{\overline{U}} \ge \|\varrho_B^U\| - 2n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu}.$$

□

Large singular values vs empirical gradient

The following lemma helps us to upper bound either the smallest singular value or ψ on either the primal or dual side.

Lemma 5.4.12. *For the polarizing partition $B \cup N = [n]$ let $Y, \overline{Y} \subseteq \mathbb{R}^N$ be any subspaces such that $Y \perp \overline{Y}$, $Y \oplus \overline{Y} = \pi_N(\hat{x}^{-1}W)$ and analogously let $Z, \overline{Z} \subseteq \mathbb{R}^B$ such that $Z \perp \overline{Z}$, $Z \oplus \overline{Z} = \pi_B(\hat{s}^{-1}W^\perp)$. Let $\sigma^{\overline{Y}}$ be the smallest singular value of $\ell_z|_{\overline{Y}}$ and analogously let $\sigma^{\overline{Z}}$ be the smallest singular value of $\ell_z^\perp|_{\overline{Z}}$. Then,*

$$\begin{aligned}\psi^{\overline{Y}}\sigma^{\overline{Y}} &\le 4n + \left(2\mu_1\mu^{-1}\gamma^{-1} + 1\right)n^{1.5}\|\ell_z|_Y\|, \\ \psi^{\overline{Z}}\sigma^{\overline{Z}} &\le 4n + \left(2\mu_1\mu^{-1}\gamma^{-1} + 1\right)n^{1.5}\|\ell_z|_Z\|.\end{aligned}$$

Proof. We only prove the first statement, since the second statement can be shown analogously. From Lemma 5.2.4, we have

$$\begin{aligned}\|\ell_z(\xi_N - \hat{x}_N^{-1}x(\mu_1)_N)\| &\le \|\xi_B - \hat{x}_B^{-1}x(\mu_1)_B\| \le \|\xi_B\| + \|\hat{x}_B^{-1}x(\mu_1)_B\| \\ &\le \sqrt{n} + (1+\beta)\|x^{-1}(\mu)_B x(\mu_1)_B\| \le 4n\,.\end{aligned} \tag{5.48}$$

Since $\sigma^{\overline{Y}}$ is the smallest singular value of $\ell_z\big|_{\overline{Y}}$, we can bound

$$\begin{aligned}\psi^{\overline{Y}}\sigma^{\overline{Y}} &\le \|\ell_z(\Pi_{\overline{Y}}(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N))\| \\ &= \|\ell_z(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N) - \ell_z(\Pi_Y(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N))\| \\ &\le \|\ell_z(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\| + \|\ell_z(\Pi_Y(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N))\| \\ &\overset{(5.48)}{\le} 4n + \|\ell_z\big|_Y\|\|\Pi_Y(\xi_N - \hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\| \\ &\le 4n + \|\ell_z\big|_Y\|\Big(\|\Pi_Y(\xi_N)\| + \|\Pi_Y(\hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\|\Big) \\ &\le 4n + \|\ell_z\big|_Y\|\Big(\sqrt{n} + \|\Pi_Y(\hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N)\|\Big)\,.\end{aligned} \tag{5.49}$$

Where the last inequality used $\|\Pi_Y(\xi_N)\| \le \|\xi_N\| \le \|\xi\| = \sqrt{n}$. From Proposition 5.2.10 and Corollary 5.3.3 we have that

$$\|\hat{x}_N^{-1}x^{\mathrm{cp}}(\mu_1)_N\| \le \sqrt{1+\beta}\|x(\mu)_N^{-1}x^{\mathrm{cp}}(\mu_1)_N\| \le \sqrt{(1+\beta)n}\|x(\mu)_N^{-1}x^{\mathrm{cp}}(\mu_1)_N\|_\infty \le 2n^{1.5}\mu_1\mu^{-1}\gamma^{-1}\,. \tag{5.50}$$

The claim follows by substituting this in (5.49). □

5.4.2 Analysis

The main result of this section is Theorem 5.4.14, which shows that in case that the residuals are small enough, then within a single step we either come within polynomial factors of the end of the polarized segment, or the smallest singular value of the expensive subspace becomes polynomial. Theorem 5.4.1 is then a simple consequence and will eventually be proven on Page 130.

We introduce the special index $\zeta(z) \in [n]$ as

$$\zeta(z) := 1 + \max\left\{ i : \sigma(z)_i \le \frac{\tau}{n^{1.5}} \right\}. \tag{5.51}$$

Then $\zeta(z) - 1$ corresponds to the number of non-zero singular values of ℓ_z (and $\ell_z^\perp$) that are not larger than $\frac{\tau}{n^{1.5}}$.

Theorem 5.4.13. *Let $\mathrm{CP}[\mu_1, \mu_0]$ be γ-polarized and let $z = (x, s) \in \mathcal{N}(\beta)$ be an iterate with parameter $\overline{\mu}(z) \in (\mu_1, \mu_0)$, and let $z^+ = (x^+, s^+)$ be the next iterate of IPM with subspace LLS run with subspaces V and U. Let $t \in [0, \tau]$ and let $Y := V_{z,N}(t)$ and $Z := U_{z,B}(t)$. If*

$$\psi^{\overline{Y}} + \psi^{\overline{Z}} + 4n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu} \le \frac{1}{16}\,, \tag{5.52}$$

then

$$\overline{\mu}(z^+) = O\Big(n^{4.5}\beta^{-1}\gamma^{-1}\Big(\mu_1 + \sigma(z)^{-1}_{\zeta(z)}\mu\Big)\Big)\,.$$

Proof. By Lemma 5.4.11, $\|\varrho_B^Z\| + \|\varrho_N^Y\| \le 1/16$, and the associated partition agrees with the polarizing partition: $\widetilde{B} = B$, $\widetilde{N} = N$.

From Proposition 5.2.10 we can bound

$$\begin{aligned}\|\delta_N^x\|_\infty = \|\varrho_N^V - \xi_N\|_\infty \le \|\varrho_N^V\|_\infty + \|\xi\|_\infty \le \frac{1}{16} + \sqrt{1+\beta} \le 2\,,\\ \|\delta_B^s\|_\infty = \|\varrho_B^U - \xi_B\|_\infty \le \|\varrho_B^U\|_\infty + \|\xi\|_\infty \le \frac{1}{16} + \sqrt{1+\beta} \le 2\,.\end{aligned} \tag{5.53}$$

Recall from Lemma 5.4.9 that

$$\|\delta_B^x\| \le \tau\sqrt{n}\,, \quad \text{and} \quad \|\delta_N^s\| \le \tau\sqrt{n}\,.$$

Consequently,

$$\begin{aligned}\left\|\frac{\Delta x^\ell \Delta s^\ell}{\mu}\right\|^2 = \|\delta^x\delta^s\|^2 = \|\delta_B^x\delta_B^s\|^2 + \|\delta_N^x\delta_N^s\|^2 \le \|\delta_B^x\|^2\|\delta_B^s\|^2 + \|\delta_N^x\|^2\|\delta_N^s\|^2\\ \le 4(\|\delta_B^x\|^2 + \|\delta_N^x\|^2) \le 8n\tau^2\,.\end{aligned} \tag{5.54}$$

Note that this bound gives us for $(\Delta x^\ell, \Delta s^\ell)$ the assumptions of Lemma 5.2.12 as $\sqrt{8n}\tau \le \beta/4$.

Further, note that

$$\begin{aligned}\left\|\frac{(x+\Delta x^\ell)(s+\Delta s^\ell)}{\mu}\right\|^2 &= \left\|\frac{(x_B+\Delta x_B^\ell)(s_B+\Delta s_B^\ell)}{\mu}\right\|^2 + \left\|\frac{(x_N+\Delta x_N^\ell)(s_N+\Delta s_N^\ell)}{\mu}\right\|^2\\ &= \|(\xi_B+\delta_B^x)(\xi_B+\delta_B^s)\|^2 + \|(\xi_N+\delta_N^x)(\xi_N+\delta_N^s)\|^2\\ &= \|\varrho_B^V\varrho_B^U\|^2 + \|\varrho_N^V\varrho_N^U\|^2\\ &\le \|\varrho_B^V\|_\infty^2\|\varrho_B^U\|^2 + \|\varrho_N^V\|^2\|\varrho_N^U\|_\infty^2\\ &\le 4\Big(\|\varrho_B^U\|^2 + \|\varrho_N^V\|^2\Big)\\ &\le 4\Big(\|\varrho_B^U\| + \|\varrho_N^V\|\Big)^2,\end{aligned} \tag{5.55}$$

where the penultimate inequality used that

$$\|\varrho_B^V\|_\infty \le \|\xi_B\|_\infty + \|\delta_B^x\|_\infty \le \sqrt{1+\beta} + \sqrt{n}\tau \le 2\,, \tag{5.56}$$

due to Proposition 5.2.10 and Lemma 5.4.9. Analogously, $\|\varrho_N^U\|_\infty \le 2$. Together with Lemma 5.4.11, we get

$$\left\|\frac{(x+\Delta x^\ell)(s+\Delta s^\ell)}{\mu}\right\| \le 2\psi^{\overline{V}} + 2\psi^{\overline{U}} + 8n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu}\,.$$

Lemma 5.2.12 guarantees a step-length

$$\alpha = 1 - \frac{4}{\beta}\left\|\frac{(x+\Delta x^\ell)(s+\Delta s^\ell)}{\mu}\right\| \tag{5.57}$$

such that $z^+ := (x+\alpha\Delta x^\ell, s+\alpha\Delta s^\ell) \in \mathcal{N}(2\beta)$ and

$$\begin{aligned}\overline{\mu}(z^+) \le \left(1 + \frac{3}{2}\frac{\beta}{\sqrt{n}}\right)(1-\alpha)\mu \le \frac{8\mu}{\beta}\left\|\frac{(x+\Delta x^\ell)(s+\Delta s^\ell)}{\mu}\right\|\\ \le \frac{16}{\beta}\Big(4\sqrt{n}\mu_1\gamma^{-1} + \psi^{\overline{V}}\mu + \psi^{\overline{U}}\mu\Big).\end{aligned} \tag{5.58}$$

Using that $(\widetilde{B}_z, \widetilde{N}_z) = (B, N)$ and therefore $\widetilde{\ell}_z = \ell_z$, $\widetilde{\ell}_s^\perp = \ell_s^\perp$, Lemma 5.4.8 implies $\|\ell_z^\perp|_U\|, \|\ell_z|_V\| \leq \tau$. Let $p := \dim(\overline{V})$, $r := \dim(\overline{U})$ and $q := \dim(\mathrm{im}(\ell_z)) = \dim(\mathrm{im}(\ell_z^\perp))$. Note that by Lemma 5.4.5 we have

$$\sigma(z)_{q-p+1} \geq \sigma^{\overline{V}} \geq \frac{1}{\sqrt{n}} \frac{\|\ell_z(\mathbf{V}_{|N|-p+1})\|}{\|\mathbf{V}_{|N|-p+1}\|} > \frac{1}{\sqrt{n}} \frac{\tau}{n} = \frac{\tau}{n^{1.5}}, \tag{5.59}$$

where the third inequality follows by definition of V. This in particular implies that $q - p + 1 \geq \zeta(z)$ and so $\sigma(z)_{q-p+1} \geq \sigma(z)_{\zeta(z)}$. Further, we have by Lemma 5.4.5 that

$$\sigma^{\overline{V}} \geq \frac{1}{n^{1.5}} \sigma(z)_{q-p+1}. \tag{5.60}$$

Analogously, we have that

$$\sigma(z)_{r-p+1} \geq \sigma(z)_{\zeta(z)} \quad \text{and} \quad \sigma^{\overline{U}} \geq \frac{1}{n^{1.5}} \sigma(z)_{r-p+1}. \tag{5.61}$$

Together with Lemma 5.4.12 with $Y := V, Z := U$, we have

$$\begin{aligned} \psi^{\overline{V}} \sigma(z)_{\zeta(z)} \leq \psi^{\overline{V}} \sigma(z)_{q-p+1} &\overset{(5.60)}{\leq} n^{1.5} \psi^{\overline{V}} \sigma^{\overline{V}}(z) \leq n^{1.5}(4n + \left(2\mu_1 \mu^{-1} \gamma^{-1} + 1\right) n^{1.5} \tau) \\ &\leq n^{1.5}(4n + \left(2\gamma^{-1} + 1\right) n^3 \sigma(z)_{\zeta(z)}), \end{aligned} \tag{5.62}$$

where the last inequality follows by definition of the index $\zeta(z)$. Analogously we get on the dual side

$$\psi^{\overline{U}} \sigma(z)_{\zeta(z)} \leq n^{1.5}(4n + \left(2\gamma^{-1} + 1\right) n^3 \sigma(z)_{\zeta(z)}). \tag{5.63}$$

Plugging these into (5.58), we get

$$\overline{\mu}(z^+) \leq \frac{256}{\beta\gamma} \left(\sqrt{n}\mu_1 + n^{4.5} \mu \sigma(z)^{-1}_{\zeta(z)}\right), \tag{5.64}$$

which proves the theorem. □

With Theorem 5.4.13 at hand we can show Theorem 5.4.14, which shows that if the residuals are not small, then this is easy to handle and we can nonetheless make a similar conclusion to the one of Theorem 5.4.13. Theorem 5.4.14 shows that we have two alternatives after a single step in the algorithm.

Let $\mu := \overline{\mu}(z)$ throughout.

Theorem 5.4.14. *Let CP$[\mu_1, \mu_0]$ be γ-polarized with polarizing partition $B \cup N = [n]$. Then, given an iterate $z = (x, s) \in \mathcal{N}(\beta)$ with parameter $\overline{\mu}(z) \in (\mu_1, \mu_0)$ in IPM with subspace LLS (Algorithm 5.1), the next iterate $z^+ = (x^+, s^+) \in \mathcal{N}(\beta)$ satisfies one of the following properties:*

(i) $\sigma(z^+)_{\zeta(z)} = O(n^{6.5} \beta^{-1} \gamma^{-2})$,

(ii) $\overline{\mu}(z^+) \leq O(n^{4.5} \beta^{-1} \gamma^{-2} \mu_1)$.

Proof of Theorem 5.4.14. Algorithm 5.1 runs the subspace LLS with partition $\widetilde{B} \cup \widetilde{N} = [n]$. Let $V := V_{z,N}(\tau/n)$ and $U = U_{z,B}(\tau/n)$ (defined in Definition 5.4.3). We distinguish two cases based on whether the terms $\psi^{\overline{V}}$ and $\psi^{\overline{U}}$ are small.

Case I. $\psi^{\overline{V}} + \psi^{\overline{U}} + 4n^{1.5} \gamma^{-1} \frac{\mu_1}{\mu} > \frac{1}{16}$.

Then, by Lemma 5.4.12 with $Y := V_{z,N}(\tau/n)$ and $Z := U_{z,B}(\tau/n)$, we have that

$$2 \cdot \Big(4n + \Big(2\mu_1\mu^{-1}\gamma^{-1} + 1\Big)n^{1.5}(\tau/n)\Big)\sigma(z)^{-1}_{\zeta(z)} + 4n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu} \geq \frac{1}{16} \ . \tag{5.65}$$

In particular, for the next iterate z^+, we have $\overline{\mu}(z^+) \leq \mu = O(n^{1.5}\gamma^{-1}(\sigma^{-1}_{\zeta(z)}\mu + \mu_1))$.

Case II. $\psi^{\overline{V}} + \psi^{\overline{U}} + 4n^{1.5}\gamma^{-1}\frac{\mu_1}{\mu} \leq \frac{1}{16}$.

By Theorem 5.4.13 we have that $\overline{\mu}(z^+) = O\Big(n^{4.5}\beta^{-1}\gamma^{-1}\Big(\mu_1 + \sigma^{-1}_{\zeta(z)}\mu\Big)\Big)$.

In either case, we have

$$\overline{\mu}(z^+) = O\Big(n^{4.5}\beta^{-1}\gamma^{-1}\Big(\mu_1 + \sigma^{-1}_{\zeta(z)}\mu\Big)\Big) \ . \tag{5.66}$$

If $\mu_1 \geq \sigma^{-1}_{\zeta(z)}\mu$, then $\overline{\mu}(z^+) = O(n^{4.5}\beta^{-1}\gamma^{-1}\mu_1)$, so we are in case (ii). Otherwise, $\overline{\mu}(z^+) = O(n^{4.5}\beta^{-1}\gamma^{-1}\sigma^{-1}_{\zeta(z)}\mu)$. By Lemma 5.4.7 we have that $\sigma(z^+) \leq 4n^2\gamma^{-2}\overline{\mu}(z^+)\mu^{-1}\sigma(z)$. In particular,

$$\sigma(z^+)_{\zeta(z)} \leq 4n^2\gamma^{-2}\overline{\mu}(z^+)\mu^{-1}\sigma(z)_{\zeta(z)} = O(n^{6.5}\beta^{-1}\gamma^{-2}). \tag{5.67}$$

Hence, we are in case (i). This proves the theorem. □

From Theorem 5.4.14 we can now derive the proof of the main theorem as follows.

Proof of Theorem 5.4.1. Consider any two consecutive iterates $z = (x, s)$ and $z^+ = (x^+, s^+)$ in IPM with subspace LLS. In case (ii) of Theorem 5.4.14, the end of the polarized segment will be reached within $O(\beta^{-1}\sqrt{n}\log(n/\gamma))$ iterations, since the affine scaling step itself leads to an $1 - \Omega(\beta/\sqrt{n})$ decrease in the normalized gap, and we always choose the better of the two steps.

In case (i) of Theorem 5.4.14, we have $\sigma(z^+)_{\zeta(z)} = O(n^{5.5}\beta^{-1}\gamma^{-1})$. Using Lemma 5.4.7, after $O(\beta^{-1}\sqrt{n}\log(n/\gamma))$ additional iterations, any subsequent iterate $z^{++} = (x^{++}, s^{++})$ satisfies $\sigma(z^{++})_{\zeta(z)} < \tau/n^{1.5}$. By definition of $\zeta(\cdot)$ in (5.51), we have $\zeta(z^{++}) > \zeta(z)$ for all these iterates. Such an event may occur at most n times. □

5.5 The Max Central Path

In this section, we deal with the properties of the max central path that we introduced in Section 5.1. Given $g \geq 0$, we denote by

$$\begin{aligned}\mathcal{P}_g &:= \{x \in \mathbb{R}^n : \mathbf{A}x = b \ , \ x \geq \mathbf{0} \ , \ \langle c, x\rangle \leq v^\star + g\} \ , \\ \mathcal{D}_g &:= \{s \in \mathbb{R}^m : \exists y \ \mathbf{A}^\top y + s = c \ , \ s \geq \mathbf{0} \ , \ \langle b, y\rangle \geq v^\star - g\}\end{aligned}$$

the feasible sets of the linear programs in (5.4). They correspond to the sets of the primal and dual feasible points $(x, s) \in \mathcal{P} \times \mathcal{D}$ with objective value within g from the optimum $v^\star$, respectively.

We recall that the duality gap of any pair $(x, (y, s))$ of primal-dual feasible points of LP fulfills $\langle c, x\rangle - \langle b, y\rangle = \langle x, s\rangle$. In particular, we have $\langle x, s^\star\rangle = \langle c, x\rangle - v^\star$ and $\langle x^\star, s\rangle = v^\star - \langle b, y\rangle$. Thus, the two sets $\mathcal{P}_g$ and $\mathcal{D}_g$ are equivalently given by

$$\mathcal{P}_g = \{x \in \mathcal{P} : \langle x, s^\star\rangle \leq g\} \ , \quad \mathcal{D}_g = \{s \in \mathcal{D} : \langle x^\star, s\rangle \leq g\} \ .$$

These expressions are in fact independent of the choice of optimal solutions $(x^\star, s^\star)$. The following claim is immediate by our assumption that $\mathcal{P}_{++}$ and $\mathcal{D}_{++}$ are non-empty.

Proposition 5.5.1. *For all $g \geq 0$, the sets $\mathcal{P}_g$ and $\mathcal{D}_g$ are bounded.*

We denote by MCP $:= \{ z^{\mathrm{m}}(g) : g \geq 0 \}$ the whole max central path. The max central path point $z^{\mathrm{m}}(g) = (x^{\mathrm{m}}(g), s^{\mathrm{m}}(g))$ is the entry-wise maximum of the set $\mathcal{P}_g \times \mathcal{D}_g$.

While the points of the max central path are not feasible in general, the following theorem shows that the max central path shares important similarities with the central path:

Theorem 5.5.2 (Centrality of the max central path)**.** *For all $g \geq 0$, we have that*

$$g \leq x_i^{\mathrm{m}}(g) s_i^{\mathrm{m}}(g) \leq 2g \quad \forall i \in [n] .$$

Proof. We first prove the upper bound. For $i \in [n]$, let $x^{(i)} \in \arg\max\{x_i : x \in \mathcal{P}_g\}$ and $s^{(i)} \in \arg\max\{s_i : s \in \mathcal{D}_g\}$. Note that $x^{(i)}, s^{(i)}$ exist by Proposition 5.5.1. Then,

$$x_i^{\mathrm{m}}(g) s_i^{\mathrm{m}}(g) = x_i^{(i)} s_i^{(i)} \leq \left\langle x^{(i)}, s^{(i)} \right\rangle = \left\langle x^{(i)}, s^\star \right\rangle + \left\langle x^\star, s^{(i)} \right\rangle \leq 2g ,$$

where the last equality follows from Proposition 5.2.1. We now prove the lower bound. We assume $g > 0$, since the statement is trivial otherwise.

Note that the dual program of $\max\{ x_i : x \in W + d, x \geq 0, \langle x, s^\star \rangle \leq g \}$ can be expressed as

$$\min\left\{ \alpha g + \left\langle u, x^\star \right\rangle : \alpha s^\star + u \geq e^i, u \in W^\perp, \alpha \geq 0 \right\} ,$$

using that $\langle u, x^\star \rangle = \langle u, d \rangle$ since $d - x^\star \in W$, $u \in W^\perp$. Similarly, the dual program of $\max\{s_i : s \in W^\perp + c, s \geq 0, \langle s, x^\star \rangle \leq g\}$ can be expressed as

$$\min\left\{ \beta g + \left\langle v, s^\star \right\rangle : \beta x^\star + v \geq e^i, v \in W, \beta \geq 0 \right\} .$$

Let us pick optimal (α, u) and (β, v) to these two programs. The product of the objective values is thus equal to $x_i^{\mathrm{m}}(g) s_i^{\mathrm{m}}(g)$; the proof is complete by showing a lower bound g.

We first claim that

$$\left\langle u, x^\star \right\rangle \geq 0 \quad \text{and} \quad \left\langle v, s^\star \right\rangle \geq 0 . \tag{5.68}$$

By symmetry, it suffices to prove the first claim. For a contradiction, assume $\langle u, x^\star \rangle < 0$. Then, there exists an index $j \in [n]$ such that $x_j^\star > 0$ and $u_j < 0$. By complementarity, $s_j^\star = 0$. This contradicts $\alpha s_j^\star + u_j \geq e_j^i$.

Next, note that the constraints in the two programs imply

$$1 = \left\langle e^i, e^i \right\rangle \leq \left\langle \alpha s^\star + u, \beta x^\star + v \right\rangle = \alpha \left\langle v, s^\star \right\rangle + \beta \left\langle u, x^\star \right\rangle . \tag{5.69}$$

Now, the product of the objective values can be written as

$$\begin{aligned} x_i^{\mathrm{m}}(g) s_i^{\mathrm{m}}(g) &= (\alpha g + \left\langle u, x^\star \right\rangle)(\beta g + \left\langle v, s^\star \right\rangle) \\ &= \alpha\beta g^2 + g\left(\alpha \left\langle v, s^\star \right\rangle + \beta \left\langle u, x^\star \right\rangle\right) + \left\langle u, x^\star \right\rangle \cdot \left\langle v, s^\star \right\rangle \geq g , \end{aligned}$$

using inequalities (5.68) and (5.69). This concludes the proof. □

Given the above, we have the following straightforward relation between the max central path and central path.

Lemma 5.5.3. *For $\mu > 0$, we have that*

$$z^{\mathrm{m}}(n\mu) \geq z^{\mathrm{cp}}(\mu) \geq \frac{z^{\mathrm{m}}(n\mu)}{2n} .$$

Proof. Recall that $z^{\mathrm{cp}}(\mu) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu))$ with $\langle x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu)\rangle = \langle x^{\mathrm{cp}}(\mu), s^\star\rangle + \langle x^\star, s^{\mathrm{cp}}(\mu)\rangle = n\mu$ using Proposition 5.2.1. Therefore, $x^{\mathrm{cp}}(\mu) \in \mathcal{P}_{n\mu}$ and $s^{\mathrm{cp}}(\mu) \in \mathcal{D}_{n\mu}$. By definition of the max central path, $z^{\mathrm{cp}}(\mu) = (x^{\mathrm{cp}}(\mu), s^{\mathrm{cp}}(\mu)) \leq (x^{\mathrm{m}}(n\mu), s^{\mathrm{m}}(n\mu)) = z^{\mathrm{m}}(n\mu)$. For the second inequality, note that

$$x^{\mathrm{cp}}(\mu) = \frac{\mu}{s^{\mathrm{cp}}(\mu)} \geq \frac{\mu}{s^{\mathrm{m}}(n\mu)} = \frac{\mu}{x^{\mathrm{m}}(n\mu)s^{\mathrm{m}}(n\mu)} x^{\mathrm{m}}(n\mu) \overset{\text{Thm. 5.5.2}}{\geq} \frac{\mu}{2n\mu}\mathbf{1}x^{\mathrm{m}}(n\mu) = \frac{x^{\mathrm{m}}(n\mu)}{2n}.$$

By a symmetric argument, $s^{\mathrm{cp}}(\mu) \geq s^{\mathrm{m}}(n\mu)/2n$. □

5.5.1 The Shadow Vertex Simplex Rule

Given a pointed polyhedron $P \subseteq \mathbb{R}^n$ and two objectives $c^{(1)}, c^{(2)} \in \mathbb{R}^n$ such that $\max_{x\in P}\langle c^{(1)}, x\rangle < \infty$ and $\max_{x\in P}\langle c^{(2)}, x\rangle < \infty$, we recall that the shadow vertex simplex rule consists in pivoting over vertices of P maximizing the objectives $(1-\lambda)c_1 + \lambda c_2$ as λ goes from 0 to 1. More formally, a sequence of vertices $v^{(1)}, \ldots, v^{(k)} \in P$ is a $(c^{(1)}, c^{(2)})$-*shadow vertex path on* P if

- $[v^{(i)}, v^{(i+1)}]$ is an edge of P, $\forall i \in [k-1]$,
- $\langle c^{(2)}, v^{(i)}\rangle < \langle c^{(2)}, v^{(i+1)}\rangle$, $\forall i \in [k-1]$, and
- there exists $0 = \lambda_0 < \lambda_1 \leq \cdots \leq \lambda_{k-1} < \lambda_k = 1$ such that $\forall i \in [k]$, $\langle v^{(i)}, (1-\alpha)c^{(1)} + \alpha c^{(2)}\rangle = \max_{x\in P}\langle x, (1-\alpha)c^{(1)} + \alpha c^{(2)}\rangle$ for all $\alpha \in [\lambda_{i-1}, \lambda_i]$.

To analyze shadow vertex paths further, we define the two-dimensional projection

$$P[c^{(1)}, c^{(2)}] := \left\{ \left(\langle c^{(1)}, x\rangle, \langle c^{(2)}, x\rangle \right) : x \in P \right\} = \left(c^{(1)}, c^{(2)}\right)^\top \cdot P.$$

Under non-degeneracy assumptions (which are easily satisfied by infinitesimally perturbing the constraints), there is a unique shadow vertex path with respect to $c^{(1)}$ and $c^{(2)}$. Non-degeneracy implies that $\lambda_0 < \lambda_1 < \cdots < \lambda_k$ above, and thus the maximizing objective moves strictly closer to $c^{(2)}$ after each simplex step. In this case, the vertices of the shadow vertex simplex path project under the map $x \mapsto (\langle c^{(1)}, x\rangle, \langle c^{(2)}, x\rangle)$ precisely to the subset of the vertices of the 2-dimensional projection $P[c^{(1)}, c^{(2)}]$ maximizing some open interval of objectives $(1-\lambda)e^1 + \lambda e^2$, $\lambda \in [0,1]$, where e^1 and e^2 stand for the unit vectors of $\mathbb{R}^2$. As $P[c^{(1)}, c^{(2)}]$ is the shadow (projection) of P onto the plane spanned by $c^{(1)}, c^{(2)}$, this justifies the name 'shadow vertex simplex rule'. In the general setting, the vertices of $P[c^{(1)}, c^{(2)}]$ maximizing an open interval of objectives in $(1-\lambda)e^1 + \lambda e^2$, $\lambda \in [0,1]$ are precisely the projections of vertices $v^{(i)}$, $i \in [k]$, on the shadow path such that $\lambda_{i-1} < \lambda_i$. The degenerate vertices $v^{(i)}$, $i \in [k]$, such that $\lambda_{i-1} = \lambda_i$, will in fact in general project into the interior of edges of $P[c^{(1)}, c^{(2)}]$.

We define $S_P(c^{(1)}, c^{(2)})$ as the number of vertices of $P[c^{(1)}, c^{(2)}]$ maximizing an open interval of objectives in $(1-\lambda)e^1 + \lambda e^2$, $\lambda \in [0,1]$. By the preceding observations, we have that $S_P(c^{(1)}, c^{(2)})$ is a lower bound on the number of vertices of any $(c^{(1)}, c^{(2)})$-shadow vertex path.

In the above, we restricted both starting and ending objectives $c^{(1)}, c^{(2)}$ to have finite objective value on P. It will be useful in the sequel to extend to the case where $c^{(2)}$ might be unbounded. In this case, we define the shadow vertex path as above, with the only modification being that we let $\lambda_k := \max\{\lambda \in [0,1] : \max_{x\in P}\langle x, (1-\lambda)c^{(1)} + \lambda c^{(2)}\rangle < \infty\}$, that is, the simplex path stops just before reaching an unbounded ray for $c^{(2)}$. In this setting, note that $S_P(c^{(1)}, c^{(2)})$ is still well-defined and continues to be a lower bound on the number of vertices on any $c^{(1)}, c^{(2)}$ shadow vertex path.

The following lemma now gives the main relation between shadow vertex paths and the number of linear segments of the max central path. Precisely, a segment

$$\mathrm{MCP}[g_1, g_0] := \{z^{\mathrm{m}}(g) : g_0 \geq g \geq g_1\}$$

is *linear* if $z^{\mathrm{m}}((1-\alpha)g_0 + \alpha g_1) = (1-\alpha)z^{\mathrm{m}}(g_0) + \alpha z^{\mathrm{m}}(g_1)$, $\forall \alpha \in [0,1]$.

Lemma 5.5.4 (Piecewise Linearity of MCP)**.**

(i) $\forall i \in [n]$, $g \mapsto x_i^{\mathrm{m}}(g)$ *is piecewise linear non-decreasing with* $S_{\mathcal{P}}(-s^\star, e^{(i)})$ *pieces.*

(ii) $\forall i \in [n]$, $g \mapsto s_i^{\mathrm{m}}(g)$ *is piecewise linear non-decreasing with* $S_{\mathcal{D}}(-x^\star, e^{(i)})$ *pieces.*

(iii) $g \mapsto z^{\mathrm{m}}(g)$ *is piecewise linear and entry-wise non-decreasing with at most*

$$\min\left\{\sum_{i=1}^{n} S_{\mathcal{P}}(-s^\star, e^i) + S_{\mathcal{D}}(-x^\star, e^i), \mathcal{V}_{\mathcal{P}} + \mathcal{V}_{\mathcal{D}}\right\}$$

pieces, where $\mathcal{V}_{\mathcal{P}}$ *and* $\mathcal{V}_{\mathcal{D}}$ *denote the number of vertices of* $\mathcal{P}$ *and* $\mathcal{D}$*, respectively.*

Proof. **Proof of (i)** For $i \in [n]$, let $Q_i = P[s^\star, e^i]$. We note that $x_i^{\mathrm{m}}(g) = \max\{v_2 : (v_1, v_2) \in Q_i, v_1 \leq g\}$. In particular, the map x_i^{m} is a non-decreasing concave function of g. Again by definition, $S_{\mathcal{P}}(-s^\star, e^i)$ equals the number of vertices of Q_i maximizing an open interval of objectives in $O := \{-(1-\lambda)e^1 + \lambda e^2 : \lambda \in [0,1]\} \subseteq \mathbb{R}^2$.

Define $\bar{u}_i(g) := \sup\{v_2 : (g, v_2) \in Q_i\}$, which is defined to equal $-\infty$ if $\{(g, v_2) \in Q_i\} = \emptyset$. By Proposition 5.5.1, note that $\bar{u}_i(g) < \infty$ for all $g \geq 0$. By convexity of Q_i, $\bar{u}_i$ is concave function on $\mathbb{R}_+$. Let $h = \sup\{\bar{u}_i(g) : g \geq 0\}$.

Assume $h = \infty$. By concavity, $\bar{u}_i$ must be a strictly increasing function on $\mathbb{R}_+$. In particular, $\bar{u}_i(g) = x_i^{\mathrm{m}}(g)$. Given this, we see that the linear pieces of $x_i^{\mathrm{m}}(g)$ are in one to one correspondence with the edges of Q_i on the upper convex hull whose projection onto the e_1-axis have positive length (i.e., excluding the potential edge $\{(0, v_2) \in Q_i\}$). Since $Q_i \subseteq \mathbb{R}^2_+$, every such edge can be uniquely associated with its left endpoint (which is always a vertex of Q_i). It is now easy to check geometrically that the set of such endpoints exactly corresponds to the set of vertices that are maximizers of the objectives in an open interval of O.

Assume $h < \infty$. Let $g_h := \min\{g \geq 0 : x_i^{\mathrm{m}}(g) = h\}$. It is direct to see that $x_i^{\mathrm{m}}(g) = \bar{u}_i(g)$ if $g \leq g_h$ and that $x_i^{\mathrm{m}}(g) = h$ for $g \geq g_h$. Furthermore, $x_i^{\mathrm{m}}(g)$ is strictly increasing on $[0, g_h]$. From this, it is easy to see geometrically that the number of linear pieces of x_i^{m} is one plus the number of edges of Q_i on the upper convex hull lying in the band $\{(v_1, v_2) : 0 \leq v_1 \leq g\}$, where the extra linear segment corresponds to constant segment between g_h and ∞. As in the previous case, these linear segments can be uniquely identified with their left endpoints, which correspond to vertices of Q_i. Furthermore, it is easy to check that these correspond to vertices of Q_i maximizing an open interval of objectives in O.

Proof of (ii). Same as (i), swapping the role of $\mathcal{D}$ and $x^\star$ for $\mathcal{P}$ and $s^\star$.

Proof of (iii). Let $0 = g_1^{i,p} < g_2^{i,p} < \cdots < g_{k^{i,p}}^{1,p}$ denote the parameters corresponding to breakpoints of linear segments of x_i^{m}, and similarly let $0 = g_1^{i,d} < g_2^{i,d} < \cdots < g_{k^{i,d}}^{1,d}$ correspond to breakpoints for s_i^{m}. Finally, let $0 = g_1 < g_2 < \cdots < g_T$ denote an ordering of the merged sequence of breakpoints (suppressing duplicates) of each $x_i^{\mathrm{m}}, s_i^{\mathrm{m}}$, $\forall i \in [n]$. Since $x_i^{\mathrm{m}}, s_i^{\mathrm{m}}$, $\forall i \in [n]$, is linear on each interval $[g_i, g_{i+1}]$, $i \in [T-1]$, and on the interval $[g_T, \infty)$, we get that z^{m} is also linear on these intervals. Furthermore, they are exactly the breakpoints of the linear segments of z^{m},

since z^{m} is linear on an interval iff $x_i^{\mathrm{m}}, s_i^{\mathrm{m}}, i \in [n]$, are linear on the interval. By associating each linear segment with its left endpoint, we see that the number of linear segments of m is $T \le \sum_{i=1}^{n} k^{i,p} + k^{i,d} = \sum_{i=1}^{n} S_{\mathcal{P}}(-s^\star, e^i) + S_{\mathcal{D}}(-x^\star, e^i)$. Furthermore, note that for each g_j, $j \in [T]$, there exists $i \in [n]$ such that either $(g_j, x_i^{\mathrm{m}}(g_j))$ is a vertex of $\mathcal{P}[s^\star, e^i]$ or $(g_j, s_i^{\mathrm{m}}(g_j))$ is a vertex of $\mathcal{P}[x^\star, e^i]$. In particular, there exists either a vertex x_j of $\mathcal{P}$ such that $\langle s^\star, x_j \rangle = g_j$ or a vertex y_j of $\mathcal{D}$ such that $\langle x^\star, y_j \rangle = g_j$. This association between breakpoints and vertices is injective (since the g_j are all distinct). Therefore, we also get the bound $T \le \mathcal{V}_{\mathcal{P}} + \mathcal{V}_{\mathcal{D}}$ as needed. □

From the above discussion, note that Lemma 5.5.4 implies that the number of linear pieces of the max-central path is at most the number of vertices on $2n$ shadow vertex paths on $\mathcal{P}$ and $\mathcal{D}$.

5.5.2 Direct Proof of Polarization along Max Central Path Segments

In this subsection, we give a simple proof that the central path can be decomposed into T polarized segments, where T is the number of linear segments of the max central path. The proof avoids using the wide neighborhood and instead directly compares the central path with the max central path. By virtue of being more direct, it also achieves a better polarization parameter. Since the polarization parameter appears under a logarithm, this improvement does not change the asymptotics of our algorithm.

Lemma 5.5.5. *Let* MCP$[\mu_1, \mu_0]$ *be a linear segment of the max-central path with* $\mu_0 \ge \mu_1 \ge 0$. *Then* CP$[\mu_1/n, \mu_0/n]$ *is* $1/(16n)$*-polarized.*

Proof. Let us fix $i \in [n]$, and let $u = \frac{x_i^{\mathrm{m}}(\mu_1)}{x_i^{\mathrm{m}}(\mu_0)}$ and $v = \frac{s_i^{\mathrm{m}}(\mu_1)}{s_i^{\mathrm{m}}(\mu_0)}$. Then for $\alpha \in [0,1]$,

$$\begin{aligned}
\frac{(1-\alpha+\alpha u)(1-\alpha+\alpha v)}{1-\alpha+\alpha uv} &= \frac{((1-\alpha)x_i^{\mathrm{m}}(\mu_0)+\alpha x_i^{\mathrm{m}}(\mu_1))((1-\alpha)s_i^{\mathrm{m}}(\mu_0)+\alpha s_i^{\mathrm{m}}(\mu_1))}{(1-\alpha)x_i^{\mathrm{m}}(\mu_0)s_i^{\mathrm{m}}(\mu_0)+\alpha x_i^{\mathrm{m}}(\mu_1)s_i^{\mathrm{m}}(\mu_1)} \\
&= \frac{x_i^{\mathrm{m}}((1-\alpha)\mu_0+\alpha\mu_1)s_i^{\mathrm{m}}((1-\alpha)\mu_0+\alpha\mu_1)}{(1-\alpha)x_i^{\mathrm{m}}(\mu_0)s_i^{\mathrm{m}}(\mu_0)+\alpha x_i^{\mathrm{m}}(\mu_1)s_i^{\mathrm{m}}(\mu_1)} && \text{(by linearity)} \\
&\ge \frac{(1-\alpha)\mu_0+\alpha\mu_1}{2((1-\alpha)\mu_0+\alpha\mu_1)} = \frac{1}{2}. && \text{(by Theorem 5.5.2)}
\end{aligned}$$

Therefore Lemma 5.3.6 yields $u + v \ge \frac{1}{4}$. Let $B := \left\{ i \in [n] : \frac{x_i^{\mathrm{m}}(\mu_1)}{x_i^{\mathrm{m}}(\mu_0)} \ge \frac{s_i^{\mathrm{m}}(\mu_1)}{s_i^{\mathrm{m}}(\mu_0)} \right\}$ and $N := [n] \setminus B$. Thus, $\frac{x_i^{\mathrm{m}}(\mu_1)}{x_i^{\mathrm{m}}(\mu_0)} \ge 1/8, \forall i \in B$, and $\frac{s_i^{\mathrm{m}}(\mu_1)}{s_i^{\mathrm{m}}(\mu_0)} \ge 1/8, \forall i \in N$.

We now show that CP$[\mu_1/n, \mu_0/n]$ is $\gamma = \frac{1}{16n}$-polarized with respect to the partition B, N as above. For $\mu \in [\mu_1/n, \mu_0/n]$ and $i \in B$, we have that

$$\frac{x_i^{\mathrm{cp}}(\mu)}{x_i^{\mathrm{cp}}(\mu_0)} \overset{\text{Lemma 5.5.3}}{\ge} \frac{x_i^{\mathrm{m}}(n\mu)}{2nx_i^{\mathrm{m}}(\mu_0)} \overset{\text{MCP monotonocity}}{\ge} \frac{x_i^{\mathrm{m}}(\mu_1)}{2nx_i^{\mathrm{m}}(\mu_0)} \overset{i\in B}{\ge} \frac{1}{16n},$$

as needed. The inequality $\frac{s_i^{\mathrm{cp}}(\mu)}{s_i^{\mathrm{cp}}(\mu_0)} \ge \frac{1}{16n}$, $i \in N$, $\mu \in [\mu_0/n, \mu_1/n]$, follows by a symmetric argument. □

5.5.3 Converting the Max Central Path into a Wide Neighborhood Path

While the max central path does not correspond to a feasible path inside $\mathcal{P} \times \mathcal{D}$, we now show that it is in fact close to a piecewise linear path that lives inside the wide neighborhood of the central path having the same number of breakpoints.

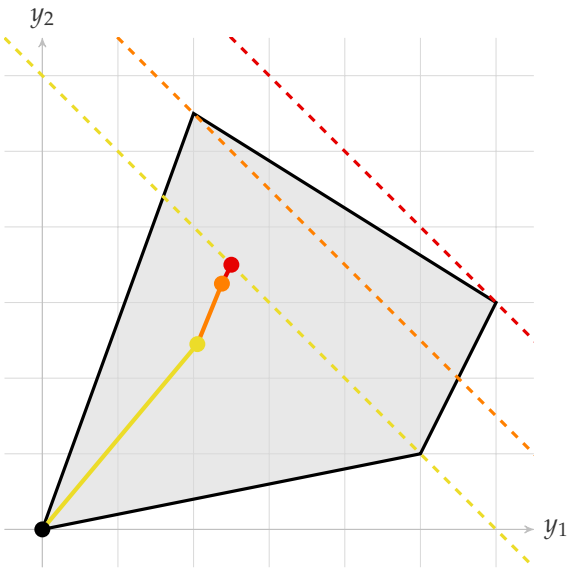


Figure 5.2: The path $\bar{\Gamma}^d(g)$ as defined in the proof of Theorem 5.5.6 for the system max $-y_1 - y_2$ s.t. $A^\top y + s = c, s \geq \mathbf{0}$, where and $A = \begin{bmatrix} 1 & 2 & 2.5 & -5.5 \\ -5 & -1 & 4 & 2 \end{bmatrix}, c = \begin{bmatrix} 0 & 9 & 27 & 0 \end{bmatrix}^\top$. Dashed lines correspond to level sets at breakpoints.

Theorem 5.5.6. *There exists a piecewise linear curve* $\Gamma\colon \mathbb{R}_+ \to \mathcal{N}^{-\infty}(1 - \frac{1}{2n})$ *with at most as many linear segments as* $g \mapsto z^{\mathrm{m}}(g)$ *satisfying* $\mu(\Gamma(s)) = s, \forall s \geq 0$.

Proof. As in the proof of Lemma 5.5.4 part (3), let $0 = g_1 < \cdots < g_T$ denote the breakpoints for $g \mapsto z^{\mathrm{m}}(g)$.

From here, pick $x^{i,j} \in \mathcal{P}_{g_j}, s^{i,j} \in \mathcal{D}_{g_j}$, $i \in [n]$, $j \in [T]$, such that $x_i^{i,j} = x_i^{\mathrm{m}}(g_j)$, $s_i^{i,j} = s_i^{\mathrm{m}}(g_j)$, and such that $\langle x^{i,1}, s^\star \rangle \leq \cdots \leq \langle x^{i,T}, s^\star \rangle$ and $\langle x^\star, s^{i,1} \rangle \leq \cdots \leq \langle x^\star, s^{i,T} \rangle$. Further, for all $i \in [n]$, choose $r^{i,p}$ and $r^{i,d}$ in the recession cone of $\mathcal{P}$ and $\mathcal{D}$ respectively, such that $\langle r^{i,p}, s^\star \rangle \in \{0, 1\}$, $\langle x^\star, r^{i,d} \rangle \in \{0, 1\}$ and $\forall t \geq 0$, $x_i^{\mathrm{m}}(g_T + t) = x_i^{i,T} + t r_i^{i,p}$, $s_i^{\mathrm{m}}(g_T + t) = s_i^{i,T} + t r_i^{i,d}$. Define $\Gamma^{i,p}(g) = \frac{g_{j+1} - g}{g_{j+1} - g_j} x^{i,j} + \frac{g - g_j}{g_{j+1} - g_j} x^{i,j+1}$ if $g_j < g \leq g_{j+1}$, $j \in [T-1]$ and $\Gamma^{i,p}(g) = x^{i,T} + (g - g_T) r^{i,p}$ if $g > g_T$. Define the dual counterpart $\Gamma^{i,d}$ similarly. By construction, $\forall i \in [n]$, we have that $\Gamma^{i,p}, \Gamma^{i,d}$ are piecewise linear with breakpoints $0 < g_1 < \cdots < g_T$, and $\forall g \geq 0$, $\Gamma^{i,p}(g) \in \mathcal{P}_g$, $\Gamma^{i,p}(g)_i = x_i^{\mathrm{m}}(g)$, $\Gamma^{i,d}(g) \in \mathcal{D}_g$, $\Gamma^{i,d}(g)_i = s_i^{\mathrm{m}}(g)$.

From here, define

$$\bar{\Gamma}(g) = \left(\bar{\Gamma}^p(g), \bar{\Gamma}^d(g)\right) := \left(\frac{1}{n}\sum_{i=1}^n \Gamma^{i,p}(g), \frac{1}{n}\sum_{i=1}^n \Gamma^{i,d}(g)\right).$$

We have that $\bar{\Gamma}(g) \in \mathcal{P}_g \times \mathcal{D}_g, \forall g \geq 0$. Therefore, $\forall i \in [n]$, by Proposition 5.2.1 we have

$$\overline{\mu}(\bar{\Gamma}(g)) = \frac{1}{n}\langle \bar{\Gamma}^p(g), \bar{\Gamma}^d(g) \rangle = \frac{1}{n}\left(\langle \bar{\Gamma}^p(g), s^\star \rangle + \langle x^\star, \bar{\Gamma}^d(g) \rangle\right) \leq \frac{2g}{n}.$$

Furthermore, for $i \in [n]$, we have

$$\begin{aligned}\bar{\Gamma}^p(g)_i\bar{\Gamma}^d(g)_i &= \left(\frac{1}{n}\sum_{i=1}^n \bar{\Gamma}^{i,p}(g)_i\right)\left(\frac{1}{n}\sum_{i=1}^n \bar{\Gamma}^{i,d}(g)_i\right)\\ &\geq \frac{1}{n^2}\bar{\Gamma}^{i,p}(g)\bar{\Gamma}^{i,d}(g) = \frac{1}{n^2}x_i^{\mathrm{m}}(g)s_i^{\mathrm{m}}(g) \geq \frac{g}{n^2} \geq \frac{\overline{\mu}(\bar{\Gamma}(g))}{2n}.\end{aligned}$$

In particular, we have that $\bar{\Gamma}(g) \in \mathcal{N}^{-\infty}(1-\frac{1}{2n})$.

Note that by construction $\bar{\Gamma}$ has at most T linear segments, where T is the number of linear segments of the max central path. To construct Γ, we will simply reparametrize $\bar{\Gamma}$ with respect to $\overline{\mu}(\bar{\Gamma}(g))$. By Proposition 5.2.2, $\overline{\mu}(\bar{\Gamma}(g))$ is linear on linear segments of $\bar{\Gamma}$, so it suffices to show that $\overline{\mu}(\bar{\Gamma}(g))$ is a strictly increasing in g. Again, by Proposition 5.2.1, we have that

$$\overline{\mu}(\bar{\Gamma}(g)) = \frac{1}{n^2}\left(\sum_{i=1}^n \left\langle \bar{\Gamma}^{i,p}(g), s^\star\right\rangle + \left\langle x^\star, \bar{\Gamma}^{i,d}(g)\right\rangle\right),$$

which is non-decreasing in g, since each term is non-decreasing by construction. Thus, it suffices to show that one of $\left\langle \bar{\Gamma}^{1,p}(g), s^\star\right\rangle$, $\left\langle x^\star, \bar{\Gamma}^{1,d}(g)\right\rangle$ is increasing. Since $x_1^{\mathrm{m}}(g)s_1^{\mathrm{m}}(g) \geq g$ at least one of x_1^{m} or s_1^{m} must be unbounded. Assume without loss of generality that x_1^{m} is unbounded. By concavity of x_1^{m}, x_1^{m} is strictly increasing. In particular $\bar{\Gamma}^{1,p}(g) \in \mathcal{P}_g$ and $\bar{\Gamma}^{1,p}(g)_i = x_i^{\mathrm{m}}(g)$ implies that $\left\langle \bar{\Gamma}^{1,p}(g), s^\star\right\rangle = g$, which is increasing in g. This proves the lemma. □

5.A Missing Proofs

Proof of Proposition 5.2.1. Since $x - x' \in W$ and $s - s' \in W^\perp$, we have that

$$0 = \langle x - x', s - s'\rangle \Leftrightarrow \langle x, s\rangle + \langle x', s'\rangle = \langle x, s'\rangle + \langle x', s\rangle. \qquad \square$$

Proof of Proposition 5.2.2. Using that $\sum_{i=1}^k \lambda_i = 1$ and the orthogonality of $x^{(i)} - d \in W$ and $s^{(i)} - d \in W^\perp$ for all $i \in [k]$ we first get

$$\begin{aligned}\left\langle \sum_{i=1}^k \lambda_i x^{(i)}, \sum_{j=1}^k \lambda_i s^{(i)}\right\rangle &= \left\langle \sum_{i=1}^k \lambda_i (x^{(i)} - d) + d, \sum_{j=1}^k \lambda_i (s^{(i)} - c) + c\right\rangle\\ &= \langle c, d\rangle + \sum_{i=1}^k \lambda_i \left(\left\langle x^{(i)} - d, c\right\rangle + \left\langle d, s^{(i)} - c\right\rangle\right)\\ &= \sum_{i=1}^k \lambda_i \left(\langle d, c\rangle + \left\langle x^{(i)} - d, c\right\rangle + \left\langle d, s^{(i)} - c\right\rangle\right)\\ &= \sum_{i=1}^k \lambda_i \left\langle x^{(i)}, s^{(i)}\right\rangle.\end{aligned}$$

Division by n yields the respective normalized duality gap. □

Proof of Proposition 5.2.6. By definition of $\mathcal{N}(\beta)$ we have for all $i \in [n]$ that $|\frac{x_i s_i}{\mu} - 1| \leq \|\frac{xs}{\mu} - \mathbf{1}\| \leq \beta$ and so $(1-\beta)\mu \leq x_i s_i \leq (1+\beta)\mu$. □

Proof of Lemma 5.2.14. Note first that by construction, the solution set of $L_I^W(x)$ is non-empty. Furthermore, the minimal norm solution exists and is unique by strict convexity of the square Euclidean norm. Thus, $w = L_I^W(x)$ is well-defined.

We now show that $w = L_I^W(x)$ is a solution to the linear system. We have that $w \in W$ and $w_I = \pi_{\pi_I(W)}(x) = x - \pi_{\pi_I(W)^\perp}(x) \in \pi_I(W)^\perp + x$. It remains to show that $w_J \in \pi_J(W^\perp) = \pi_J((W \cap \mathbb{R}^n_J)^\perp)$. Take any $z \in \mathbb{R}^n_J \cap W$ with $\|z\|^2 = 1$. It suffices to show that $\langle w, z\rangle = \langle w_J, z_J\rangle = 0$. Noting that $w + \beta z \in W$ and $(w + \beta z)_I = w_I = \pi_{\pi_I(W)}(x)$, by the definition of $w = L_I^W(x)$, we must have that $\|w\|^2 \le \min_{\beta\in\mathbb{R}} \|w + \beta z\|^2 = \|w\|^2 - \langle z, w\rangle^2$. In particular, $\langle z, w\rangle = 0$, as needed.

Now take any $w \in W$, $w_J \in \pi_J(W^\perp)$, $w_I \in \pi_I(W)^\perp + x$. We wish to show that $w = L_I^W(x)$. Firstly, by the above argument, this system always has a solution, namely $L_I^W(x)$. Secondly, note that $z \in W$ implies that $w_I \in (\pi_I(W)^\perp + x) \cap \pi_I(W)$. By the uniqueness of the orthogonal decomposition, this implies that $z_I = \pi_{\pi_I(W)}(x)$. Thus, z is in the solution set of the $L_I^W(x)$ program. It remains to show that z has minimum norm. Letting $w' \in W \setminus \{w\}$ satisfy $w'_I = \pi_{\pi_I(W)}(x)$, we must show that $\|w'\|^2 > \|w\|^2$. Noting that $z = w' - w \in W \cap \mathbb{R}^n_J \setminus \{0\}$ and recalling that $w \in (W \cap \mathbb{R}^n_J)^\perp$, we have that $\|w + z\|^2 = \|w\|^2 + 2\langle z, w\rangle + \|z\|^2 = \|w\|^2 + \|z\|^2 > \|w\|^2$. Thus, $w = L_I^W(x)$ as needed.

Given the above, we see that the solution to the linear system always exists and is unique (since $L_I^W(x)$ is well-defined). Secondly, since $L_I^W(x)$ is the unique solution to a linear system of equations depending linearly on x, we have that L_I^W is a linear map. □

Proof of Lemma 5.2.15. To prove the statement, it suffices to show that for all $x \in \mathbb{R}^I$ that

$$L_I^W(x) = (\Pi_{\pi_I(W)}(x), \ell_I^W(x)) = (\Pi_{\pi_I(W)}(x), -\ell_J^{W^\perp *}(x)).$$

Letting $z = (\Pi_{\pi_I(W)}(x), -\ell_J^{W^\perp *}(x))$, by Lemma 5.2.14, it suffices to show that

1. $z \in W$,
2. $z_I \in \pi_I(W)^\perp + x$,
3. $z_J \in \pi_J(W^\perp)$.

Property (2) follows directly from $z_I = \Pi_{\pi_I(W)}(x) = x - \Pi_{(\pi_I(W))^\perp}(x)$. To show (1), it suffices to show that $\langle v, z\rangle = 0, \forall v \in W^\perp$. For $v \in W^\perp$, we see that

$$\begin{aligned}
\langle z, v\rangle &= \left\langle \Pi_{\pi_I(W)}(x), v_I\right\rangle - \left\langle \ell_J^{W^\perp *}(x), v_J\right\rangle = \left\langle \Pi_{\pi_I(W)}(x), v_I\right\rangle - \left\langle x, \ell_J^{W^\perp}(v_J)\right\rangle \\
&= \left\langle \Pi_{\pi_I(W)}(x), v_I\right\rangle - \left\langle \Pi_{\pi_I(W)}(x), \ell_J^{W^\perp}(v_J)\right\rangle \quad \left(\text{since } \ell_J^{W^\perp}(v_J) \in \pi_I(W)\right) \\
&= \left\langle \Pi_{\pi_I(W)}(x), v_I - \ell_J^{W^\perp}(v_J)\right\rangle \\
&= \left\langle L_I^W(x), (v_I - \ell_J^{W^\perp}(v_J), 0_J)\right\rangle = \left\langle L_I^W(x), v - L_J^{W^\perp}(v_J)\right\rangle \quad (\text{since } v_J \in \pi_J(W^\perp)) \\
&= 0,
\end{aligned}$$

where the last equality follows since $L_I^W(x) \in W$ and $v - L_J^{W^\perp}(v_j) \in W^\perp$.

To show (3), we must show that $\langle z, w\rangle = 0, \forall w \in W \cap \mathbb{R}^n_J$. For $w \in W \cap \mathbb{R}^n_J$, we first claim that $L_J^{W^\perp}(w_J) = \mathbf{0}_n$. By Lemma 5.2.14, this follows since 0 a solution to the linear system

$$x_J \in \pi_J(W^\perp)^\perp = \pi_J(W \cap \mathbb{R}^n_J) = \pi_J(W \cap \mathbb{R}^n_J) + w_J, \quad x \in W^\perp, \quad x_I \in W_I. \tag{5.70}$$

From the claim, we must have $\ell_J^{W^\perp}(w_J) = (L_J^{W^\perp}(w_J))_I = \mathbf{0}_I$. Therefore,

$$\langle z, w\rangle = -\left\langle \ell_J^{W^\perp *}(x), w_J\right\rangle = -\left\langle x, \ell_J^{W^\perp}(w_J)\right\rangle = -\langle x, \mathbf{0}_I\rangle = 0.$$

Thus, $\ell_I^W = -\ell_J^{W^*}$ as needed. The equality of singular values follows from the fact that adjoints always have the same non-zero singular values. □

Proof of Lemma 5.3.7. By Proposition 5.2.1, we have

$$\begin{aligned}\|x(\mu_1)/x(\mu_0)\|_1 + \|s(\mu_1)/s(\mu_0)\|_1 &= \frac{1}{\mu_0}(\langle x(\mu_1), s(\mu_0)\rangle + \langle x(\mu_0), s(\mu_1)\rangle) \\ &= \frac{1}{\mu_0}(\langle x(\mu_0), s(\mu_0)\rangle + \langle x(\mu_1), s(\mu_1)\rangle) = \left(1 + \frac{\mu_1}{\mu_0}\right)n. \end{aligned} \tag{5.71}$$

Let $B \cup N = [n]$ be the polarization partition. Now assume that $\gamma \geq 1$. We will show then that $\gamma = 1$ and that CP$[\mu_1, \mu_0]$ is linear.

Then for $i \in B$, we see that

$$\frac{x_i(\mu_1)}{x_i(\mu_0)} + \frac{s_i(\mu_1)}{s_i(\mu_0)} = \frac{x_i(\mu_1)}{x_i(\mu_0)} + \frac{\mu_1}{\mu_0}\frac{x_i(\mu_0)}{x_i(\mu_1)} \geq \min_{\beta \geq \gamma}\left(\beta + \frac{1}{\beta}\frac{\mu_1}{\mu_0}\right) = \gamma + \frac{1}{\gamma}\frac{\mu_1}{\mu_0}, \tag{5.72}$$

where the last inequality uses $\gamma \geq 1$, $\mu_1/\mu_0 \leq 1$, and that the function $\beta \to \beta + \frac{1}{\beta}\frac{\mu_1}{\mu_0}$ is increasing for $\beta \geq \sqrt{\frac{\mu_1}{\mu_0}}$.

Swapping the role of x and s, we also get $\frac{x_i(\mu_1)}{x_i(\mu_0)} + \frac{s_i(\mu_1)}{s_i(\mu_0)} \geq \gamma + \frac{\mu_1}{\mu_0}/\gamma, \forall i \in N$. Combining with (5.71), we get that

$$n\left(\gamma + \frac{\mu_0}{\mu_1\gamma}\right) \leq \|x(\mu_1)/x(\mu_0)\|_1 + \|s(\mu_1)/s(\mu_0)\|_1 = n\left(1 + \frac{\mu_0}{\mu_1}\right).$$

For $\gamma \geq 1$, this inequality can only hold if $\gamma = 1$. If $\gamma = 1$, then all the inequalities in Equation (5.72) and their analogs for $i \in N$ must hold at equality. In particular, we get that $x_i(\mu_1) = x_i(\mu_0)$, $i \in B$, and $s_i(\mu_1) = s_i(\mu_0)$, $i \in N$.

We now use this to show that CP$[\mu_1, \mu_0]$ is linear. Define

$$\mu_\alpha := (1-\alpha)\mu_0 + \alpha\mu_1 \quad \text{and} \quad z^{(\alpha)} = (x^{(\alpha)}, s^{(\alpha)}) := (1-\alpha)z(\mu_0) + \alpha z(\mu_1) \quad \text{for } \alpha \in [0,1]\,.$$

To prove linearity, it suffices to show that $z(\mu_\alpha) = z_\alpha$. To see this, note that for $i \in B$, we have $x_i^{(\alpha)} = (1-\alpha)x_i(\mu_0) + \alpha x_i(\mu_1) = (1-\alpha)x_i(\mu_0) + \alpha x_i(\mu_0) = x_i(\mu_0)$ and $\frac{s_i(\mu_1)}{s_i(\mu_0)} = \frac{\mu_1}{\mu_0}$. That implies

$$x_i^{(\alpha)}s_i^{(\alpha)} = x_i(\mu_0)\left((1-\alpha)s_i(\mu_0) + \alpha\frac{\mu_1}{\mu_0}s_i(\mu_0)\right) = x_i(\mu_0)s_i(\mu_0)\left((1-\alpha) + \alpha\frac{\mu_1}{\mu_0}\right) = \mu_\alpha.$$

By a symmetric argument, $x_i^{(\alpha)}s_i^{(\alpha)} = \mu_\alpha, \forall i \in N$. Since $z^{(\alpha)} \in \mathcal{P} \times \mathcal{D}$ and $x^{(\alpha)}s^{(\alpha)} = \mu_\alpha \mathbf{1}$, we must have $z^{(\alpha)} = z(\mu_\alpha)$ as needed.

Now assume that CP$[\mu_1, \mu_0]$ is linear, that is, that $z(\mu_\alpha) = z_\alpha, \forall \alpha \in [0,1]$. We must show that CP$[\mu_1, \mu_0]$ is 1-polarized. For $i \in [n]$, $\alpha \in [0,1]$, we have that

$$\frac{((1-\alpha)x_i^{(0)} + \alpha x_i^{(1)})((1-\alpha)s_i^{(0)} + \alpha s_i^{(1)})}{(1-\alpha)x_i^{(0)}s_i^{(0)} + \alpha x_i^{(1)}s_i^{(1)}} = \frac{x_i^{(\alpha)}s_i^{(\alpha)}}{\mu_\alpha} = \frac{x(\mu_\alpha)s(\mu_\alpha)}{\mu_\alpha} = 1.$$

Thus, by Lemma 5.3.6 applied to $\frac{x_i^{(1)}}{x_i^{(0)}}, \frac{s_i^{(1)}}{s_i^{(0)}}$ and $\gamma = 1$, using that $\frac{x_i^{(1)}}{x_i^{(0)}}\frac{s_i^{(1)}}{s_i^{(0)}} = \frac{\mu_1}{\mu_0}$, we get that

$$\frac{x_i^{(1)}}{x_i^{(0)}} + \frac{s_i^{(1)}}{s_i^{(0)}} \geq \left(1 + \sqrt{\frac{\mu_1}{\mu_0}}\right)^2 - 2\sqrt{\frac{\mu_1}{\mu_0}} = 1 + \frac{\mu_1}{\mu_0}.$$

Since this holds for all $i \in [n]$, by the same argument as above, the inequality above must hold at equality for all $i \in [n]$. In particular, for each $i \in [n]$, we must have either (1) $\frac{x_i^{(1)}}{x_i^{(0)}} = 1$ and $\frac{s_i^{(1)}}{s_i^{(0)}} = \frac{\mu_1}{\mu_0}$ or (2) $\frac{x_i^{(1)}}{x_i^{(0)}} = \frac{\mu_1}{\mu_0}$ and $\frac{s_i^{(1)}}{s_i^{(0)}} = 1$ (note that $\frac{\mu_1}{\mu_0} < 1$ implies that these cases are disjoint). Let $B \subseteq [n]$ denote the indices satisfying case (1) and $N = [n] \setminus B$ be the indices satisfying case (2). It is now direct to verify B, N yield a 1-polarized partition for $\mathrm{CP}[\mu_1, \mu_0]$, as needed. □

Proof of Proposition 5.5.1. We restrict to the proof of the boundedness of $\mathcal{P}_g$, since the proof is analogous for $\mathcal{D}_g$. Let $s^\circ \in \mathcal{D}_{++}$ be a strictly feasible point of the dual, and $x \in \mathcal{P}_g$. By Proposition 5.2.1, we have

$$\left\langle x, s^\star \right\rangle + \left\langle x^\star, s^\circ \right\rangle = \langle x, s^\circ \rangle + \left\langle x^\star, s^\star \right\rangle .$$

Since $\langle x^\star, s^\star \rangle = 0$, we deduce that $\langle x, s^\circ \rangle \leq g + \langle x^\star, s^\circ \rangle$. As $s^\circ > 0$, this implies that $x_i \leq (g + \langle x^\star, s^\circ \rangle)/s_i^\circ$ for all $i \in [n]$. □

6 Curvature Bounds

In this chapter we study the notion of curvature that was introduced by Sonnevend, Stoer and Zhao [SSZ91]. It is a natural notion of complexity of IPM and as such can provide lower and upper bound of reasonable path-following methods. The curvature is a purely geometric notion, independent of implementations of IPM. We will present two main results. The first is an exponential bound on the total curvature, the methods used and the proof resembling closely the proof of the exponential curvature bound in Chapter 5. As a second result we sharpen the total curvature bound provided by [MT08] in terms of the scaling-invariant condition measure $\bar{\chi}^*$. We believe that curvature bounds are of independent interest as they give explicit information about the geometry of a polytope which in IPM is used only implicitly.

The content of this chapter are preliminary results, which are not published yet.

Contents

6.1 Introduction

Recall the LP formulation in primal-dual form

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

where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $\operatorname{rank}(\mathbf{A}) = m \leq n$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$ 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.

If both the primal and dual problems in (LP) are strictly feasible, the central path for (LP) is the curve $\{ (x(\mu), y(\mu), s(\mu) : \mu > 0 \}$ defined by

$$\begin{aligned} x(\mu)_i s(\mu)_i &= \mu, \quad \forall i \in [n] \\ \mathbf{A}x(\mu) &= b,\ x(\mu) > \mathbf{0}, \\ \mathbf{A}^\top y(\mu) + s(\mu) &= c,\ s(\mu) > \mathbf{0}, \end{aligned} \tag{CP}$$

which converges to complementary optimal primal and dual solutions (x^*, y^*, s^*) as $\mu \to 0$, recalling that the optimality gap at time μ is exactly $\langle x(\mu), s(\mu) \rangle = n\mu$. We thus refer to μ as the normalized dualized gap.

6.1.1 Curvature integral

The curvature of the central path introduced by [SSZ91] is defined as follows:

$$\Upsilon(\nu) := \sqrt{\|\nu \dot{x}(\nu)\dot{s}(\nu)\|} \tag{6.1}$$

where $\dot{x}(\mu) = \frac{\mathrm{d}x(\mu)}{\mathrm{d}\mu}$, $\dot{s}(\mu) = \frac{ds(\mu)}{d\mu}$. The *total curvature* of a segment from μ_0 to μ_1, $\mu_1 \leq \mu_0$ of (LP) is defined as

$$\mathcal{I}(\mu_1, \mu_0) := \int_{\mu_1}^{\mu_0} \frac{\Upsilon(\nu)}{\nu}\, \mathrm{d}\nu = \int_{\mu_1}^{\mu_0} \sqrt{\left\| \frac{\dot{x}(\nu)\dot{s}(\nu)}{x(\nu)s(\nu)} \right\|}\, \mathrm{d}\nu\,. \tag{6.2}$$

The *total path curvature* of the central path is defined as $\mathcal{I}(0, \infty)$.

The curvature has following basic properties.

Lemma 6.1.1 ([MT08, Lemmma 2.1]). *The following statements hold:*

1. $\Upsilon(\nu) \leq \sqrt{n/2}$ *for all* $\nu > 0$;
2. *If* $\Upsilon(\nu_0) = 0$ *for some* $\nu_0 > 0$ *then* $\Upsilon(\nu) = 0$ *for every* $\nu > 0$.

6.2 A simplex path bound on the total curvature

In this section we prove Corollary 6.2.3, which is the curvature equivalent of Proposition 5.2.9 (ii). It states that the curvature integral from $\Upsilon(\mu)^2\mu$ to μ is bounded by $O(\sqrt{n})$ if $\Upsilon(\mu) \leq 1$. Similarly, Proposition 5.2.9 (ii) states that a single AS step is able to decrease the duality gap by a term proportional to $\|\Delta x^a \Delta s^a\|/\mu$ if this term is ≤ 1. Note that these two terms coincide if (x, s) is perfectly centered. Let us begin by a few properties of the central path and its derivatives.

$$\begin{aligned} \dot{x}(\mu)_i s(\mu)_i + x(\mu)_i \dot{s}(\mu)_i &= 1, \quad \forall i \in [n] \\ \mathbf{A}\dot{x}(\mu) &= \mathbf{0}, \\ \mathbf{A}^\top \dot{y}(\mu) + \dot{s}(\mu) &= \mathbf{0}, \end{aligned} \tag{CP-gradient}$$

by differentiating by μ in (CP).

Let $W(\mu) \coloneqq x(\mu)^{-1}W$. The equations in (CP-gradient) can equivalently be written as

$$\begin{aligned} \frac{\dot{x}(\mu)}{x(\mu)} + \frac{\dot{s}(\mu)}{s(\mu)} &= \frac{1}{\mu}\mathbf{1} \\ \frac{\dot{x}(\mu)}{x(\mu)} &\in W(\mu), \\ \frac{\dot{s}(\mu)}{s(\mu)} &\in W(\mu)^{\perp}, \end{aligned} \tag{6.3}$$

Given μ and subset $S \subseteq [n]$ we write

$$\hat{x}(\mu, S) \coloneqq \operatorname*{arg\,min}_{x \in \pi_S(W)+d_S} \left\| \frac{x}{x(\mu)_S} \right\|, \quad \text{and} \quad \hat{s}(\mu, S) \coloneqq \operatorname*{arg\,min}_{s \in \pi_S(W^{\perp})+c_S} \left\| \frac{s}{s(\mu)_S} \right\| \tag{6.4}$$

Note that $|\hat{x}(\mu, S)| \leq \sqrt{|S|}x(\mu)_S$, where $|\cdot|$ is taken coordinate-wise.

We define for $S \subseteq [n]$

$$Rx(\mu, S) = \frac{\hat{x}(\mu, S)}{x(\mu)_S}, \quad \text{and} \quad Rs(\mu, S) = \frac{\hat{s}(\mu, S)}{s(\mu)_S} \tag{6.5}$$

and in slight abuse of notation $Rx(\mu) = Rx(\mu, [n])$. This corresponds to the notation Rx used in the previous chapters.

Note that $Rx(\mu) \in W(\mu)^{\perp}$ and $Rs(\mu) \in W(\mu)$. In fact,

$$Rx(\mu) = \Pi_{W(\mu)^{\perp}}(\mathbf{1}), \quad Rs(\mu) = \Pi_{W(\mu)}(\mathbf{1}) \tag{6.6}$$

So, in particular $Rx(\mu) + Rs(\mu) = \mathbf{1}$ and we can match

$$Rx(\mu) = \mu\frac{\dot{s}(\mu)}{s(\mu)}, \quad Rs(\mu) = \mu\frac{\dot{x}(\mu)}{x(\mu)} \tag{6.7}$$

Note that we further have

$$\Upsilon(\mu)^2 = \|\mu\dot{x}(\mu)\dot{s}(\mu)\| = \|Rx(\mu)Rs(\mu)\| \leq \sqrt{n}\sqrt{\|Rx(\mu)_N\|^2 + \|Rs(\mu)_B\|^2} \tag{6.8}$$

for any partition $B \cup N = [n]$.

Lemma 6.2.1 (Curvature bound). *If* $16\Upsilon(\mu_0)^2 \leq 1$, *then for all* $\mu \in [16\Upsilon(\mu_0)^2\mu_0, \mu_0]$ *we have* $\Upsilon(\mu) \leq \sqrt{2}\sqrt{\frac{\mu_0}{\mu}}\Upsilon(\mu_0)$.

Proof. For a matrix $\mathbf{M} \in \mathbb{R}^{n\times m}$, weights $w \in \mathbb{R}^n_{>0}$ and direction $h \in \mathbb{R}^n$, let $\Pi(\mathbf{WA})$ be the projection onto $\operatorname{im}(\mathbf{WA})$. Here $\mathbf{W} \coloneqq \operatorname{diag}(w)$. Then

$$\frac{\mathrm{d}\Pi(\mathbf{WA})}{\mathrm{d}w}[h] = \mathbf{H}\mathbf{W}^{-1}\Pi(\mathbf{WA}) + \Pi(\mathbf{WA})\mathbf{H}\mathbf{W}^{-1} - 2\Pi(\mathbf{WA})\mathbf{H}\mathbf{W}^{-1}\Pi(\mathbf{WA}). \tag{6.9}$$

Applied to Rx and using the notation $\mathbf{X}(\mu) := \mathrm{diag}(x(\mu))$, $\dot{\mathbf{X}}(\mu) := \mathrm{diag}(\dot{x}(\mu))$ we obtain

$$\begin{aligned}
\dot{Rx}(\mu) &= \frac{\mathrm{d}Rx(\mu)}{\mathrm{d}\mu} = \frac{\mathrm{d}\Pi_{W(\mu)^\perp}}{\mathrm{d}(x(\mu)^{-1})}\left[\frac{\mathrm{d}x(\mu)^{-1}}{\mathrm{d}\mu}\right](\mathbf{1}) \\
&= -\frac{\mathrm{d}\Pi_{W(\mu)^\perp}}{\mathrm{d}(x(\mu)^{-1})}\left[\frac{\dot{x}(\mu)}{x(\mu)^2}\right](\mathbf{1}) \\
&= \left[\frac{\dot{\mathbf{X}}(\mu)}{\mathbf{X}(\mu)^2}\mathbf{X}(\mu)\Pi_{W(\mu)^\perp} + \Pi_{W(\mu)^\perp}\frac{\dot{\mathbf{X}}(\mu)}{\mathbf{X}(\mu)^2}\mathbf{X}(\mu) - 2\Pi_{W(\mu)^\perp}\frac{\dot{\mathbf{X}}(\mu)}{\mathbf{X}(\mu)^2}\mathbf{X}(\mu)\Pi_{W(\mu)^\perp}\right](\mathbf{1}) \\
&= \frac{\dot{\mathbf{X}}(\mu)}{\mathbf{X}(\mu)}Rx(\mu) + 0 - 2\mu^{-1}\Pi_{W(\mu)^\perp}\big(Rx(\mu)Rs(\mu)\big) \\
&= \mu^{-1}(\mathbf{I} - 2\Pi_{W(\mu)^\perp})[Rx(\mu)Rs(\mu)]\,.
\end{aligned} \tag{6.10}$$

Analogously one can show that

$$\dot{Rs}(\mu) = \mu^{-1}(\mathbf{I} - 2\Pi_{W(\mu)})[Rx(\mu)Rs(\mu)]\,. \tag{6.11}$$

Let us further define $\varrho(\mu) := Rx(\mu)Rs(\mu)$. Noting that $\mathbf{I} - 2\Pi_{W(\mu)^\perp} = -(\mathbf{I} - 2\Pi_{W(\mu)})$ we get

$$\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}\mu}\|\varrho(\mu)\|_2^2 &= 2\left\langle\varrho(\mu), \frac{\mathrm{d}}{\mathrm{d}\mu}\varrho(\mu)\right\rangle \\
&= 2\left\langle\varrho(\mu), \dot{Rx}(\mu)Rs(\mu) + Rx(\mu)\dot{Rs}(\mu)\right\rangle \\
&= \frac{2}{\mu}\left\langle\varrho(\mu), (\mathbf{I} - 2\Pi_{W(\mu)^\perp})[\varrho(\mu)]Rs(\mu) + (\mathbf{I} - 2\Pi_{W(\mu)})[\varrho(\mu)]Rx(\mu)\right\rangle \\
&= \frac{2}{\mu}\left\langle\varrho(\mu), (\mathbf{I} - 2\Pi_{W(\mu)})[\varrho(\mu)](Rx(\mu) - Rs(\mu)\right\rangle \\
&\geq -\frac{2}{\mu}\|\varrho(\mu)\|_2^2\|Rx(\mu) - Rs(\mu)\|_\infty\,.
\end{aligned}$$

Note that

$$\begin{aligned}
\|Rx(\mu) - Rs(\mu)\|_\infty^2 &= \|Rx(\mu)^2 + Rs(\mu)^2 - 2\varrho(\mu)\|_\infty^2 \\
&= \|Rx(\mu)(\mathbf{1} - Rs(\mu)) + Rs(\mu)(\mathbf{1} - Rx(\mu)) - 2\varrho(\mu)\|_\infty^2 \\
&= \|\mathbf{1} - 4\varrho(\mu)\|_\infty^2 \\
&\leq (1 + 4\|\varrho(\mu)\|_2)^2\,.
\end{aligned} \tag{6.12}$$

Hence, with $\Upsilon(\mu) = \sqrt{\|\varrho(\mu)\|_2}$ we have

$$\frac{\mathrm{d}}{\mathrm{d}\mu}\Upsilon(\mu)^4 \geq -\frac{2}{\mu}\Upsilon(\mu)^4\big(1 + 4\Upsilon(\mu)^2\big)\,, \tag{6.13}$$

and so

$$\dot{\Upsilon}(\mu) = \frac{1}{4\Upsilon(\mu)^3}\frac{\mathrm{d}}{\mathrm{d}\mu}\Upsilon(\mu)^4 \geq -\frac{1}{2\mu}\Upsilon(\mu)\big(1 + 4\Upsilon(\mu)^2\big)\,. \tag{6.14}$$

To get good bounds on the evolvement of Υ let us consider following more general differential equation. For $a \in \mathbb{R}$

$$\dot{\gamma}_a(\mu) = -\frac{1}{2\mu}\gamma_a(\mu)(1 + a\gamma_a(\mu)^2) \tag{6.15}$$

with condition $\gamma_a(\mu_0) = \Upsilon(\mu_0)$ has the closed form solution

$$\gamma_a(\mu) = \left(\left(\Upsilon(\mu_0)^{-2} + a\right)\frac{\mu}{\mu_0} - a\right)^{-1/2}. \tag{6.16}$$

For $a = 4$ we have $\Upsilon(\mu_0)^2 \leq 1/(4a)$. Then for $\mu \geq 4a\Upsilon(\mu_0)^2\mu_0$ we have that

$$\left(1 - \frac{\mu}{\mu_0}\right)a \leq \frac{1}{2}\Upsilon(\mu_0)^{-2}\frac{\mu}{\mu_0}, \tag{6.17}$$

and so

$$\begin{aligned}
\gamma_a(\mu) &= \left(\left(\Upsilon(\mu_0)^{-2} + a\right)\frac{\mu}{\mu_0} - a\right)^{-1/2} \\
&= \left(\Upsilon(\mu_0)^{-2}\frac{\mu}{\mu_0} + \left(\frac{\mu}{\mu_0} - 1\right)a\right)^{-1/2} \\
&\leq \left(\Upsilon(\mu_0)^{-2}\frac{\mu}{\mu_0} - \frac{1}{2}\Upsilon(\mu_0)^{-2}\frac{\mu}{\mu_0}\right)^{-1/2} \\
&= \sqrt{2}\Upsilon(\mu_0)\sqrt{\frac{\mu_0}{\mu}} \\
&= \sqrt{2}\gamma_0(\mu),
\end{aligned} \tag{6.18}$$

which proves the lemma. □

Lemma 6.2.2. *For all $\mu_0 \in \mathbb{R}_+$ with $16\Upsilon(\mu_0)^2 \leq 1$ we have*

$$\mathcal{I}\left(16\Upsilon(\mu_0)^2\mu_0, \mu_0\right) = O(1). \tag{6.19}$$

Proof. Using Lemma 6.2.1 we get

$$\begin{aligned}
\mathcal{I}\left(16\Upsilon(\mu_0)^2\mu_0, \mu_0\right) &= \int_{\Upsilon(4\cdot 4\mu_0)^2\mu_0}^{\mu_0} \frac{\Upsilon(\mu)}{\mu} d\mu \\
&\leq \int_{4\cdot 4\Upsilon(\mu_0)^2\mu_0}^{\mu_0} \frac{\gamma_a(\mu)}{\mu} d\mu \\
&\leq \sqrt{2}\Upsilon(\mu_0)\int_{4\cdot 4\Upsilon(\mu_0)^2\mu_0}^{\mu_0} \frac{\mu_0^{1/2}}{\mu^{3/2}} d\mu \\
&= \frac{1}{2}\sqrt{2}\Upsilon(\mu_0)\mu_0^{1/2}\left((4\cdot 4\Upsilon(\mu_0)^2\mu_0)^{-1/2} - \mu_0^{-1/2}\right) \\
&= 2^{-1/2}\cdot(2^{-2} - \Upsilon(\mu_0)) \\
&= O(1),
\end{aligned}$$

which proves the lemma. □

From here it is straightforward to prove the main result of this section.

Corollary 6.2.3. *For all $\mu \in \mathbb{R}_+$ with $\Upsilon(\mu) \leq 1$ we have $\mathcal{I}\left(\Upsilon(\mu)^2\mu, \mu\right) = O(\sqrt{n})$.*

Proof. Using Lemma 6.2.2 and the bound $\Upsilon(\mu) \leq \sqrt{n}$ we have

$$\begin{aligned}
\mathcal{I}\left(\Upsilon(\mu_0)^2\mu_0, \mu_0\right) &= \int_{\Upsilon(\mu_0)^2\mu_0}^{\mu_0} \frac{\Upsilon(\mu)}{\mu} d\mu \\
&= \int_{\Upsilon(\mu_0)^2\mu_0}^{4\cdot 4\Upsilon(\mu_0)^2\mu_0} \frac{\Upsilon(\mu)}{\mu} d\mu + \int_{4\cdot 4\Upsilon(\mu_0)^2\mu_0}^{\mu_0} \frac{\Upsilon(\mu)}{\mu} d\mu \\
&\leq \sqrt{n}\int_{\Upsilon(\mu_0)^2\mu_0}^{4\cdot 4\Upsilon(\mu_0)^2\mu_0} \frac{1}{\mu} d\mu + O(1) \\
&= O(\sqrt{n})
\end{aligned}$$

□

6.2.1 Curvature on polarized segments

The main result of this section is a total curvature analogue of Theorem 5.1.2.

Theorem 6.2.4. *We have that*

$$\mathcal{I}(0,\infty) = O\left(n^{1.5}\log n\ \min\left\{\sum_{i=1}^{n} S_{\mathcal{P}}(-s^\star, e^i) + S_{\mathcal{D}}(-x^\star, e^i), \mathcal{V}_{\mathcal{P}} + \mathcal{V}_{\mathcal{D}}\right\}\right),$$

where $\mathcal{V}_{\mathcal{P}}$, $\mathcal{V}_{\mathcal{P}}$, $S_{\mathcal{P}}$ and $S_{\mathcal{D}}$ are as in Chapter 5.

The proof is similarly to the proof strategy of the previous chapter based on showing that polarized segments have small curvature. To this end, consider a γ-polarized segment $\mathrm{CP}[\mu_1, \mu_0]$ of the central path with polarizing partition $B \cup N = [n]$. We are going to prove the following theorem, of which Theorem 6.2.4 is a trivial consequence.

Theorem 6.2.5. *For a γ-polarized segment $\mathrm{CP}[\mu_1, \mu_0]$ we have that*

$$\mathcal{I}(\mu_1, \mu_0) = \widetilde{O}\left(n^{1.5}\mathrm{poly}(\log \gamma^{-1})\right). \tag{6.20}$$

Let us first introduce some basic notation and show basic results.

The important property we will use in the following is that for every $\xi \in \mathbb{R}^S$, $S \subseteq [n]$ such that $\xi \in \pi_S(W)$ we have

$$\left\langle \frac{\xi}{x(\mu)_S}, \frac{\hat{x}(\mu, S)}{x(\mu)_S} \right\rangle = 0. \tag{6.21}$$

Let us denote $\ell_\mu := \ell_N^{x(\mu)^{-1}W}$, where the latter notation is the same as in the previous chapter defined in Definition 5.2.13. We can show following bound on the projected residuals in terms of subspace-restriction of ℓ_μ.

Lemma 6.2.6 (SLLS-proximity)**.** *For any subspace $V \subseteq \pi_N(W(\mu))$ we have*

$$\|\Pi_V(Rx(\mu)_N)\| = \left\|\Pi_V\left(\frac{\hat{x}(\mu,[n])_N}{x(\mu)_N}\right)\right\| \leq \sqrt{n}\|\ell_\mu|_V\|. \tag{6.22}$$

Proof. We have

$$\begin{aligned}
\left\|\Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right)\right\|^2 &= \left\langle \Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right), \Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right)\right\rangle \\
&= \left\langle \Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right), \frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right\rangle \\
&= -\left\langle L(\mu,N)\left[\Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right)\right]_B, \frac{\hat{x}([n],\mu)_B}{x(\mu)_B}\right\rangle \\
&\leq \ell(\mu,N,V)\left\|\Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right)\right\|\left\|\frac{\hat{x}([n],\mu)_B}{x(\mu)_B}\right\| \\
&\leq \sqrt{n}\ell(\mu,N,V)\left\|\Pi_V\left(\frac{\hat{x}([n],\mu)_N - \hat{x}(N,\mu)}{x(\mu)_N}\right)\right\|
\end{aligned} \tag{6.23}$$

Division on both sides and noting that $\frac{\hat{x}(\mu,N)}{x(\mu)} \perp \pi_N(W(\mu)) \supseteq V$ gives the result. □

Let us now consider the usual orthogonal decompositions $V(\mu) \perp \overline{V}(\mu)$, $V(\mu) \oplus \overline{V}(\mu) = \pi_N(W(\mu))$ and $U(\mu) \perp \overline{U}(\mu)$, $U(\mu) \oplus \overline{U}(\mu) = \pi_B(W(\mu)^\perp)$ into *cheap* primal/dual and *expensive* primal/dual

subspaces. We choose $V(\mu)$ as the subspace spanned by the kernel of ℓ_μ as well as the smallest k non-zero singular vectors of ℓ_μ for some $k \in [n]$. Analogously choose $U(\mu)$. Recall the duality of these non-zero singular vectors of ℓ_μ and $\ell_\mu^\perp$.

We proceed by upper bounding the curvature $\Upsilon(\mu)$ by finding bound on $Rx(\mu)_N$ and $Rs(\mu)_B$ in light of (6.8).

The curvature at a parameter μ can be upper bounded by

$$\frac{\Upsilon(\mu)^2}{\sqrt{n}} = \frac{1}{\sqrt{n}}\|Rx(\mu)Rs(\mu)\| \leq \|Rx(\mu)_N\| + \|Rs(\mu)_B\|\,. \tag{6.24}$$

We can further upper bound

$$\begin{aligned}
\|Rx(\mu)_N\| &\leq \left\|\frac{x(\mu_1)_N}{x(\mu)_N}\right\| + \left\|Rx(\mu)_N - \frac{x(\mu_1)_N}{x(\mu)_N}\right\| \\
&= \left\|\frac{x(\mu_1)_N}{x(\mu)_N}\right\| + \left\|\Pi_V\left(Rx(\mu)_N - \frac{x(\mu_1)_N}{x(\mu)_N}\right)\right\| + \left\|\Pi_{\overline{V}}\left(Rx(\mu)_N - \frac{x(\mu_1)_N}{x(\mu)_N}\right)\right\| \\
&\leq 2\left\|\frac{x(\mu_1)_N}{x(\mu)_N}\right\| + \|\Pi_V(Rx(\mu)_N)\| + \left\|\Pi_{\overline{V}(\mu)}(Rx(\mu)_N - \mathbf{1}_N)\right\| + \left\|\Pi_{\overline{V}(\mu)}\left(\mathbf{1}_N - \frac{x(\mu_1)_N}{x(\mu)_N}\right)\right\|.
\end{aligned} \tag{6.25}$$

Note that the last term corresponds to the empirical gradient from the previous chapter. In abuse of notation we write $\sigma_i(\mu) \coloneqq \sigma_i((x(\mu), s(\mu)))$. Let us now further bound the terms on the right. With Lemma 6.2.6 we have that

$$\|\Pi_V(Rx(\mu)_N)\| = \left\|\Pi_V\left(\frac{\hat{x}(\mu, [n])_N}{x(\mu)_N}\right)\right\| \leq \sqrt{n}\ell(\mu, N, V) = \sqrt{n}\sigma_k(\mu)\,. \tag{6.26}$$

Let $w \in W(\mu)$ be any vector. Note that

$$\left(\Pi_{V(\mu)}(w_N), \ell_\mu\left(\Pi_{V(\mu)}(w_N)\right)\right) \in W(\mu)$$

and so

$$\left(\Pi_{\overline{V}(\mu)}(w_N), w_B - \ell_\mu\left(\Pi_{V(\mu)}(w_N)\right)\right) = \left(w_N - \Pi_{V(\mu)}(w_N), w_B - \ell_\mu\left(\Pi_{V(\mu)}(w_N)\right)\right) \in W(\mu)\,. \tag{6.27}$$

Therefore, in particular

$$\begin{aligned}
\sigma(\mu)_{k+1}\|\Pi_{\overline{V}(\mu)}(w_N)\| &\leq \left\|\ell_\mu(\Pi_{\overline{V}(\mu)}(w_N))\right\| \leq \left\|w_B - \ell_\mu\left(\Pi_{V(\mu)}(w_N)\right)\right\| \\
&\leq \|w_B\| + \sigma(\mu)_k\|\Pi_{V(\mu)}(w_N)\| \\
&\leq \|w_B\| + \sigma(\mu)_k\|w_N\|\,.
\end{aligned} \tag{6.28}$$

Applying the above with $w = Rx(\mu) - \mathbf{1} \in W(\mu)$ and noting that $\|Rx(\mu) - \mathbf{1} \in W(\mu)\| \leq \sqrt{n}$ we get

$$\sigma(\mu)_{k+1}\|\Pi_{\overline{V}(\mu)}(Rx(\mu)_N - \mathbf{1}_N)\| \leq \sqrt{n}(1 + \sigma(\mu)_k). \tag{6.29}$$

Applying it with $w = x(\mu_1)/x(\mu) - \mathbf{1}$ and noting that with Lemma 5.2.4 we have

$$\left\|\frac{x(\mu_1)}{x(\mu)} - \mathbf{1}\right\| \leq \left\|\frac{x(\mu_1)}{x(\mu)}\right\|_1 + \sqrt{n} \leq 2n + \sqrt{n} \leq 3n\,, \tag{6.30}$$

and so

$$\sigma(\mu)_{k+1}\|\Pi_{\overline{V}(\mu)}(x(\mu_1)_N - \mathbf{1}_N)\| \leq 3n(1+\sigma(\mu)_k)\,. \tag{6.31}$$

Plugging (6.26), (6.29) and (6.31) into (6.25), we get

$$\begin{aligned}\|Rx(\mu)_N\| &\leq 2\left\|\frac{x(\mu_1)_N}{x(\mu)_N}\right\| + \sqrt{n}\sigma(\mu)_k + \sqrt{n}(1+\sigma(\mu)_k)\sigma(\mu)_{k+1}^{-1} + 3n(1+\sigma(\mu)_k)\sigma(\mu)_{k+1}^{-1} \\ &\leq 2\left\|\frac{x(\mu_1)_N}{x(\mu)_N}\right\| + \sqrt{n}\sigma(\mu)_k + 4n(1+\sigma(\mu)_k)\sigma(\mu)_{k+1}^{-1}\,.\end{aligned} \tag{6.32}$$

With analogous computation on the dual side we get

$$\|Rs(\mu)_B\| \leq 2\left\|\frac{s(\mu_1)_B}{s(\mu)_B}\right\| + \sqrt{n}\sigma(\mu)_k + 4n(1+\sigma(\mu)_k)\sigma(\mu)_{k+1}^{-1}\,. \tag{6.33}$$

If we combine these two bounds with the stability of the singular values from the previous chapter, we are able to bound the curvature until a new singular value becomes small. Let $\widetilde{\mu} \coloneqq \max\{p\mu_1, p\sigma(\mu_0)_{k+1}^{-1}\mu_0\}$, where $p = \mathrm{poly}(\gamma^{-1}, n)$ that we are going to choose later. For every $\mu \in [\widetilde{\mu}, \mu_0]$ we have by Corollary 5.3.3 that

$$\begin{aligned} x(\mu)_N &\geq \frac{\mu}{n\mu_0}x(\mu_0)_N, \quad s(\mu)_B \geq \frac{\mu}{n\mu_0}s(\mu_0)_B \quad \text{and} \\ x(\mu_1)_N &\leq \frac{\mu_1}{\gamma\mu_0}x(\mu_0)_N, \quad s(\mu_1)_B \leq \frac{\mu_1}{\gamma\mu_0}s(\mu_0)_B \end{aligned} \tag{6.34}$$

Further, we have by Lemma 5.4.7 that

$$\sigma(\mu)_k \leq \sigma(\mu_0)_k\frac{4n^2\mu}{\gamma^2\mu_0}, \quad \sigma(\mu)_{k+1} \geq \sigma(\mu_0)_{k+1}\frac{\gamma^2\mu}{4n^2\mu_0}\,. \tag{6.35}$$

With these inequalities, we are ready to prove the main theorem of the section.

Proof of Theorem 6.2.5. Let us now choose

$$k \coloneqq \max\left\{\, i : \sigma(\mu_0)_i \leq \frac{\gamma^2}{4n^2} \,\right\}. \tag{6.36}$$

By the above, we therefore have $\sigma(\mu)_k \leq 1$ for all $\mu \in [\widetilde{\mu}, \mu_0]$. We can now bound

$$\begin{aligned}
\mathcal{I}(\widetilde{\mu}, \mu_0) &= \int_{\widetilde{\mu}}^{\mu_0} \frac{\Upsilon(\nu)}{\nu} \mathrm{d}\nu \overset{(6.2.1)}{\leq} n^{1/4} \int_{\widetilde{\mu}}^{\mu_0} \frac{\sqrt{\|Rx(\nu)_N\| + \|Rs(\nu)_B\|}}{\nu} \mathrm{d}\nu \\
&\leq 2 \int_{p\mu_1}^{\mu_0} \frac{1}{\nu} \sqrt{\left\| \frac{s(\mu_1)_B}{s(\nu)_B} \right\| + \left\| \frac{x(\mu_1)_N}{x(\nu)_N} \right\|} \mathrm{d}\nu + 2n^{1/4} \int_{p\mu_1}^{\mu_0} \frac{\sqrt{\sigma(\nu)_k}}{\nu} \mathrm{d}\nu \\
&\quad + 8n^{1/2} \int_{p\sigma(\mu_0)_{k+1}^{-1}\mu_0}^{\mu_0} \frac{1}{\sqrt{\sigma(\nu)_{k+1}}\nu} \mathrm{d}\nu \\
&\leq 4n^{1/4} \int_{p\mu_1}^{\mu_0} \frac{1}{\nu} \sqrt{\frac{n\mu_1}{\gamma\nu}} \mathrm{d}\nu + 2n^{1/4} \int_{p\mu_1}^{\mu_0} \frac{1}{\nu} \sqrt{\sigma(\mu_0)_k \frac{4n^2\nu}{\gamma^2\mu_0}} \mathrm{d}\nu \\
&\quad + 8n^{1/2} \int_{p\sigma(\mu_0)_{k+1}^{-1}\mu_0}^{\mu_0} \frac{1}{\nu} \sqrt{\frac{4n^2\mu_0}{\gamma^2\nu\sigma(\mu_0)_{k+1}}} \mathrm{d}\nu \\
&= 8n^{3/4}\gamma^{-1/2}\sqrt{\mu_1}[-\nu^{-1/2}]_{p\mu_1}^{\mu_0} + 8n^{5/4}\sqrt{\sigma(\mu_0)_k}\gamma^{-1}\mu_0^{-1/2}[\sqrt{\nu}]_{p\mu_1}^{\mu_0} \\
&\quad + 32n^{3/2}\sigma(\mu_0)_{k+1}^{-1/2}\gamma^{-1}\mu_0^{1/2}[-\nu^{-1/2}]_{p\sigma(\mu_0)_{k+1}^{-1}\mu_0}^{\mu_0} \\
&\leq 8n^{3/4}\gamma^{-1/2}p^{-1/2} + 8n^{5/4}\sqrt{\sigma(\mu_0)_k}\gamma^{-1} + 32n^{3/2}\gamma^{-1}p^{-1/2} \\
&= O\left(n^{3/4}\gamma^{-1/2}p^{-1/2} + n^{5/4}\sqrt{\sigma(\mu_0)_k}\gamma^{-1} + n^{3/2}p^{-1/2}\sqrt{\sigma(\mu_0)_k}\gamma^{-1}\right).
\end{aligned}$$

This bound can be used to further bound

$$\begin{aligned}
\mathcal{I}(p^{-1}\widetilde{\mu}, \mu_0) &= \mathcal{I}(p^{-1}\widetilde{\mu}, \widetilde{\mu}) + \mathcal{I}(\widetilde{\mu}, \mu_0) \\
&\leq \sqrt{n}\log(p) + O\left(n^{3/4}\gamma^{-1/2}p^{-1/2} + n^{5/4}\sqrt{\sigma(\mu_0)_k}\gamma^{-1} + n^{3/2}p^{-1/2}\sqrt{\sigma(\mu_0)_k}\gamma^{-1}\right),
\end{aligned} \tag{6.37}$$

where the inequality used Lemma 6.1.1. Choosing $p \geq n^{3/2}\gamma$ we therefore get

$$\mathcal{I}(p^{-1}\widetilde{\mu}, \mu_0) = O\left(\sqrt{n}\log(\gamma^{-1} + n) + 1\right) = O\left(\sqrt{n}\log(\gamma^{-1} + n)\right). \tag{6.38}$$

We can now make a case distinction. If $\mu_1 \geq \sigma(\mu_0)_{k+1}^{-1}\mu_0$, then $\mu_1 = p^{-1}\widetilde{\mu}$ and so

$$\mathcal{I}(\mu_1, \mu_0) = \mathcal{I}(p^{-1}, \widetilde{\mu}, \mu_0) = O(\sqrt{n}\log(\gamma^{-1} + n)). \tag{6.39}$$

Otherwise, $\sigma(\mu_0)_{k+1}^{-1}\mu_0 = p^{-1}\mu_0$ and then

$$\mathcal{I}(\sigma(\mu_0)_{k+1}^{-1}\mu_0, \mu_0) = \mathcal{I}\left(p^{-1}, \widetilde{\mu}, \mu_0\right) = O\left(\sqrt{n}\log(\gamma^{-1} + n)\right). \tag{6.40}$$

But now note that again the stability of singular values Lemma 5.4.7, we have for $q := \frac{\gamma^2}{16n^4}$.

$$\sigma\left(q\sigma(\mu_0)_{k+1}^{-1}\mu_0\right)_{k+1} \leq \sigma(\mu_0)_{k+1}\frac{4n^2 q\sigma(\mu_0)_{k+1}^{-1}\mu_0}{\mu_0} = q4n^2 = \frac{\gamma^2}{4n^2}, \tag{6.41}$$

and so when we choose the next k as in (6.36), we would increase the index by at least one. Such an increase can only happen n times. It remains to prove that the curvature corresponding to $q\sigma(\mu_0)_{k+1}^{-1}\mu_0$ is small. But this can be done with the standard bound

$$\begin{aligned}\mathcal{I}(q\sigma(\mu_0)_{k+1}^{-1}\mu_0, \mu_0) &= \mathcal{I}(q\sigma(\mu_0)_{k+1}^{-1}\mu_0, \sigma(\mu_0)_{k+1}^{-1}\mu_0) + \mathcal{I}(\sigma(\mu_0)_{k+1}^{-1}\mu_0), \mu_0) \\ &\leq \sqrt{n}\log(q) + O(\sqrt{n}\log(\gamma^{-1}+n)) \\ &= O\left(\sqrt{n}\log(\gamma^{-1}+n)\right),\end{aligned} \tag{6.42}$$

which proves the theorem. □

6.3 Bounding the curvature with circuit imbalances

The relationship between the total curvature and the iteration complexity of the Mizuno-Todd-Ye (MTY) algorithm has been extensively studied in [MT08]. One of their main results is following theorem.

Theorem 6.3.1 ([MT08]). *Let $\beta \in (0, 1/2]$. For given $w^0 \in \mathcal{N}(\beta)$ and $0 < \nu_f < \mu(w^0)$, denote by $\#(w^0, \nu_f, \beta)$ the number of iterations of the MTY Predictor-Corrector algorithm with $\beta \in (0, 1/2)]$ needed to reduce the duality gap from $\nu_i := \mu(w^0)$ to ν_f. Then,*

$$\lim_{\beta\to 0} \frac{\mathcal{I}(\nu_f, \nu_i)/\sqrt{\beta}}{\#(w^0, \nu_f, \beta)} = 1\,. \tag{6.43}$$

Another central theorem in [MT08] is that the curvature can be bounded in terms κ (and $\bar{\chi}$). As the curvature is scaling-invariant this yields the same bound for κ^* (and $\bar{\chi}^*$). Their main result is

Theorem 6.3.2 ([MT08]). *For LP, the total curvature of the standard central path $\mathcal{I}(0,\infty)$ is bounded by*

$$\mathcal{I}(0,\infty) = O(n^{3.5}\log(\bar{\chi}^*_{\mathbf{A}} + n))\,. \tag{6.44}$$

The proof of their result relies on two ingredients. First, to show that the worst case analysis for affine scaling steps recovers a total curvature bound of $O(\sqrt{n}\log(\mu_0/\mu_1))$ to reduce the centrality parameter from μ_0 to μ_1. The other ingredient is showing that the total curvature along a segment traversed within a single LLS step is bounded by a constant. We recover this result for the special case of only two layers.

The VY algorithm shows that $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}} + n))$ affine scaling steps with worst-case-analysis and $O(n^2)$ LLS steps suffice to find an optimal solution to LP via path-following. For the affine scaling steps we note that it requires $O(\beta^{-1}\sqrt{n})$ AS iterations to decrease the duality gap by a factor of 2. And so with Lemma 6.1.1

$$\mathcal{I}(\mu/2, \mu) = \int_{\mu/2}^{\mu} \frac{\Upsilon(\nu)}{\nu}\,\mathrm{d}\nu \leq \sqrt{n}\int_{\mu/2}^{\mu}\frac{1}{\nu}\,\mathrm{d}\nu = \sqrt{n}\ln(2) = O(\sqrt{n})\,. \tag{6.45}$$

In particular, for $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}} + n))$ AS steps we get that the total curvature of the segments traversed in these steps is $O(n^{3.5}\log(\bar{\chi}_{\mathbf{A}} + n))$.

To bound the curvature of LLS steps we require the theory from the previous section. Recall the acceleration on cheap subspaces in the proof of Theorem 6.2.5. As for LLS steps with partition $B \cup N$ we have that $V(\mu) = \pi_N(W)$, the proof gives us immediately that the curvature for an LLS step is bounded by $O(1)$.

In Chapter 4 we have seen an improvement over the [VY96] bound and presented an algorithm with $O(n^2)$ LLS steps but only $\tilde{O}(n^{2.5} \log \bar{\chi}^*_{\mathbf{A}})$ many AS steps. With the reasoning in this chapter this immediately yields the following theorem.

Theorem 6.3.3. *We have that*

$$\mathcal{I}(0, \infty) = \widetilde{O}(n^{2.5} \log \bar{\chi}^*_{\mathbf{A}}) \,. \tag{6.46}$$

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

In breakthrough work, Tardos [Tar86] 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 $\min\langle c, x\rangle$, $\mathbf{A}x = b$, $x \geq \mathbf{0}$, $\mathbf{A} \in \mathbb{Z}^{m\times n}$, to solving $O(nm)$ LPs in $\mathbf{A}$ having small integer coefficient objectives and right-hand sides using *any exact* LP algorithm. This gives rise to an LP algorithm in time $\text{poly}(n, m \log \Delta_{\mathbf{A}})$, where $\Delta_{\mathbf{A}}$ is the largest subdeterminant of $\mathbf{A}$. 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 $\text{poly}(n, m, \log \bar{\chi}_{\mathbf{A}})$, depending on Stewart's $\bar{\chi}_{\mathbf{A}}$, 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 $\mathbf{A}$ is inverse polynomial in n and $\log \bar{\chi}_{\mathbf{A}}$. Plugging in the recent algorithm of van den Brand [Bra20], our method computes an optimal primal and dual solution using $\widetilde{O}(mn^{\omega+1} \log(\bar{\chi}_{\mathbf{A}}))$ arithmetic operations, outperforming the specialized interior point method of Vavasis and Ye and its recent improvement by Dadush et al [DHNV20]. By applying the preprocessing algorithm of the latter paper, the dependence can also be reduced from $\bar{\chi}_{\mathbf{A}}$ to $\bar{\chi}^*_{\mathbf{A}}$, the minimum value of $\bar{\chi}_{\mathbf{AD}}$ attainable via column rescalings. Our framework is applicable to achieve the $\text{poly}(n, m, \log \bar{\chi}^*_{\mathbf{A}})$ bound using essentially any weakly polynomial LP algorithm, such as the ellipsoid method.

At a technical level, our framework combines 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.

This chapter is based on joint work with Daniel Dadush and László A. Végh [DNV20].

Contents

7.1 Introduction

In this chapter, we consider the task of computing exact primal and dual solutions for linear programs (LP) in standard form System 1.1.

Here, $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\mathrm{rk}(\mathbf{A}) = m \leq n$, $d \in \mathbb{R}^n$, $c \in \mathbb{R}^n$ are given in the input, and $x, s \in \mathbb{R}^n$, $y \in \mathbb{R}^m$ are the variables. Standard formulations of linear programs often feature a vector $b \in \mathbb{R}^m$ instead of the vector $d \in \mathbb{R}^n$. Such standard formulations can be obtained by setting $b = \mathbf{A}d$. On the other hand, our formulation can be obtained from such a standard formulation by solving a linear system of the form $\mathbf{A}d = b$ to obtain d.

Informally, the goal is to find an algorithm that uses $\mathrm{poly}(n)$ 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.

As mentioned in the introduction, 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, [OV20; Vég17], 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 [DVZ18; IFF01].

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 $\mathbf{A} \in \mathbb{Z}^{m \times n}$ using a number of operations depending only on n and the logarithm of $\Delta_{\mathbf{A}}$, the maximum absolute value of the determinant of any square submatrix of $\mathbf{A}$. This algorithm is strongly polynomial for minimum-cost flow, noting that digraph incidence matrices are totally unimodular, and therefore $\Delta_{\mathbf{A}} = 1$. 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 [CGST86; Hof52]). More precisely, Tardos reduces computing exact primal-dual solutions to $\max \langle c, x \rangle, \mathbf{A}x = b, x \geq \mathbf{0}$ to computing exact primal-dual solutions to $O(nm)$ LPs in $\mathbf{A}$ with "rounded" objectives c' and right hand sides b' having integer coefficients of size $O(n^2\Delta_{\mathbf{A}})$. 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 $\bar{\chi}_{\mathbf{A}}$ dependence While Tardos's framework is powerful, it inherently relies on the determinant bound $\Delta_{\mathbf{A}}$. 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 $O(n^{3.5} \log(\bar{\chi}_{\mathbf{A}} + n))$ iterations. Improved iteration bounds were later given for certain special cases, in particular, $O(\sqrt{n} \log(\bar{\chi}_{\mathbf{A}} + n))$ for homogeneous conic feasibility [VY95] and $O(n^{2.5} \log(\bar{\chi}_{\mathbf{A}} + n))$ 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 $\Delta_{\mathbf{A}}$ is Stewart's $\bar{\chi}_{\mathbf{A}}$ [Ste89], which for integer matrices satisfies $\bar{\chi}_{\mathbf{A}} \leq n\Delta_{\mathbf{A}}$. In contrast with $\Delta_{\mathbf{A}}$ that relies on the entry numerics, $\bar{\chi}_{\mathbf{A}}$ is a geometric measure that depends only on the kernel of $\mathbf{A}$; Formally, letting $W \coloneqq \ker(\mathbf{A})$ and $\pi_I(W) = \{ x_I : x \in W \}$, one may define $\bar{\chi}_{\mathbf{A}} \coloneqq \bar{\chi}_W$ 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.

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 $O(n^{2.5} \log n \log(\bar{\chi}^*_{\mathbf{A}} + n))$ iterations, where $\bar{\chi}^*_{\mathbf{A}}$ is the minimum $\bar{\chi}_{\mathbf{A}D}$ 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 $\bar{\chi}_{\mathbf{A}D} \leq n(\bar{\chi}^*_{\mathbf{A}})^3$. Thus, by suitable preprocessing, any algorithm achieving $\bar{\chi}_{\mathbf{A}}$ dependence can be converted into one with $\bar{\chi}^*_{\mathbf{A}}$ dependence.

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

Harnessing the progress in approximate solvers The complexity of fast approximate LP algorithms has seen substantial improvements in recent years [Bra20; BTSS20; CLS19; JSWZ21; LS19; LSZ19]. Taking the recent algorithm [Bra20], given a feasible LP $\min \langle c, x\rangle, \mathbf{A}x = b, x \geq \mathbf{0}$, having an optimal solution of ℓ_2 norm at most R, for $\varepsilon > 0$ the approximate solver computes a point $\tilde{x}$ such that $\mathbf{A}\tilde{x} = b$ satisfying

$$\langle c, \tilde{x}\rangle \leq \min_{\mathbf{A}x=b, x\geq 0} \langle c, x\rangle + \varepsilon \cdot \|c\|_2 \cdot \mathrm{poly}(\kappa_{\mathbf{A}}, n) \quad \text{and} \quad \|\tilde{x}^-\|_2 \leq \varepsilon \cdot \mathrm{poly}(\kappa_{\mathbf{A}}, n)\|\mathbf{A}_B^{-1} b\|, \quad \text{(APX-LP)}$$

for any basis $B \in \mathcal{B}(\mathbf{A})$ in deterministic time $\widetilde{O}(n^\omega \log(n/\varepsilon))$, where $\omega < 2.38$ is the matrix multiplication exponent.

Tardos's framework requires an exact black box solver for systems with the same matrix $\mathbf{A}$ 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 $\mathbf{A} \in \mathbb{Z}^{m\times n}$, $b \in \mathbb{Z}^m$, $c \in \mathbb{Z}^n$ and $\|b\|_\infty, \|c\|_\infty \leq n^{O(1)}\Delta^t$, and let OPT denote the optimum value of (1.1). We may call (1.1) in a suitable extended system with $\varepsilon = 1/\left(n^{O(1)}\Delta_{\mathbf{A}}^{O(t)}\right)$, 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 $R = n^{O(1)}\Delta_{\mathbf{A}}^{O(t)}$ from Cramer's rule, and for showing that for any primal feasible solution x, $\langle c, x\rangle < \mathrm{OPT}$ implies $\langle c, x\rangle < \mathrm{OPT} - \varepsilon\|c\|_2 R$. For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, we cannot obtain an exact solver by applying the approximate solver for high enough accuracy in terms of the condition numbers $\bar{\chi}_{\mathbf{A}}$ or $\kappa_{\mathbf{A}}$. 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 $\bar{\chi}_{\mathbf{A}}$ 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 [BTSS20; LS19], 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 $\bar{\chi}_{\mathbf{A}}$ solvability of LP is due to "IPM magic". It would therefore be desirable to have an elementary proof of the $\bar{\chi}_{\mathbf{A}}$ solvability of LP.

Partial progress on this question was given by Ho and Tunçel [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 $\tau_{\mathbf{A}} > 0$, depending only on $\mathbf{A}$. Second, they showed that using a blackbox approximate LP solver, these rounded instances can be solved $\text{poly}(n, \log \tau_{\mathbf{A}}, \log(\Delta_{\mathbf{A}}/\delta_{\mathbf{A}}))$ time, where $\delta_{\mathbf{A}}$ is the absolute value of the minimum non-zero determinant of any square submatrix of $\mathbf{A}$. Here, they prove the relation $\bar{\chi}_{\mathbf{A}} \leq n\Delta_{\mathbf{A}}/\delta_{\mathbf{A}}$ and note that $\Delta_{\mathbf{A}}/\delta_{\mathbf{A}}$ can be arbitrarily larger than $\bar{\chi}_{\mathbf{A}}$. Lastly, they provide a different algorithm that removes the dependence on $\tau_{\mathbf{A}}$, assuming one has access to the VY algorithm as a subroutine only on instances with $b \in \{\pm 1, 0\}^m, c \in \{0, \pm 1\}^n$.

7.1.1 Our Contributions

As our main contribution, we provide a substantially improved Tardos style framework for LP which achieves both $\bar{\chi}_{\mathbf{A}}$ 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 [Bra20] is summarized below. The more precise technical statements generalized to non-deterministic solvers are given as Theorem 7.5.2 for feasibility and Theorem 7.6.2 for optimization.

The system (7.14) is an extended system used for initialization.

Theorem 7.1.1 (Enhanced Tardos Framework for Feasibility)**.** *Assume we are given a feasibility LP* $\mathbf{A}x = b, x \geq \mathbf{0}$ *with data* $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\text{rank}(\mathbf{A}) = m$, *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 $\varepsilon = 1/(n\bar{\chi}_{\mathbf{A}})^{O(1)}$, on extended systems of the form (Init-LP), together with additional $\widetilde{O}(mn^{\omega})$ arithmetic operations. This gives a total complexity $\widetilde{O}(mn^{\omega} \log(\bar{\chi}_{\mathbf{A}}))$ using the solver of van den Brand [Bra20].

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

Next, we state our result for optimization:

Theorem 7.1.2 (Enhanced Tardos Framework for Optimization)**.** *Assume we are given primal-dual (1.1) with data* $\mathbf{A} \in \mathbb{R}^{m \times n}$, $\text{rank}(\mathbf{A}) = m$, $b \in \mathbb{R}^m$, $c \in \mathbb{R}^n$.

(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 7.1.1(i), together with an additional $\widetilde{O}(mn^{\omega+1})$ arithmetic operations. This gives a total complexity $\widetilde{O}(mn^{\omega+1} \log(\bar{\chi}_{\mathbf{A}}))$ using [Bra20].

(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 $\widetilde{O}(n^3m^2 \log \log(\bar{\chi}_{\mathbf{A}}))$ additional arithmetic operations.

This theorem yields the first LP algorithm achieving $\bar{\chi}_{\mathbf{A}}$ 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 3.5. 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

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

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

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

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 $S^* = \operatorname{supp}(x^*)$, 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

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

where $\tilde{c} \in \mathbb{Z}^n$, $\|\tilde{c}\|_\infty \leq n^2\Delta_{\mathbf{A}}$, 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 $\|\tilde{b}\| = \Theta(\sqrt{m}\Delta_{\mathbf{A}}^n)$. As explained above, we can obtain an exact solution to ($\hat{P}$) by calling (APX-LP) for accuracy $\varepsilon = 1/\left(n^{O(1)}\tilde{\Delta}_{\mathbf{A}}^{O(n)}\right)$.

To conclude that $i \in T$ for some $i \in [n]$, Tardos uses a proximity theorem that is a variant of Lemma 3.5.5. 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 7.5 proceeds as follows. If $c \geq \mathbf{0}$, 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 a constraint of the form

$$\|s - c\|_\infty \leq p(\kappa_{\mathbf{A}}, n)\|c^-\|_1\,, \tag{7.1}$$

where $\kappa_{\mathbf{A}}$ is the circuit imbalance measure; for integer matrices $\kappa_{\mathbf{A}} \leq \Delta_{\mathbf{A}}$ and $p(\kappa_{\mathbf{A}}, n) = \operatorname{poly}(\kappa_{\mathbf{A}}, n)$ is some polynomial. A proximity result (Corollary 3.5.2) implies that whenever (D) is feasible, there is a feasible solution also satisfying (7.1).

We can use (APX-LP) directly to obtain a solution $(\tilde{y}, \tilde{s})$ such that $\mathbf{A}^\top\tilde{y} + \tilde{s} = c$, $\|\tilde{s} - c\|_\infty \leq \frac{1}{2}p(\kappa_{\mathbf{A}}, n)\|c^-\|_1$, and $\|\tilde{s}^-\|_\infty \leq \varepsilon\|c^-\|_1$ for $\varepsilon = 1/O(n^4\kappa_{\mathbf{A}}^4)$. 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 (7.1). 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 $\mathbf{A}$. 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 3.5.5 and Corollary 3.5.2 do not rely on integrality; we can use here $\kappa_{\mathbf{A}}$ instead of $\Delta_{\mathbf{A}}$.

Even for integer matrices and $\kappa_{\mathbf{A}} = \Theta(\Delta_{\mathbf{A}})$, 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 (7.1). This allows us to use $\varepsilon = 1/(n^{O(1)}\kappa_{\mathbf{A}}^{O(1)})$; otherwise, we would need to require a higher precision $\varepsilon = 1/(n^{O(1)}\kappa_{\mathbf{A}}^{O(n)})$. 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 $\kappa_{\mathbf{A}} = \Theta(\Delta_{\mathbf{A}})$ and $m = \Omega(n)$, the asymptotic running time bounds will be the same.

We now outline Tardos's approach. Given an optimization problem (1.1), 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 $\|\tilde{b}\|_\infty \leq n^2\Delta_{\mathbf{A}}$. 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 7.6, 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 3.5 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 $\mathbf{A}$ such that the set of optimal bases coincide in the two systems. The perturbed instance can be solved in $\text{poly}(n, m, \log \Delta_{\mathbf{A}})$; 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 (1.1). 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 $x_i(\mu^0)\bar{\chi}^n_{\mathbf{A}} \geq x_j(\mu^0)$ 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 $\mu/\bar{\chi}^n_{\mathbf{A}}$ 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 $\varepsilon = 1/(n\bar{\chi}_{\mathbf{A}})^{O(1)}$ 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 $(n\bar{\chi}_{\mathbf{A}})^{O(1)}$ 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 $\kappa_{\mathbf{A}}$. This is a common assumption shared by most previous literature: Tardos's algorithm uses an estimate of $\Delta_{\mathbf{A}}$; Vavasis and Ye require a bound on $\bar{\chi}_{\mathbf{A}}$. These parameters are hard to compute [Kha95a; Tun99]. However, knowing these values is not required, and we can use the following simple guessing procedure, attributed to J. Renegar in [VY96]. We start with a low guess on $\bar{\chi}_{\mathbf{A}}$ (or some other parameter), say $M = 2$. 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 $\log(\bar{\chi}_{\mathbf{A}} + n)$, 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 $W = \ker(\mathbf{A})$, 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 $\kappa_{\mathbf{A}}$ satisfies $\|z\|_\infty \leq \kappa_{\mathbf{A}}\|y\|_1$ (Proposition 3.3.7). Whenever our algorithm fails due to underestimating $M < \kappa_{\mathbf{A}}$, this will be certified by an index set $I \subseteq [n]$ and a vector $y \in \pi_I(W)$, and lift z with $\|z\|_\infty > M\|y\|_1$.

7.1.2 Organization

In Section 7.2, we recall the subspace formulation of LP and review important properties of the condition numbers $\bar{\chi}_{\mathbf{A}}$ and its combinatorial cousin, the circuit imbalance measure $\kappa_{\mathbf{A}}$. In Section 3.5, we gave an overview of existing and presented new LP proximity results, based on Hoffman type bounds. In Section 7.3, 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 7.4, 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 7.4.6 and 7.4.7 respectively. The proofs of these theorems are deferred to Section 7.7, where we also describe the LP extended system we use System 7.14. In Section 7.5, give the describe our framework for LP feasibility, and in Section 7.6 our framework for LP optimization.

7.2 Preliminaries

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

$$\begin{aligned}\bar{\chi}_{\mathbf{A}} &:= \sup\left\{\left\|\mathbf{A}^\top\left(\mathbf{A}\mathbf{D}\mathbf{A}^\top\right)^{-1}\mathbf{A}\mathbf{D}\right\| : D \in \mathfrak{D}\right\}\\ &= \sup\left\{\frac{\|\mathbf{A}^\top y\|}{\|p\|} : y \text{ minimizes } \left\|\mathbf{D}^{1/2}(\mathbf{A}^\top y - p)\right\| \text{ for some } 0 \neq p \in \mathbb{R}^n \text{ and } \mathbf{D} \in \mathfrak{D}\right\}.\end{aligned} \tag{7.2}$$

It is important to note that $\bar{\chi}\mathbf{A}$ only depends on the subspace $W = \ker(\mathbf{A})$. Hence, we can also write $\bar{\chi}_W$ for a subspace $W \subseteq \mathbb{R}^n$, defined to be equal to $\bar{\chi}\mathbf{A}$ for some matrix $A \in \mathbb{R}^{k\times n}$ with $W = \ker(\mathbf{A})$. We will use the notations $\bar{\chi}\mathbf{A}$ and $\bar{\chi}_W$ interchangeably. We studied several more properties of $\bar{\chi}$ in Chapter 3.

Recall the definition of the lifting map $L_I^W : \pi_I(W) \to W$ by

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

The affine subspace condition number We will now introduce condition numbers related to $\bar{\chi}$. They are going to be needed in the analysis of the optimization algorithm (Algorithm 7.2). For a subspace $W \subseteq [n]$ and a vector $d \in \mathbb{R}^n$ we define

$$\chi_W(d) := \max\left\{\, \|x^{B_1} - x^{B_2}\| : x^{B_i} \in W + d, \exists B_i \in \mathcal{B}_W \text{ s.t. } \operatorname{supp}(x^{B_i}) \subseteq B_i \text{ for } i = 1,2 \,\right\} \tag{7.3}$$

Note that we have that

$$\begin{aligned}\chi_W(d) &\leq \max\left\{\, \|x^{B_1}\| + \|x^{B_2}\| : x^{B_i} \in W + d, \exists B_i \in \mathcal{B}_W \text{ s.t. } \operatorname{supp}(x^{B_i}) \subseteq B_i \text{ for } i = 1,2 \,\right\}\\ &\leq 2\bar{\chi}\|\Pi_{W^\perp}(d)\|\,.\end{aligned} \tag{7.4}$$

Proposition 7.2.1. *Given a subspace $W \subseteq \mathbb{R}^n$, a vector $d \in \mathbb{R}^n$ and coordinates $I \subseteq [n]$ such that $\operatorname{supp}(d) \subseteq I$. Then, we have $\chi_W(d) \geq \chi_{W_I}(d_I)$.*

Proof. Any two bases $B_1, B_2 \in \mathcal{B}(W_I)$ with corresponding basic solution $x^{B_1}, x^{B_2} \in W_I + d_I$ can be extended to a bases $\widetilde{B}_1, \widetilde{B}_2 \in \mathcal{B}(W)$ with corresponding basic solutions $x^{\widetilde{B}_1} = (x^{B_1}, \mathbf{0}_{[n]\setminus I}), x^{\widetilde{B}_2} = (x^{B_2}, \mathbf{0}_{[n]\setminus I})$ such that $x^{\widetilde{B}_1}, x^{\widetilde{B}_2} \in W + d$. □

Definition 7.2.2 (Robust lifts). *Given a subspace $W \subseteq \mathbb{R}^n$, vector $d \in \mathbb{R}^n$, $d \notin W$ and $\varepsilon > 0$, we define the* robust *condition number*

$$\widetilde{\chi}_W(d, \varepsilon) := \sup_{x \in W+d} \{ \chi_W(\tilde{x}) : \|\tilde{x} - x\| \leq \varepsilon \}. \tag{7.5}$$

We have that $\widetilde{\chi}_W(d, \cdot)$ is monotonically increasing in the second argument and $\widetilde{\chi}_W(d, 0) = \chi_W(d)$.

Proposition 7.2.3 (Robust lifts under trivial subspaces). *Given a subspace $W \subseteq \mathbb{R}^n$, vector $d \in \mathbb{R}^n$, $\varepsilon > 0$ and some $I \subsetneq [n]$ such that $\pi_I(W) = \{\mathbf{0}\}$, then for $J := [n] \setminus I$ we have*

$$\widetilde{\chi}_W(d, \varepsilon) = \widetilde{\chi}_{\pi_J(W)}(d_J, \varepsilon). \tag{7.6}$$

Proof. We have

$$\begin{aligned}
\widetilde{\chi}_W(d, \varepsilon) &= \sup_{x \in W+d} \{ \chi_W(\tilde{x}) : \|\tilde{x} - x\| \leq \varepsilon \} \\
&= \sup_{x \in W+d} \left\{ \chi_{\pi_J(W)}(\tilde{x}_J) : \|\tilde{x} - x\| \leq \varepsilon \right\} \\
&= \sup_{x_J \in \pi_J(W)+d_J} \left\{ \chi_{\pi_J(W)}(\tilde{x}_J) : \|\tilde{x}_J - x_J\| \leq \varepsilon \right\} \\
&= \widetilde{\chi}_{\pi_J(W)}(d_J, \varepsilon).
\end{aligned}$$

□

The estimate M and lifting certificates The value of κ_W and $\bar{\chi}\mathbf{A}$ may not be known. In fact, these are hard to approximate even within a factor $2^{\text{poly}(m)}$ [Tun99]. Throughout our algorithms, we maintain a guess M on the value of κ_W, initialized as $M = 1$. 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 $M < \kappa$ by Proposition 3.3.7. Such a pair (I, p) is called a *lifting certificate* of $M < \kappa$. We can then restart the algorithm with an updated estimate $M \leftarrow \max\{2\|L_I^W(p)\|_\infty / \|p\|_1, M^2\}$. We formally define the set of lifting certificates

Definition 7.2.4 (Lifting certificates). *For a subspace $W \subseteq \mathbb{R}^n$ and a number $M \geq 1$ we define the set of lifting certificates $\mathfrak{C}(W, M)$ as*

$$\mathfrak{C}(W, M) := \left\{ (I, p) \in 2^{[n]} \times \mathbb{R}^n : p \in \pi_I(W), \|L_I^W(p)\|_\infty > M\|p\|_1 \right\}. \tag{7.7}$$

Definition 7.2.5 (Farkas certificates). *For a subspace $W \subseteq \mathbb{R}^n$ and a vector $d \in \mathbb{R}^n$ we define the set of* Farkas certificates *$\mathfrak{F}(W, d)$ as*

$$\mathfrak{F}(W) := \left\{ v \in W^\perp : v \geq 0, \langle v, d \rangle < 0 \right\}. \tag{7.8}$$

Note, that $LP(\mathbf{A}, d, \cdot)$ is primally feasible if and only if $\mathfrak{F}(\ker(\mathbf{A}), d) = \emptyset$.

Remark 7.2.6. *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 3.2.17 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 chapter.*

Fact 7.2.7. *Given a subset* $I \subseteq [n]$ *and an instance of* Primal$(W, d, \cdot)$. *A certificate of primal feasibility* y *of the subsystem* Primal$(\pi_W, d_I, \cdot)$ *extends canonically to a certificate of primal infeasibility* $(\mathbf{0}_{[n]\setminus I}, y)$ *of* Primal$(W, d, \cdot)$. *In particular,*

$$\left\{ (\mathbf{0}_{[n]\setminus I}, z) : z \in \mathfrak{F}(\pi_I(W), d, c) \right\} \subseteq \mathfrak{F}(W, d, c). \tag{7.9}$$

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

$$\bar{\chi}^*_W = \bar{\chi}^*\mathbf{A} = \inf\{\bar{\chi}_{WD} : D \in \mathfrak{D}\}$$
$$\kappa^*_W = \kappa^*\mathbf{A} = \inf\{\kappa_{WD} : D \in \mathfrak{D}\}$$

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

Theorem 7.2.8 ([DHNV20])**.** *There is an* $O(n^2m^2 + n^3)$ *time algorithm that for any matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *and* $W = \ker(\mathbf{A})$, *computes a value* t *such that*

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

and a $D \in \mathfrak{D}$ *such that*

$$\kappa_{WD} \leq (\kappa^*_W)^3 \text{ and } \bar{\chi}_{WD} \leq n(\bar{\chi}^*_W)^3.$$

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.

7.2.1 Treewidth

The following definitions are taken from [DLY21]. We simplify the notation as theirs handles general block-structured matrices $\mathbf{A}$. We essentially consider the special case where every block has size 1.

Definition 7.2.9 (Dual Graph of a Matrix)**.** *The* dual graph *of matrix* $\mathbf{A} \in \mathbb{R}^{m\times n}$ *with is the undirected graph* $G_\mathbf{A} = (V, E)$ *with* $V = [m]$, *and* $\{i, j\} \in E$ *if and only if* $\mathbf{A}_{i,r} \neq 0$ *and* $\mathbf{A}_{j,r} \neq 0$ *for some* $r \in [n]$.

Definition 7.2.10 (Tree decomposition)**.** *A* tree-decomposition *of a graph* G *is a pair* (X, T), *where* T *is a tree, and* $X\colon V(T) \to 2^{V(G)}$ *is a family of subsets of* $V(G)$ *called* bags *labelling the vertices of* T, *such that*

1. $\bigcup_{t\in V(T)} X(t) = V(G)$,
2. *for each* $v \in V(G)$, *the nodes* $t \in V(T)$ *with* $v \in X(t)$ *induce a connected subgraph of* T, *and*
3. *for each* $e = \{u, v\} \in E(G)$, *there exists a node* $t \in V(T)$ *such that* $\{u, v\} \subseteq X(t)$.

The width *of a tree-decomposition* (X, T) *is defined to be* $\max\{ |X(t)| - 1 : t \in T \}$.

Definition 7.2.11 (Treewidth)**.** *The* treewidth tw *of* G *is the minimum width over all tree-decompositions of* G. *Intuitively, the treewidth of a graph captures how close the graph is to being a tree. We denote by* $\mathrm{tw}(\mathbf{A})$ *the treewidth of the dual graph* $G_\mathbf{A}$ *of* $\mathbf{A}$.

7.3 Constructive proximity algorithms

System 7.1. *Sign-Consistent-Circuit*

Data: A pair (W, y), where $W \subseteq \mathbb{R}^n$ is a subspace, $y \in W$.

Task: Find a vector $z \in W$ that is sign-consistent with y and such that $\operatorname{supp}(z) \in \mathcal{C}(W)$.

We let $\mathcal{CD}(W, y)$ denote the set of sign-consistent circuit decompositions of a vector y in a subspace W.

System 7.2. *Sign-Consistent-Circuit-Decomposition*

Data: A pair (W, y), where $W \subseteq \mathbb{R}^n$ is a subspace, $y \in W$.

Task: Find a sign-consistent circuit decomposition of y, i.e., an element in $\mathcal{CD}(W, y)$.

We begin this section with two basic lemmas.

Lemma 7.3.1. *Sign-Consistent-Circuit can be implemented in time* $O(n\mathcal{T}(\mathfrak{L}(\mathbf{A})))$ *if* $W = \ker(\mathbf{A})$ *or* $W = \operatorname{im}(\mathbf{A}^\top)$.

Proof. A basis B of $\mathbf{A}$ such that $B \cap \operatorname{supp}(y)$ is maximized, can be found within n linear system solves with matrix $\mathbf{A}$. Given a basis $B \in \mathcal{B}(\mathbf{A})$, one can solve the linear system $\mathbf{A}_B x = \mathbf{A}_i$ for some $i \in \operatorname{supp}(y) \setminus B$. Let x be the solution. Then we can augment y by a scalar multiple α of x such that the vector $y' := y - \alpha x$ is sign-consistent with y and has strictly smaller support than y, i.e., $\operatorname{supp}(y') \subsetneq \operatorname{supp}(y)$. Repeat this procedure with a new basis B' of $\mathbf{A}$ which can be found by pivoting within a single linear system solve until y' is supported on a circuit, i.e., $\operatorname{supp}(y') \in \mathcal{C}(W)$. This gives an overall $O(n)$ linear system solves. An analogous algorithm can be implemented for $\operatorname{im}(\mathbf{A}^\top)$. □

Lemma 7.3.2. *There is an algorithm for System 7.2 that runs in time* $O(n^2\mathcal{T}(\mathfrak{L}(\mathbf{A})))$ *if* $W = \ker(\mathbf{A})$ *or* $W = \operatorname{im}(\mathbf{A}^\top)$.

Proof. We repeatedly invoke a solver to System 7.1 to find a vector $g \in W$, $\operatorname{supp}(g) \in \mathcal{C}(W)$ that is sign-consistent with y and set $y' := y - \alpha g$, where α is chosen such that y' is sign-consistent with y and has strictly smaller support than y. We invoke System 7.1 at most n times. With Lemma 7.3.1 this gives a total of $O(n^2)$ linear system solves. □

In this section, we give an algorithmic implementation of the Hoffman Proximity Theorem (Theorem 3.5.1), assuming that a feasible solution is already given: we obtain another feasible solution in strongly polynomial time that also satisfies the proximity bounds.

This 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 7.3.3, and the main algorithm described in Lemma 7.3.4.

System 7.3. *Hoffman-proximity-1*

Data: A tuple $(\mathbf{A}, y, J, M)$ with $\mathbf{A} \in \mathbb{R}^{m\times n}$, $y \in \ker(\mathbf{A})$, $J \subseteq [n]$, $M \geq 1$ and let $W = \ker(\mathbf{A})$.

Task: Find one of the following:

1. A vector $z \in W$, such that z is sign-consistent with y, $\|z - y\|_\infty \leq M\|y_J\|_1$, and $z_J = \mathbf{0}$,
2. A pair $(I, p) \in \mathfrak{C}(W, M)$.

We denote algorithms that implement System 7.3 by $\hbar_{\ker}$. Analogously, we denote the algorithms for System 7.3, where $\ker(\mathbf{A})$ is replaced by $\mathrm{im}(\mathbf{A}^\top)$ by $\hbar_{\mathrm{im}}$.

We denote by $\mathcal{T}(\hbar_{\ker}, \mathbf{A})$ and $\mathcal{T}(\hbar_{\mathrm{im}}, \mathbf{A})$ the smallest of the runtimes over all algorithms that implement $\hbar_{\ker}(\mathbf{A}, \cdot)$ resp. $\hbar_{\mathrm{im}}(\mathbf{A}, \cdot)$.

Lemma 7.3.3. *System 7.3 can be implemented with $O(n)$ linear system solves. That is, $\mathcal{T}(\hbar_{\ker}) = O(n\mathcal{T}(\mathfrak{L}(\mathbf{A})))$ and $\mathcal{T}(\hbar_{\mathrm{im}}) = O(n\mathcal{T}(\mathfrak{L}(\mathbf{A})))$.*

Proof. The proof shares similarities to the one in Lemma 7.3.1. Find a basis B of $\mathbf{A}_{n\setminus J}$ and solve $\mathbf{A}_B x = \mathbf{A}_J y_J$. Then set $y' := y - \alpha x$ where $\alpha \geq 0$ is chosen maximal such that y' is sign-consistent with y. Repeat this procedure until $y'_J = \mathbf{0}$. The proximity bound follows automatically if we havenot encountered a certificate in $\mathfrak{C}(W, M)$. An analogous argument works on the image of $\mathbf{A}^\top$. □

Lemma 7.3.4. *There exists an implementation of System 7.4 that requires $O(n^2)$ linear system solves. That is $\mathcal{T}(\mathscr{H}_{\ker}) = O(n^2\mathcal{T}(\mathfrak{L}(\mathbf{A})))$ and $\mathcal{T}(\mathscr{H}_{\mathrm{im}}) = O(n^2\mathcal{T}(\mathfrak{L}(\mathbf{A})))$.*

Proof. Given an instance (W, y, J, M) of System 7.4 compute a sign-consistent circuit decomposition of y using Lemma 7.3.2 or conclude with an element in $\mathfrak{C}(W, M)$. Let the circuit decomposition be $y = \sum_{i=1}^k \alpha_i g^i$ with $k \leq n$ and for all $i \in [k]$, $\mathrm{supp}(g^i) \in C(W)$, $\alpha_i > 0$. We let $z^0 := \mathbf{0}$ and set iteratively $z^i := z^{i-1} + \beta_i g^i$ where $0 \leq \beta_i \leq \alpha_i$ is chosen as the smallest value β such that $\|(\ell - z^{i-1} - \beta g^i)^+\| + \|(u - z^{i-1} - \beta g^i)^-\|$ is minimized. Intuitively, this means that β is raised as long as it decreases the gap on a coordinate in $\mathrm{supp}(\ell^+)$ or a coordinate in $\mathrm{supp}(u^-)$. For $z := z^k$ it is now easy to see that $z \in W, \ell \leq z \leq u$ and $\|z\|_1 \leq M\|\ell^+ + u^-\|_\infty$. □

The following two corollaries will make use of the assumption that $n - m = \Theta(n)$.

System 7.4. *Hoffman-proximity-2*

Data: A tuple (W, y, ℓ, u, M) with $W \subseteq \mathbb{R}^n$, $y \in W$, $\ell \in (\mathbb{R} \cup \{-\infty\})^n, u \in (\mathbb{R} \cup \{\infty\})^n$ such that $\ell \leq y \leq u$ and $M \geq 1$.

Task: Find one of the following:

1. A vector $z \in W$ such that $\ell \leq z \leq u$ and $\|z\|_\infty \leq M\|\ell^+ + u^-\|_1$,
2. A pair $(I, p) \in \mathfrak{C}(W, M)$.

We denote algorithms that implement System 7.4 where $W = \ker(\mathbf{A})$ by $\mathscr{H}_{\ker}$. Analogously, we denote the algorithms for System 7.4, where $\ker(\mathbf{A})$ is replaced by $\mathrm{im}(\mathbf{A}^\top)$ by $\mathscr{H}_{\mathrm{im}}$.

We denote by $\mathcal{T}(\hbar_{\ker}, \mathbf{A})$ and $\mathcal{T}(\hbar_{\mathrm{im}}, \mathbf{A})$ the smallest of the runtimes over all algorithms that implement $\hbar_{\ker}(\mathbf{A}, \cdot)$ resp. $\hbar_{\mathrm{im}}(\mathbf{A}, \cdot)$.

Note that using fast matrix multiplication and techniques from Section 4.2 of [Bra] the time complexity per pivot can be reduced to $O(n^{1.529})$ instead of a linear system solve, leading to improved running times in settings, where no structure of $\mathbf{A}$ can be exploited in linear system solves.

7.4 Black-box algorithms

7.4.1 Black-box exact linear algebra

System 7.5. *Apx-Projection-Kernel*

Data: A tuple $(\mathbf{A}, d, \Gamma, I)$, where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $W \coloneqq \ker(\mathbf{A})$, $d \in \mathbb{R}^n$, $\Gamma \geq 1$ and $I \subseteq [n]$.

$$\begin{aligned} &\text{find} && x \in \mathbb{R}^n && \\ &\text{s.t.} && x \in W + d && \text{(Primal subspace constraint),} && (7.10a)\\ & && \|x_I\| \leq \Gamma \min\{\, \|y_I\| : y \in W + d \,\} && \text{(Apx. projection onto } I\text{)} && (7.10b) \end{aligned}$$

System 7.6. *Apx-Projection-Image*

Data: A tuple $(\mathbf{A}, c, \Gamma, I)$, where $\mathbf{A} \in \mathbb{R}^{m\times n}$, $W \coloneqq \ker(\mathbf{A})$, $c \in \mathbb{R}^n$, $\Gamma \geq 1$ and $I \subseteq [n]$.

$$\begin{aligned} &\text{find} && s \in \mathbb{R}^n && \\ &\text{s.t.} && s \in W^\perp + c && \text{(Dual subspace constraint),} && (7.11a)\\ & && \|s_I\| \leq \Gamma \min\{\, \|y_I\| : y \in W^\perp + c \,\} && \text{(Apx. projection onto } I\text{)} && (7.11b) \end{aligned}$$

Definition 7.4.1. *For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ we denote by $\mathfrak{P}(\mathbf{A})$ the asymptotic running time to solve System 7.5 and System 7.6 exactly, i.e., for $\Gamma = 1$, that is for data $(\mathbf{A}, d, 1, I)$ for any $d \in \mathbb{R}^n$ and $I \subseteq [n]$.*

Definition 7.4.2. *For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ we denote by $\mathfrak{P}_\approx(\mathbf{A}, \Gamma)$ the asymptotic running time such that System 7.5 and System 7.6 can be solved with Γ for any $d \in \mathbb{R}^n$ and $I \subseteq [n]$.*

Definition 7.4.3. *For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ we denote by $\mathfrak{L}(\mathbf{A})$ the running time to solve any linear system of the form $\mathbf{A}x = b$ and $\mathbf{A}\mathbf{D}\mathbf{A}^\top x = b$, $\mathbf{D} \in \mathfrak{D}$ for x exactly.*

Note that for Definition 7.4.3 it would suffice to be able to solve $\mathbf{A}\mathbf{D}\mathbf{A}^\top x = b$, as any solution x automatically gives a solution y to $\mathbf{A}y = b$ via $y = \mathbf{D}\mathbf{A}^\top x$ with computational complexity $\mathrm{nnz}(\mathbf{A})$.

7.4.2 Black-box linear programming solvers

Our feasibility and optimization algorithms in Sections 7.5 and 7.6 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 7.4.4 ([Bra+21; Bra20; DLY21; JSWZ21; LS19]). *Blackbox-Approx-Solver (Oracle 7.3) with an initial point around μ_0 and with precision ε can be implemented in time*

(1) In $\widetilde{O}\big((mn + m^{2.5})\log(\mu_0/\varepsilon)\big)$ expected running time [Bra+21].

(2) In $\widetilde{O}(n^\omega \log(\mu_0/\varepsilon))$ deterministic running time, assuming $\omega \geq 13/6$ [Bra20]. The same expected running time is achievable assuming $\omega \geq 2 + 1/18$ [JSWZ21].

(3) In $\widetilde{O}\big((\mathrm{nnz}(\mathbf{A}) + m^2)\sqrt{m}\log(\mu_0/\varepsilon)\big)$ expected running time [LS19], where $\mathrm{nnz}(\mathbf{A})$ denotes the number of nonzero entries in $\mathbf{A}$.

(4) In $\widetilde{O}(n\mathrm{tw}(\mathbf{A})^2(\mu_0/\varepsilon))$ expected running time, where $\mathrm{tw}(\mathbf{A})$ is the treewidth of matrix $\mathbf{A}$ [DLY21].

We use the notation $\Psi(\mathbf{A})$ to denote the 'cost per unit' in these results. That is, we define $\Psi(\mathbf{A})$ as the smallest runtime such that Blackbox-Approx-Solver can be implemented in time $\widetilde{O}(\Psi(\mathbf{A})\log(\mu_0/\varepsilon))$. By Theorem 7.4.4 we have that

$$\Psi(\mathbf{A}) \leq \min\left\{mn + m^{2.5}, n^\omega, \sqrt{m}\left(\mathrm{nnz}(\mathbf{A}) + m^2\right), n\mathrm{tw}(\mathbf{A})^2\right\}. \tag{7.12}$$

We are going to make some mild assumptions regarding $\Psi(\mathbf{A})$ under modifications to the matrix $\mathbf{A}$. This is important for the recursive calls to theblackbox LP solvers that are going to deploy as well as for the initialisation system used in System 7.14.

Assumption 7.4.5. *Given a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, then for any submatrices $\mathbf{B} \in \mathbb{R}^{p\times q}$ of $\mathbf{A}$ and $\mathbf{C} \in \mathbb{R}^{p\times r}$ and a vector $v \in \mathbb{R}^{q+r}$ we have for*

$$\overline{\mathbf{A}} := \begin{bmatrix} -\mathbf{B} & \mathbf{A} \\ & v^\top \end{bmatrix} \tag{7.13}$$

that $\Psi\big(\overline{\mathbf{A}}\big) = O(\Psi(\mathbf{A}))$. We further assume that $\Psi(\mathbf{A}) = \Omega(\mathrm{nnz}(\mathbf{A}))$.

The second part of Assumption 7.4.5 is very natural and is true in most cases $\mathrm{nnz}(\mathbf{A})$ is the time to process a matrix. We require this assumption because we will repeatedly compute matrix-vector products $\mathbf{A}x$ and $\mathbf{A}^\top y$, both procedure take $\mathrm{nnz}(\mathbf{A})$ time and should not dominate the time to solve a linear system. On the other hand, this second assumption might not be true in an amortized sense, were maybe many linear systems are solved with the matrix $\mathbf{A}$ such that through preprocessing a basis or other structure of the matrix could be determined such that the amortized runtime per linear system solve might drop below $\mathrm{nnz}(\mathbf{A})$. It could also be the case that the matrix is given implicitly through a lot of structure, e.g., as a product of two matrices, both of which allow cheap linear system solves; or the columns of $\mathbf{A}$ could correspond to some underlying combinatorial structure. We want to exclude all these scenarios with the second part of Assumption 7.4.5.

We now state the main forms of feasibility and optimization oracles we use. In Section 7.7, we derive these results from Theorem 7.4.4, by running the algorithms on an extended system. The oracles used in Sections 7.5 and 7.6 can be implemented from Theorems 7.4.6 and 7.4.7.

System 7.7. *Apx-Feas*

Data: A tuple (W, d, M, ε), where $W \subseteq \mathbb{R}^n$ is a subspace, $d \in \mathbb{R}^n$, guess $M \geq 1$, and $\varepsilon > 0$.

$$
\begin{aligned}
&\text{find} && x \in \mathbb{R}^n && && \\
&\text{s.t.} && x \in W + d && \text{(Primal subspace constraint),} && (7.14a)\\
& && \|x^-\| \leq \varepsilon \|d\| && \text{(Primal apx. feasibility),} && (7.14b)\\
& && \|x\| \leq C_f \|d\| && \text{(Primal norm bound)} && (7.14c)
\end{aligned}
$$

where $C_f := n^{1.5} M = O(\text{poly}(n, M))$.

We want to remark that the choice of C_f is somewhat optimized in that the exponent in n and M is as small as possible for the upcoming proofs in this section to work out. Nonetheless, any choice of C_f that is polynomial in n and M would be fine to get the desired runtimes for the blackbox algorithms.

Theorem 7.4.6 (Proof on p. 182)**.** *There exists a pair of algorithms (Algorithm 7.4 and Algorithm 7.5) that returns either of the following outcomes:*

(F1) Near feasible primal-dual solutions (x, s) such that

(F1.1) x is feasible to Apx-Feas(W, d, M, ε), and

(F1.2) s is feasible to Apx-Feas$(W^\perp, c, M, \varepsilon)$.

(F2) A Farkas certificate of primal infeasibility: $y \in \mathfrak{F}(W, d)$,

(F3) A Farkas certificate of dual infeasibility: $y \in \mathfrak{F}(W^\perp, c)$,

(F4) A lifting certificate $(I, p) \in \mathfrak{C}(W, M)$.

The running time depends on the outcome and is presented in Table 7.1.

Outcome	Running time
(F1)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big)$
(F2)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathcal{K}_{\text{im}}^{n\times m}(\mathbf{A}^\top))$
(F3)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathcal{K}_{\text{ker}}^{m\times n}(\mathbf{A}))$
(F4)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathcal{K}_{\text{im}}^{n\times m}(\mathbf{A}^\top)) + \mathcal{T}(\mathcal{K}_{\text{ker}}^{m\times n}(\mathbf{A}))$

Table 7.1: Running times to implement Theorem 7.4.6

System 7.8. *Apx-Opt*

Data: A tuple $(W, d, c, M, \varepsilon)$, where $W \subseteq \mathbb{R}^n$ is a subspace, $d, c \in \mathbb{R}^n$, guess $M \geq 1$, and $\varepsilon > 0$.

find $x, s \in \mathbb{R}^n$

s.t.	$x \in W + d$	(Primal subspace constraint),	(7.15a)
	$s \in W^\perp + c$	(Dual subspace constraint),	(7.15b)
	$\|x^-\| \leq \varepsilon \|d\|$	(Primal apx. feasibility),	(7.15c)
	$\|s^-\| \leq \varepsilon \|c\|$	(Dual apx. feasibility),	(7.15d)
	$\|x \circ s\| \leq \varepsilon \|c\| \|d\|$	(Approximate complementarity),	(7.15e)
	$\|x\| \leq C_o \|d\|$	(Primal norm bound),	(7.15f)
	$\|s\| \leq C_o \|c\|$	(Dual norm bound)	(7.15g)

where $C_o := 4n^{3/2}M = O(\mathrm{poly}(n, M))$.

Theorem 7.4.7 (Proof on p. 185)**.** *Let an instance of a linear program of the form System 1.2 and $M \geq 1$, $\varepsilon > 0$ be given. There exists an algorithm (Algorithm 7.6) that returns either of the following outcomes:*

(M1) A pair of primal and dual near-feasible and near-optimal solutions (x, s) as solutions to System 7.8, i.e., Apx-Opt$(W, d, c, M, \varepsilon)$.

(M2) A Farkas certificate of primal infeasibility $y \in \mathfrak{F}(W, d)$,

(M3) A Farkas certificate of dual infeasibility $y \in \mathfrak{F}(W^\perp, c)$, or

(M4) A lifting certificate $(I, p) \in \mathfrak{C}(W, M)$.

The running time depends on the outcome and is as follows:

Outcome	Running time
(M1)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big)$
(M2)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}^{n\times m}_{\mathrm{im}}(\mathbf{A}^\top))$
(M3)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}^{m\times n}_{\ker}(\mathbf{A}))$
(M4)	$\widetilde{O}\big(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}^{n\times m}_{\mathrm{im}}(\mathbf{A}^\top)) + \mathcal{T}(\mathscr{H}^{m\times n}_{\ker}(\mathbf{A}))$

7.5 The feasibility algorithm

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

A key insight is to work with a stronger system, including a proximity constraint. According to Corollary 3.5.2, whenever the problem $x \in W + d$, $x \geq \mathbf{0}$ is feasible and $\kappa_W \leq M$, then System 7.9 is also feasible. In fact, this would be true even with the stronger bound M instead of $C^{\mathrm{alg}}_{\mathrm{pf}}$; we use this weaker bound to leave sufficient slack for the recursive argument. Note that if $d \geq \mathbf{0}$, then the only feasible solution is $x = d$.

System 7.9. *Alg-Feas*

Data: A tuple (W, d, M), where $W \subseteq \mathbb{R}^n$ is a subspace, $d \in \mathbb{R}^n$, guess $M \geq 1$.

$$\begin{aligned} \text{find} \quad & x \in \mathbb{R}^n \\ \text{s.t.} \quad & x \in W + d && \text{(Subspace constraint)}, && \text{(7.16a)} \\ & \|x - d\| \leq C_{\text{pf}}^{\text{alg}} \|d^-\| && \text{(Proximity constraint)}, && \text{(7.16b)} \\ & x \geq \mathbf{0} && \text{(Non-negativity)} && \text{(7.16c)} \end{aligned}$$

where $C_{\text{pf}}^{\text{alg}} := 8C_{\text{pf}} = O(\text{poly}(n, M))$.

We use a black-box approach assuming an oracle that returns an approximately feasible solution. The oracle will give a solution to Prox-Feas and is implemented by Oracle 7.1. Outcome (i) gives an approximately feasible solution with a bound on the negative components and a slightly stronger proximity guarantee than in System 7.9. Outcome (ii) gives a Farkas certificate of infeasibility in $\mathfrak{F}(W, d)$, whereas outcome (iii) gives a lifting certificate of $M < \kappa_W$, i.e, an element in $\mathfrak{C}(W, M)$.

System 7.10. *Prox-Feas*

Data: A tuple (W, d, M, ε), where $W \subseteq \mathbb{R}^n$ is a subspace, $d \in \mathbb{R}^n$, guess $M \geq 1$, and $\varepsilon > 0$.

$$\begin{aligned} \text{find} \quad & x \in \mathbb{R}^n \\ \text{s.t.} \quad & x \in W + d && \text{(Subspace constraint)}, && \text{(Q1)} \\ & \|x - d\| \leq C_{\text{pf}} \|d^-\| && \text{(Proximity constraint)}, && \text{(Q2)} \\ & \|x^-\| \leq \varepsilon \|d^-\| && \text{(Apx. feasibility constraint)} && \text{(Q3)} \end{aligned}$$

where $C_{\text{pf}} := 8C_f M n^{3/2} = O(\text{poly}(n, M))$.

Oracle 7.1. *Prox-Feas-Solver*

Input: A tuple (W, d, M, ε), with a subspace $W \subseteq \mathbb{R}^n, d \in \mathbb{R}^n$, a guess $M \in \mathbb{R}_+$ and parameter $\varepsilon > 0$.

Output: One of the following outcomes:

(i) A solution x to System 7.10, i.e., Prox-Feas(W, d, M, ε),

(ii) A Farkas certificate $y \in \mathfrak{F}(W, d)$,

(iii) A subset $I \subseteq [n]$ and a vector $p \in \pi_I(W)$ such that $(I, p) \in \mathfrak{C}(\pi_I(W), M)$.

This oracle can be derived from Theorem 7.4.6, by finding an approximately feasible solution to a modification of the system $x \in W + d, x \geq \mathbf{0}$. The derivation is given in Section 7.7.2; the running time is stated as follows.

Lemma 7.5.1 (Proof in Section 7.7). *Given a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ and vector $d \in \mathbb{R}^n$. Let $W = \ker(\mathbf{A})$, and M be an estimate on κ_W. Further, let $0 < \varepsilon < 1$. There exists an implementation of Oracle 7.1 for data (W, d, M, ε) with following outcome-dependent running times:*

Outcome	Running time
(i)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big)$
(ii), *(iii)*	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))$

The algorithm is stated in Algorithm 7.1. It allows to solve Alg-Feas(W, d, M) by combining an approximate solution x to Prox-Feas(W, d', M, ε) for some $d' \in W + d$ with an exact solution to Alg-Feas$(\pi_I(W), d_I, M)$ obtained recursively from a smaller system on a coordinate subset $I \subseteq [n]$.

Recall that for a set $K \subseteq [n]$, $\mathrm{cl}(K)$ denotes the closure of K, i.e., the unique largest set $C \subseteq [n]$ such that $C \subseteq J$ and $\mathrm{rk}(\mathbf{A}_C) = \mathrm{rk}(\mathbf{A}_K)$ where $\ker(\mathbf{A}) = W$ as usual.

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 $x^* \in W + d$, $x^* \geq \mathbf{0}$ with $x^*_i \gg 0$. We project out all these indices, along with all other indices $J := \mathrm{cl}(K) \setminus K$ in their closure, and recurse on the remaining index set $I := [n] \setminus \mathrm{cl}(K) = [n] \setminus (K \cup J)$. We note that the purpose of the set J is to to reduce the dimension of the dual space in the recursive call.

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 $d' = \Pi_{W^\perp}(d)$; hence $W + d' = W + d$. However, this choice is crucial due to the proximity bounds: System 7.9 features $\|d^-\|$ as well as a bound on $\|x - d\|$.

In particular, if $\|d^-\|$ is 'too big', then we may end up with an empty index set K and cannot recurse. In this case, we swap to $d' = \Pi_{W^\perp}(d)$; otherwise, we keep $d' = d$. We note that always swapping to $d' = \Pi_{W^\perp}(d)$ does not work either: System 7.9 features the bound $\|x - d\|$, and using $\|x - \Pi_{W^\perp}(d)\|$ 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 $\Pi_{W^\perp}(d)$ is due to the bound $\|x\| \geq \|\Pi_{W^\perp}(d)\|$ for any $x \in W + d$.

The overall feasibility algorithm is given in Algorithm 7.1. The output can be *(i)* a feasible solution to System 7.9; *(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 $\|L^W_I(p)\|_\infty > M\|p\|_1$. In this case, we can restart the entire algorithm, after updating M to $\max\{\|L^W_I(p)\|_\infty/\|p\|_1, 2M^2\}$.

The algorithm calls Oracle 7.1. 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 $\pi_I(W)$.

We are now ready to state the central theorem of this section, which in particular proves Theorem 7.1.1.

Theorem 7.5.2. *Algorithm 7.1 is correct. The runtime depends on the outcome and is as follows:*

Outcome	Running time
Primal(W, d)	$\widetilde{O}\big(m(\Psi(\mathbf{A})\log(M) + \mathcal{T}(\mathfrak{P}(\mathbf{A})) + \mathcal{T}(\mathrm{cl}(\mathbf{A})))\big)$
$\mathfrak{F}(W, d)$	$\widetilde{O}\big(m\Psi(\mathbf{A})\log(M) + m\mathcal{T}(\mathfrak{P}(\mathbf{A})) + m\mathcal{T}(\mathrm{cl}(\mathbf{A})) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))\big)$
$\mathfrak{C}(W, M)$	$\widetilde{O}\big(m\Psi(\mathbf{A})\log(M) + m\mathcal{T}(\mathfrak{P}(\mathbf{A})) + m\mathcal{T}(\mathrm{cl}(\mathbf{A})) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))\big)$

Algorithm 7.1: Feasibility-Algorithm

Input : $\mathbf{A} \in \mathbb{R}^{m\times n}$, $W \subseteq \mathbb{R}^n$ subspace, $d \in \mathbb{R}^n$, guess $M \geq 1$.
Output : An element in Alg-Feas(W, d, M), $\mathfrak{F}(W, d)$ or $\mathfrak{C}(W, M)$.

if $\|d^-\| \geq \max\left\{\|\Pi_{W^\perp}(d)\|, \frac{\|d\|}{2C_{\text{pf}}}\right\}$ **then**
  $d \leftarrow \Pi_{W^\perp}(d)$
if $d \geq 0$ **then**
  return d ▷ in Alg-Feas(W, d, M)
$T \leftarrow 2\sqrt{n}MC_{\text{pf}}^{\text{alg}}$
$\varepsilon \leftarrow \frac{1}{\sqrt{n}T}$
$\diamond \leftarrow$ Prox-Feas-Solver(W, d, M, ε)
switch $\diamond$ **do**
  case *(i)* $\diamond \in$ Prox-Feas(W, d, M, ε) **do**
    go to 15
  case *(ii)* $\diamond \in \mathfrak{F}(\pi_I(W), x_I)$ **do**
    return *Farkas certificate* $\diamond$
  case *(iii)* $\diamond \in \mathfrak{C}(\pi_I(W), M)$ **do**
    return *lifting certificate* $\diamond$
$x \leftarrow \diamond$
$K \leftarrow \{\, i \in [n] : x_i \geq T\|x^-\| \,\}$
$J \leftarrow \text{cl}(K) \setminus K$
$I \leftarrow [n] \setminus (J \cup K)$
$\spadesuit \leftarrow$ Feasibility-Algorithm$(\pi_I(W), x_I, M)$
switch $\spadesuit$ **do**
  case *(i)* $\spadesuit \in$ Alg-Feas$(\pi_I(W), x_I, M)$ **do**
    $w \leftarrow \spadesuit$
    $\bar{w} \leftarrow L^W_{I\cup J}(w - x_I, x_J^-)$
    if $\|\bar{w}\|_\infty \leq M\|(w - x_I, x_J^-)\|_1$ **then**
      return $x + \bar{w}$ ▷ in Alg-Feas(W, d, M)
    else
      return $(I \cup J, (w - x_I, x_J^-))$ ▷ $\in \mathfrak{C}(W, M)$
  case *(ii)* $\spadesuit \in \mathfrak{F}(\pi_I(W), x_I)$ **do**
    $y' \leftarrow (\spadesuit, \mathbf{0}_{J\cup K})$
    return y' ▷ in $\mathfrak{F}(W, d)$
  case *(iii)* $\spadesuit \in \mathfrak{C}(\pi_I(W), M)$ **do**
    return $\spadesuit$ ▷ in $\mathfrak{C}(W, M)$

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

Feasibility of x': Let us first show that x' is well-defined; this requires 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 $\hat{w} \in W + x$. We have that $w - x_I = (\hat{w} - x)_I \in \pi_I(W)$. 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 $W + x = W + d$. Hence, we have (7.16a). 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 the non-negativity constraint (7.16c) holds for $x' \geq 0$. 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, $x_k \geq T\|x^-_{I\cup J}\|$. Then, $x'_k \geq 0$ follows as

$$\begin{aligned}\|L^W_{I\cup J}\big((w - x_I, x_J^-)\big)\|_\infty \leq M\|(w - x_I, x_J^-)\|_1 \leq \sqrt{n}M\|(w - x_I, x_J^-)\| \leq \sqrt{n}M(C_{\text{pf}}^{\text{alg}} + 1)\|x^-_{I\cup J}\| \\ \leq T\|x^-_{I\cup J}\|\end{aligned} \tag{7.18}$$

To complete the proof that x' is feasible to Alg-Feas(W, d, M), it remains to verify the proximity bound (7.16b), i.e., $\|x' - d\| \leq C_{\text{pf}}^{\text{alg}}\|d^-\|$. First, we need an auxiliary claim.

Claim 7.5.2.1. *$\|x - d\| \leq 4C_{\text{pf}}\|d^-\|$ and $\|x^-\| \leq \varepsilon\|d^-\|$.*

Proof. If $d' = d$, then from the feasibility of x to Prox-Feas(W, d', M, ε), we have $\|x - d\| \leq C_{\text{pf}}\|d^-\|$ and $\|x^-\| \leq \varepsilon\|d^-\|$. If $d' \neq d$, then $d' = \Pi_{W^\perp}(d)$, $\|d^-\| > \|\Pi_{W^\perp}(d)\|$ and $\|d^-\| > \frac{\|d\|}{2C_{\text{pf}}}$ and so

$$\begin{aligned}\|x - d\| &\leq \|x - \Pi_{W^\perp}(d)\| + \|\Pi_{W^\perp}(d)\| + \|d\| \\ &\leq C_{\text{pf}}\|\Pi_{W^\perp}(d)^-\| + \|\Pi_{W^\perp}(d)\| + 2C_{\text{pf}}\|d^-\| \\ &\leq (C_{\text{pf}} + 1)\|\Pi_{W^\perp}(d)\| + 2C_{\text{pf}}\|d^-\| \\ &\leq (2C_{\text{pf}} + 2C_{\text{pf}})\|d^-\| \\ &\leq 4C_{\text{pf}}\|d^-\|\,, \quad \text{and} \\ \|x^-\| &\leq \varepsilon\|\Pi_{W^\perp}(d)^-\| \leq \varepsilon\|\Pi_{W^\perp}(d)\| \leq \varepsilon\|d^-\|.\end{aligned}$$

proving the claim. ∎

Using Claim 7.5.2.1 as well as the bound in (7.18), we see that

$$\begin{aligned}\|x' - d\| &\leq \|x - d\| + \|x' - x\| = \|x - d\| + \|L^W_{I\cup J}((w - x_I, x_J^-))\| \leq 4C_{\text{pf}}\|d^-\| + \sqrt{n}\mathrm{T}\|x^-_{I\cup J}\| \\ &\leq (4C_{\text{pf}} + \sqrt{n}\mathrm{T}\varepsilon)\|d^-\| \leq C^{\text{alg}}_{\text{pf}}\|d^-\|.\end{aligned}$$

Recursion on smaller subset: We show that $K \neq \emptyset$. If $d = d'$ then either $\|d^-\| \leq \|\Pi_{W^\perp}(d)\|$ or $\|d^-\| \leq \|d\|/(2C_{\text{pf}})$. If $\|d^-\| \leq \|\Pi_{W^\perp}(d)\|$ then,

$$\|x\|_\infty \geq \frac{\|x\|}{\sqrt{n}} \geq \frac{\|\Pi_{W^\perp}(d)\|}{\sqrt{n}} \geq \frac{\|d^-\|}{\sqrt{n}} \geq \frac{\|x^-\|}{\varepsilon\sqrt{n}} \geq \mathrm{T}\|x^-\|\,,$$

and if $\|d^-\| \leq \frac{\|d\|}{2C_{\text{pf}}}$, then $\|x - d\| \leq C_{\text{pf}}\|d^-\|$ by the call to Prox-Feas(W, d', M) and so

$$\|x\| \geq \|d\| - \|x - d\| > 2C_{\text{pf}}\|d^-\| - C_{\text{pf}}\|d^-\| = C_{\text{pf}}\|d^-\| \geq C_{\text{pf}}\varepsilon^{-1}\|x^-\| \geq \sqrt{n}\mathrm{T}\|x^-\|\,,$$

and hence $\|x\|_\infty \geq \mathrm{T}\|x^-\|$. The remaining case is $d' = \Pi_{W^\perp}(d)$. Then we have

$$\|x\|_\infty \geq \frac{\|x\|}{\sqrt{n}} \geq \frac{\|\Pi_{W^\perp}(d)\|}{\sqrt{n}} \geq \frac{\|x^-\|}{\varepsilon\sqrt{n}} \geq \mathrm{T}\|x^-\|\,.$$

Infeasibility certificate: Assume our algorithm resulted in a certificate of primal feasibility of $LP(\pi_I(W), x_I)$. Then by Fact 7.2.7, this certificate can canonically be lifted to a certificate of primal infeasibility of $LP(W, d)$. □

7.6 The optimization algorithm

in this section, we show how LP can be solved using an approximate LP solver. We present an algorithm that comprises an Inner (Algorithm 7.3)and an Outer Loop (Algorithm 7.2). The calls to the approximate LP solver will happen inside the Inner Loop.

The outer loop gives an algorithmic implementation of Corollary 3.5.4. The subroutine Inner-Loop$(W, d, c, M, \varepsilon)$ returns a solution (x, s), to System 7.11 for some $\varepsilon = \Theta(\text{poly}(M, n)^{-1})$. Note that Algorithm 7.3 in fact returns a solution to the stronger system Alg-Opt, but this is only required for the recursive calls of Algorithm 7.3. Given (x, s), we can now for ε small enough fix some primal variables to 0 via Corollary 3.5.4. Note that the role of the primal and dual in Corollary 3.5.4 and System 7.11 is reversed.

System 7.11. *Primal-Feasible-Apx-Complementary*

Data: A tuple $(W, d, c, M, \varepsilon)$, where $W \subseteq \mathbb{R}^n$ is a subspace, $c, d \in \mathbb{R}^n$, guess $M \geq 1$, and $\varepsilon > 0$.

find $x, s \in \mathbb{R}^n$

s.t.	$x \in W + d$	(Primal subspace constraint),	(7.19a)
	$s \in W^\perp + c$	(Dual subspace constraint),	(7.19b)
	$\|s_{\Lambda(s,x)}\| \leq \varepsilon \|c\|$	(Complementarity),	(7.19c)
	$x \geq 0$	(Primal non-negativity)	(7.19d)

The subroutine InnerLoop will be described in Section 7.6.2; we now state the running time.

Lemma 7.6.1. *Algorithm 7.3 (InnerLoop) is correct. The running time depends on the outcome and is as follows:*

Outcome	Running time
Alg-Opt$(W, d, c, M, \varepsilon_{\text{sub}})$	$\widetilde{O}\left(m\Psi(\mathbf{A}) \log(M{\varepsilon_{\text{sub}}}^{-1}) + m\mathcal{T}(\mathfrak{P}(\mathbf{A}))\right)$
$\mathfrak{F}(W, d)$	$\widetilde{O}\left(m\Psi(\mathbf{A}) \log(M{\varepsilon_{\text{sub}}}^{-1}) + m\mathcal{T}(\mathfrak{P}(\mathbf{A})) + \mathcal{T}(\mathscr{H}^{n\times m}_{\text{im}}(\mathbf{A}^\top))\right)$
$\mathfrak{C}(W, M)$	$\widetilde{O}\left(m\Psi(\mathbf{A}) \log(M{\varepsilon_{\text{sub}}}^{-1}) + m\mathcal{T}(\mathfrak{P}(\mathbf{A})) + \mathcal{T}(\mathscr{H}^{m\times n}_{\text{ker}}(\mathbf{A}))\right)$

The overall algorithm described in Section 7.6.1 repeatedly calls InnerLoop to set primal and dual variables to 0 according to Corollary 3.5.4, and recurses to lower dimensional subspaces. The final optimal solutions are obtained via calling the feasibility algorithm on both primal and dual side. Providing a certificate that $M < \kappa_W$ in case that the algorithm fails is non-trivial, but we will show how it can be done efficiently.

7.6.1 The Outer Loop

Consider an instance of System 1.2 and an estimate M on κ_W. We first use the feasibility algorithms and check if the primal system $x \in W + d, x \geq \mathbf{0}$ and the dual system $s \in W^\perp + c, s \geq \mathbf{0}$ are feasible. The dual feasible vector is in particular important as the input of InnerLoop requires a feasible dual vector. The overall algorithm is presented in Algorithm 7.2. We will maintain an index set $B \subseteq [n]$, initialized as $B = [n]$. We gradually remove indices from B. A removed index i has the property that due to proximity results (Corollary 3.5.4) there exist optimal dual solutions s^* such that $s^*_i > 0$. By complementary slackness we therefore have $x^*_i = 0$ in all primal optimal solutions.

Hence, at any stage, assuming that $M \geq \kappa_W$ we are guaranteed that there exists an optimal primal solution x^* with $x^*_{[n]\setminus B} = \mathbf{0}$. At every iteration, we have the index set $B \subseteq [n]$ of 'undecided indices' and the corresponding subspace $\bar{W} \subseteq \mathbb{R}^B$ that is $\bar{W} = W_B$.

The algorithm terminates when $\bar{c}$ is contained in $\bar{W}^\perp$. The remaining indices B are split up between B and N based on whether they are in the support of the current primal feasible solution $\tilde{x}$. Finally, we obtain the primal and dual solutions by solving feasibility problems on the subsets B and $N = [n] \setminus B$. 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 and construct a certificate of this outcome.

Algorithm 7.2: Optimization Algorithm

Input : An instance (W, d, c) of LP and guess $M \geq 1$.
Output : Either solution (x^*, s^*) optimal for $\mathrm{LP}(W, d, c)$, or a certificate in $\mathfrak{F}(W, d)$, $\mathfrak{F}(W^\perp, c)$ or $\mathfrak{C}(W, M)$.

1 $\spadesuit \leftarrow$ Feasibility-Algorithm(W, d, M)
2 **if** $\spadesuit \notin$ Primal(W, d) **then**
3 **return** $\spadesuit$
4 $\diamond \leftarrow$ Feasibility-Algorithm$(W^\perp, c, M)$
5 **if** $\diamond \notin$ Dual(W, c) **then**
6 **return** $\diamond$
7 $\bar{d} \leftarrow \spadesuit; \bar{c} \leftarrow \diamond$
8 $\bar{W} \leftarrow W$; $\tilde{x} \leftarrow \bar{d}$; $\tilde{s} \leftarrow \bar{c}$
9 $B \leftarrow [n]$
10 $\varepsilon_{\mathrm{sub}} \leftarrow \frac{1}{2} n^{-3} M^{-2}$
11 $t \leftarrow 1$
12 **while** $B \neq \emptyset$ **and** $\bar{c} \notin \bar{W}^\perp$ **do**
13 $\diamond \leftarrow$ InnerLoop$(\bar{W}, \bar{d}, \bar{c}, M, \varepsilon_{\mathrm{sub}})$
14 **if** $\diamond \notin$ Alg-Opt$(\bar{W}, \bar{d}, \bar{c}, M, \varepsilon_{\mathrm{sub}})$ **then**
15 **return** $\diamond$ // $\diamond$ is one of the certificates
16 **else**
17 $(\tilde{x}, \tilde{s}) \leftarrow \diamond$
18 **if** $\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| > \varepsilon_{\mathrm{sub}} \|\Pi_W(c)\|$ **then**
19 ▷ In this case we find a certificate that the guess was too small as the norm of basic solutions is too large on the support of $\tilde{s}$
20 $I \leftarrow \mathrm{supp}(\mathrm{ess}(\tilde{s}, \varepsilon_{\mathrm{sub}} \|\Pi_W(c)\|))$
21 $J \leftarrow [n] \setminus I$
22 **return** $(J, \tilde{s}_J)$ ▷ $\in \mathfrak{C}(W, M)$
23
24 $B^{(t)} \leftarrow B, W^{(t)} \leftarrow W, \tilde{s}^{(t)} \leftarrow \tilde{s}$ ▷ Needed only for analysis certificate generation
25 $t \leftarrow t + 1$
26 $B \leftarrow B \setminus \{ i \in I : |\tilde{s}_i| = \|\tilde{s}\|_\infty \}$
27 $\bar{W} \leftarrow \bar{W}_B, \bar{c} \leftarrow \tilde{s}_B, \bar{d} \leftarrow \tilde{x}_B$
28 $B \leftarrow \mathrm{supp}(\tilde{x})$
29 $N \leftarrow [n] \setminus B$
30 // Find primal feasible solution supported on B
31 $\hat{d} \leftarrow \left[d - L_N^W(d_N) \right]_B$
32 $\diamond \leftarrow$ Feasibility-Algorithm$(W_B, \hat{d}, M)$
33 **if** $\diamond \notin$ Primal$(W_B, \hat{d})$ **then**
34 // $\diamond$ is in $\mathfrak{C}(W, M)$ as primal infeasibility would have been detected in the last call to InnerLoop
35 **return** $\diamond$
36 **else**
37 $x^* \leftarrow (\diamond, \mathbf{0}_N)$
38 // Find dual feasible solution supported on N
39 $\hat{c} \leftarrow \left[c - L_B^{W^\perp}(c_B) \right]_N$
40 $\diamond \leftarrow$ Feasibility-Algorithm$((W^\perp)_N, \hat{c}, M)$
41 **if** $\diamond \in$ Dual$((W^\perp)_N, \hat{c})$ **then**
42 $s^* \leftarrow (\mathbf{0}_B, \diamond)$
43 **return** (x^*, s^*)
44 **else if** $\diamond \in \mathfrak{C}((W^\perp)_N, M)$ **then**
45 **return** $\diamond$
46 **else**
47 // this means $\diamond \in \mathfrak{F}((W^\perp)_N, \hat{c})$
48 $\spadesuit \leftarrow$ Feasibility-Algorithm$(W^\perp, c)$
49 **if** $\spadesuit \notin$ Dual$(W^\perp, c)$ **then**
50 **return** $\spadesuit$
51 **else**
52 $\tilde{c} \leftarrow \spadesuit$
53 Use $\tilde{c}$ and the dual vectors $\tilde{s}^{(1)}, \ldots, \tilde{s}^{(t-1)}$ to find a certificate $z \in \mathfrak{C}(W, M)$
54 **return** z

Theorem 7.6.2. *Algorithm 7.2 is correct. This means, for data (W, d, c) it finds a primal-dual optimal solution (x^*, s^*), a Farkas certificate in $\mathfrak{F}(W, d)$ or $\mathfrak{F}(W^\perp, c)$ or concludes that the guess M was to small and certifies this with an element in $\mathfrak{C}(W, M)$.*

The running time depends on the outcome and is as follows:

Outcome	Running time
optimal in $\mathrm{LP}(W, d, c)$	$\widetilde{O}\big(mn(\Psi(\mathbf{A}) \log(M) + \mathcal{T}(\mathfrak{P}(\mathbf{A})) + \mathcal{T}(\mathrm{cl}(\mathbf{A})))\big)$
$\mathfrak{F}(W, d)$	$\widetilde{O}\big(m\Psi(\mathbf{A}) \log(M) + m\mathcal{T}(\mathfrak{P}(\mathbf{A})) + m\mathcal{T}(\mathrm{cl}(\mathbf{A})) + \mathcal{T}(\mathscr{H}_{\mathrm{im}}(\mathbf{A}^\top))\big)$
$\mathfrak{F}(W^\perp, c)$	$\widetilde{O}\big(n\Psi(\mathbf{A}) \log(M) + n\mathcal{T}(\mathfrak{P}(\mathbf{A})) + n\mathcal{T}(\mathrm{cl}(\mathbf{A})) + \mathcal{T}(\mathscr{H}_{\ker}(\mathbf{A}))\big)$
$\mathfrak{C}(W, M)$	$\widetilde{O}\big(mn\Psi(\mathbf{A}) \log(M) + mn\mathcal{T}(\mathfrak{P}(\mathbf{A})) + mn\mathcal{T}(\mathrm{cl}(\mathbf{A})) + \mathcal{T}(\mathscr{H}_{\mathrm{im}}(\mathbf{A}^\top))\big)$

Proof. The main part of the poof is recognizing that the variables that are removed from B in Line 26 belong to the set N^* of variables for which $x^*_{N^*} = \mathbf{0}$ for all optimal solutions x^* of $\mathrm{LP}(W, d, c)$.

Note that the guarantee of InnerLoop is that

$$\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| \leq n\varepsilon_{\text{sub}}\widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\text{sub}}\chi_{W^\perp}(c)) \leq 2n\varepsilon_{\text{sub}}\bar{\chi}_{W^\perp}\|\Pi_W(c)\|\,. \tag{7.20}$$

In case that the `if`-condition in line 18 evaluates to `true`, we see via (7.20) that

$$\kappa_W \overset{\text{Thm. 3.3.8}}{\geq} \frac{\bar{\chi}_{W^\perp}}{n} \geq \frac{\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\|}{2n^2\varepsilon_{\text{sub}}\|\Pi_W(c)\|} > \frac{2n^2M\varepsilon_{\text{sub}}\|\Pi_W(c)\|}{2n^2\varepsilon_{\text{sub}}\|\Pi_W(c)\|} = M\,. \tag{7.21}$$

We can certify this by the fact that all basic solutions on the last recursive call to InnerLoop have large cost. This can be recovered by defining the set I in line 20 and returning a corresponding lifting certificate.

So, from on assume that the `if` condition in line 18 evaluated to `false`, i.e., $\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| < 2n^2M\varepsilon_{\text{sub}}\|\tilde{s}\|$. We then move on and remove elements from B in line 26. In case that $M \geq \kappa_W$ this conclusion is correct and follows automatically by the proximity result Corollary 3.5.4 as

$$\begin{aligned}\|\tilde{s}\|_\infty \geq \frac{1}{\sqrt{n}}\|\tilde{s}\| \geq \frac{1}{\sqrt{n}}\|\Pi_W(c)\| \overset{(7.20)}{>} \frac{1}{2n^{1.5}\varepsilon_{\text{sub}}\bar{\chi}_{W^\perp}}\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| \overset{\text{Thm. 3.3.8}}{\geq} \frac{1}{2n^{2.5}\varepsilon_{\text{sub}}\kappa_W}\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| \\ \geq M\sqrt{n}\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\| \geq \kappa_W\|\tilde{s}_{\Lambda(\tilde{s},\tilde{x})}\|_1\,,\end{aligned} \tag{7.22}$$

and so the proximity result gives $x^*_i = 0$ for all i such that $\tilde{s}_i = \|\tilde{s}\|_\infty$.

However, in case that we underestimate κ_W, i.e., $M < \kappa_W$, we might erroneously remove elements from B in line 26 and this may remain undetected until later, when we fail to construct a dual feasible solution on N.

But $M < \kappa_W$ might not be detected in the `while`-loop in line 12, but only much later, when we detect infeasibility of the final dual system $\text{Dual}((W^\perp)_N, \hat{c})$.

We now sketch how a certificate of in $\mathfrak{C}(W, M)$ can be found nonetheless, by using the constructive proximity results in Section 7.3 in conjunction with the dual solutions $\tilde{s}^{(1)}, \ldots, \tilde{s}^{(t)}$ returned by the calls to InnerLoop as well as the terminal dual solution $u^{(t+1)} := \mathbf{0}_B$. First, we find the unique index $i \in [t]$ such that the system $s \in W_{B^{(i)}} + c_{B^{(i)}}, s_B = \mathbf{0}, s \geq \mathbf{0}$ is infeasible and the system $s \in W_{B^{(i+1)}} + c_{B^{(i+1)}}, s_B = \mathbf{0}, s \geq \mathbf{0}$ is feasible. Such an i must exist as the corresponding system for B for $i = t+1$ is feasible, while the system for $i = 0$ is not. This property is monotone in the sense that if the system corresponding to i is feasible, so is the system for all $j \geq i$. Therefore, we are able to find the index i via binary search in $\log(n)$ calls to the feasibility algorithm for the dual systems.

Let now $\bar{s}$ be such a solution to the system corresponding to $i+1$. Note that the final primal solution x is supported on B, hence $\left\langle x^*_{B^{(i+1)}}, \bar{s} \right\rangle = \mathbf{0}$. Let us extend $\bar{s}$ arbitrarily to an element $\check{s} \in W_{B^{(i)}} + c_{B^{(i)}}$, i.e., $\check{s}_{B^{(i+1)}} = \bar{s}$. Then, we still have $\left\langle x^*_{B^{(i)}}, \check{s} \right\rangle = \mathbf{0}$. We now apply the constructive Hoffman proximity theorem $\mathscr{H}$ to $\check{s} - \tilde{s}^{(i)}$, where we use appropriate lower and upper bounds such that all the augmentations we perform on $\tilde{s}^{(i)}$ are either increasing negative coordinates of $\tilde{s}^{(i)}$ or are improving the objective with respect to $x^{(i)}$. We must have that this call to $\mathscr{H}$ results in a certificate in $C(W, M)$. The alternative would be that the augmentation succeeded. But then, by choice of the coordinates in $B^{(i)} \setminus B^{(i+1)}$ we would have that the resulting vector $\hat{s}$ would be non-negative on $B^{(i)} \setminus B^{(i+1)}$, hence it would fulfill $\hat{s} \geq \mathbf{0}$. Further, its objective value is no worse than the objective value of $\check{s}$, hence $\langle x^*, \hat{s} \rangle = 0$ and so $\hat{s}_B = \mathbf{0}$. In particular, $\hat{s}$ is feasible to $s \in W_{B^{(i)}} + c_{B^{(i)}}, s_B = \mathbf{0}, s \geq \mathbf{0}$. This is a contradiction to the assumption that the system is infeasible. □

By proving Theorem 7.6.2, we also proved the special case Theorem 7.1.2.

7.6.2 The Inner Loop

Proof of Lemma 7.6.1. The proof will be a generalization of the proof of Theorem 7.5.2. We assume the case that the algorithm returned $(x', s') \in \text{Alg-Opt}(W, d, c, M, \varepsilon_{\text{sub}})$. The other scenarios are handled as *bad case* scenarios in the analysis. We use the variable names in the algorithm and begin by showing the primal proximity constraint (7.41d). For this, we first prove a proximity claim regarding x.

Claim 7.6.2.1. *$\|x - d\| \leq 4C_{\text{po}}\|d_{\Lambda(d,c)}\|$ and $\|x^-\| \leq \varepsilon\|d_{\Lambda(d,c)}\|$.*

Proof. If $\bar{d} = d$, then from the feasibility of (x, s) to $\text{Prox-Opt}(W, \bar{d}, c, \check{c}, M, \varepsilon)$, we have $\|x - d\| \leq C_{\text{po}}\|d_{\Lambda(d,c)}\|$ and $\|x^-\| \leq \varepsilon\|d_{\Lambda(d,c)}\|$, so we are done.

In the case $\bar{d} \neq d$, we have $\bar{d} = \Pi_{W^\perp}(d)$, $\|d_{\Lambda(d,c)}\| > \|\Pi_{W^\perp}(d)\|$ and $\|d_{\Lambda(d,c)}\| > \frac{\|d\|}{2C_{\text{po}}}$. So

$$\begin{aligned}
\|x - d\| &\leq \|x - \Pi_{W^\perp}(d)\| + \|\Pi_{W^\perp}(d)\| + \|d\| \\
&\leq C_{\text{po}}\|\Pi_{W^\perp}(d)\| + \|\Pi_{W^\perp}(d)\| + 2C_{\text{po}}\|d_{\Lambda(d,c)}\| \\
&\leq (C_{\text{po}} + 1)\|\Pi_{W^\perp}(d)\| + 2C_{\text{po}}\|d_{\Lambda(d,c)}\| \\
&\leq (2C_{\text{po}} + 2C_{\text{po}})\|d_{\Lambda(d,c)}\| \\
&\leq 4C_{\text{po}}\|d_{\Lambda(d,c)}\|\,, \quad \text{and} \\
\|x^-\| &\leq \varepsilon\|\Pi_{W^\perp}(d)^-\| \leq \varepsilon\|\Pi_{W^\perp}(d)\| \leq \varepsilon\|d_{\Lambda(d,c)}\|\,,
\end{aligned}$$

proving the claim. ∎

If $\bar{d} = d$, then $\|x \circ s\| \leq \varepsilon\|d_{\Lambda(d,c)}\|\|c\|$ by (7.42c) in System 7.13. Else, $\bar{d} = \Pi_{W^\perp}(d)$ and $\Pi_{W^\perp}(d) \geq \|d_{\Lambda(d,c)}\|$, implying $\|x \circ s\| \leq \varepsilon\|\Pi_{W^\perp}(d)\|\|c\| \leq \|d_{\Lambda(d,c)}\|\|c\|$ as well. Further, we have for all $j \in \text{supp}(\tilde{s})$ that $s_j \geq n^{-1/2}\varepsilon_{\text{sub}}\|\check{c}\|$. Hence, for all $j \in \text{supp}(\tilde{s})$ we have

$$x_j \leq \frac{\|x \circ \tilde{s}\|}{\tilde{s}_j} \leq \frac{\|x \circ s\|}{\tilde{s}_j} \leq \frac{\sqrt{n}\varepsilon\|d_{\Lambda(d,c)}\|\|\check{c}\|}{\varepsilon_{\text{sub}}\|\check{c}\|} = \frac{\sqrt{n}\varepsilon\|d_{\Lambda(d,c)}\|}{\varepsilon_{\text{sub}}}\,, \tag{7.23}$$

and so with the equation above and Claim 7.6.2.1

$$\|x_{\Lambda(x,\tilde{s})}\| \leq \|x^-\| + \|x_{\text{supp}(\tilde{s})}\| \overset{(7.23)}{\leq} \varepsilon\|d_{\Lambda(d,c)}\| + \frac{n\varepsilon}{\varepsilon_{\text{sub}}}\|d_{\Lambda(d,c)}\| \leq \frac{2\varepsilon n}{\varepsilon_{\text{sub}}}\|d_{\Lambda(d,c)}\|\,. \tag{7.24}$$

We can assume that $\|w\|_\infty \leq M\|(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}})\|_1$ as otherwise we would have concluded with a certificate in $\mathfrak{C}(W, M)$ on Line 30.

Therefore, by assumption,

$$\begin{aligned}
\|w\| &\leq \sqrt{n}\left\|L^W_{I\cup J}\left(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}}\right)\right\|_\infty \\
&\leq M\sqrt{n}\|(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}})\|_1 \\
&\leq M\sqrt{n}\left(C^{\text{alg}}_{\text{po}}\|x_{\Lambda(x_I,\tilde{s}_I)}\|_1 + \|x_{\Lambda(x_J,\tilde{s}_J)}\|_1\right) \\
&\leq M\sqrt{n}C^{\text{alg}}_{\text{po}}\|x_{\Lambda(x,\tilde{s})}\|_1 \\
&\leq Mn C^{\text{alg}}_{\text{po}}\|x_{\Lambda(x,\tilde{s})}\| \\
&\overset{(7.24)}{\leq} \frac{2\varepsilon n^2 M C^{\text{alg}}_{\text{po}}}{\varepsilon_{\text{sub}}}\|d_{\Lambda(d,c)}\|\,,
\end{aligned} \tag{7.25}$$

where we used the feasibility of (v,u) to Alg-Opt$(\pi_I(W), x_I, \tilde{s}_I, M, \varepsilon_{\text{sub}})$ and that $\tilde{J} \subseteq \Lambda(x,\tilde{s})$ by definition. UsingClaim 7.6.2.1 and the inequlaity above we can deduce that

$$\begin{aligned}
\|x' - d\| &\leq \|x - d\| + \|x' - x\| && (7.26)\\
&\leq \|x - d\| + \|w\| && (7.27)\\
&\overset{(7.25)}{\leq} \left(4C_{\text{po}} + \frac{2\varepsilon n^2 M}{\varepsilon_{\text{sub}}} C_{\text{po}}^{\text{alg}}\right) \|d_{\Lambda(d,c)}\| && (7.28)\\
&\leq C_{\text{po}}^{\text{alg}} \|d_{\Lambda(d,c)}\|\,, && (7.29)
\end{aligned}$$

showing (7.41d).

Recursion on smaller subset: We show that $K \neq \emptyset$. If $\bar{d} = d$ then either $\|d_{\Lambda(d,c)}\| \leq \|\Pi_{W^\perp}(d)\|$ or $\|d_{\Lambda(d,c)}\| \leq \|d\|/(2C_{\text{po}})$. If $\|d_{\Lambda(d,c)}\| \leq \|\Pi_{W^\perp}(d)\|$ then,

$$\|x\|_\infty \geq \frac{\|x\|}{\sqrt{n}} \geq \frac{\|\Pi_{W^\perp}(d)\|}{\sqrt{n}} \geq \frac{\|d_{\Lambda(d,c)}\|}{\sqrt{n}} \overset{(7.24)}{\geq} \frac{\varepsilon_{\text{sub}}}{2\varepsilon n^{1.5}} \|x_{\Lambda(x,s)}\| \geq \mathrm{T}\|x_{\Lambda(x,s)}\|\,,$$

and if $\|d_{\Lambda(d,c)}\| \leq \frac{\|d\|}{2C_{\text{po}}}$, then $\|x - \bar{d}\| \leq C_{\text{po}}\|d_{\Lambda(d,c)}\|$ by feasibility to Prox-Opt$(W, \bar{d}, c, \check{c}, M, \varepsilon)$ and so

$$\|x\| \geq \|d\| - \|x-d\| \geq 2C_{\text{po}}\|d_{\Lambda(d,c)}\| - C_{\text{po}}\|d_{\Lambda(d,c)}\| = C_{\text{po}}\|d_{\Lambda(d,c)}\| \overset{(7.24)}{\geq} C_{\text{po}} \frac{\varepsilon_{\text{sub}}}{2\varepsilon n} \|x_{\Lambda(x,s)}\| \geq \sqrt{n}\mathrm{T}\|x_{\Lambda(x,s)}\|\,,$$

and hence $\|x\|_\infty \geq \mathrm{T}\|x^-\|$. The remaining case is $\bar{d} = \Pi_{W^\perp}(d)$. Then we have $\|x^-\| \leq \varepsilon\|\Pi_{W^\perp}(d)\|$ and

$$x_j \leq \frac{\|x \circ \tilde{s}\|}{\tilde{s}_j} \leq \frac{\|x \circ s\|}{\tilde{s}_j} \leq \frac{\sqrt{n}\varepsilon\|\Pi_{W^\perp}(d)\|\|\check{c}\|}{\varepsilon_{\text{sub}}\|\check{c}\|} = \frac{\sqrt{n}\varepsilon\|\Pi_{W^\perp}(d)\|}{\varepsilon_{\text{sub}}}\,. \tag{7.30}$$

and therefore

$$\|x_{\Lambda(x,\tilde{s})}\| \leq \|x^-\| + \|x_{\text{supp}(\tilde{s})}\| \overset{(7.23)}{\leq} \varepsilon\|\Pi_{W^\perp}(d)\| + \frac{n\varepsilon}{\varepsilon_{\text{sub}}}\|\Pi_{W^\perp}(d)\| \leq \frac{2\varepsilon n}{\varepsilon_{\text{sub}}}\|\Pi_{W^\perp}(d)\|\,. \tag{7.31}$$

and so

$$\|x\|_\infty \geq \frac{\|x\|}{\sqrt{n}} \geq \frac{\|\Pi_{W^\perp}(d)\|}{\sqrt{n}} \geq \frac{\varepsilon_{\text{sub}}}{2\varepsilon n^{1.5}}\|x_{\Lambda(x,\tilde{s})}\| \leq \mathrm{T}\|x_{\Lambda(x,\tilde{s})}\|\,.$$

In all cases we therefore conclude that $\|x\|_\infty \geq \mathrm{T}\|x^-\|$. Hence $K \neq \emptyset$ and $|I| \leq n-1$.

We now show that x' is well-defined. This requires that w is well-defined, which in turn requires that $(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}}) \in \pi_{I\cup J}(W)$ (Line 28). By definition of J, we have that $\pi_J(W) = \mathbb{R}^J$ and $\pi_{I\cup J}(W) = \pi_I(W) \times \mathbb{R}^J$. As v is the primal variable of the recursive call to Alg-Opt we have that $v \in \pi_I(W) + x_I$ and so $v - x_I \in \pi_I(W)$, implying $(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}}) \in \pi_I(W) \times \mathbb{R}^J = \pi_{I\cup J}(W)$. The containment $x' \in W + d$ is now immediate as $L^W_{I\cup J}\left(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}}\right) \in W$. Hence, we have (7.41a).

For the dual subspace constraint (7.41b) note that $u \in (W^\perp)_I + \tilde{s}_I$ and so $s' = (\tilde{s}_{J\cup K}, u) + s - \tilde{s} \in W^\perp + \tilde{s} + s - \tilde{s} = W^\perp + s = W^\perp + c$.

Let us verify the primal non-negativity constraint (7.41e). For $i \in I$ we have $x'_i = x_i + (v_i - x_i) = v_i \geq 0$. For $j \in \tilde{J}$ we have $x'_j = x_j - x_j = 0$ and for $j \in J \setminus \tilde{J}$ we have $x'_j = x_j \geq 0$. It remains to show the non-negativity on K. By definition of v and the constraint (7.41d) we have that $\|v - x_I\| \leq C_{\text{po}}^{\text{alg}}\|x_{\Lambda(x_I,\tilde{s}_I)}\|$.

Hence, for $i \in K$ we have $x'_i \geq x_i - \|L^W_{I\cup J}\Big((v - x_I, x^-_J)\Big)\|_\infty \geq (\mathrm{T} - Mn\mathrm{C}^{\mathrm{alg}}_{\mathrm{po}})\|x_{\Lambda(x,s)}\| = 0$, proving (7.41e).

We now show (7.41c). Note that $\tilde{s} \geq \mathbf{0}$ as $\|s^-\|_\infty \leq \|s^-\| \leq \varepsilon_{\mathrm{sub}}\|\check{c}\|$ and so $\tilde{s}_i = 0$ for all $i \in [n]$ with $s_i \leq 0$. Further note that $x'_{\tilde{J}} = \mathbf{0}$ and $\tilde{s}_{K\cup(J\setminus\tilde{J})} = \mathbf{0}$ by construction. Hence,

$$\|\tilde{s}_{\Lambda(s',x')\cap(KUJ)}\| = 0\,, \tag{7.32}$$

Note that $\pi_J\big((W^\perp)_{J\cup I}\big) = \{\mathbf{0}\}$ because $J = \mathrm{cl}(K) \setminus K$, so we can apply Proposition 7.2.3.
Further, note that $\|s - \tilde{s}\| \leq \varepsilon_{\mathrm{sub}}\|\check{c}\| \leq \varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)$. And so

$$\begin{aligned}
\widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)) &= \widetilde{\chi}_{W^\perp}(s, n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)) \\
&\geq \widetilde{\chi}_{W^\perp}(s - (s - \tilde{s}), n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c) - \|s - \tilde{s}\|) \\
&\geq \widetilde{\chi}_{W^\perp}(\tilde{s}, (n-1)\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)) \\
&\geq \widetilde{\chi}_{(W^\perp)_{I\cup J}}\big(\tilde{s}_{I\cup J}, (n-1)\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)\big) \\
&\geq \widetilde{\chi}_{(W^\perp)_I}(\tilde{s}_I, (n-1)\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c))\,,
\end{aligned} \tag{7.33}$$

where the third inequality uses $\tilde{s}_K = \mathbf{0}$ with Proposition 7.2.1 and the last inequality follows from Proposition 7.2.3 where we use $\pi_J\big((W^\perp)_{J\cup I}\big) = \{\mathbf{0}\}$. Hence,

$$\begin{aligned}
\|s'_{\Lambda(s',x')}\| &\leq \Big\|\big(u, \tilde{s}_{J\cup K}\big)_{\Lambda(s',x')}\Big\| + \|\tilde{s} - s\| && (7.34)\\
&\leq \|u_{\Lambda(u,x')\cap I}\| + \|\tilde{s}_{\Lambda(s',x')\cap(K\cup J)}\| + \|\tilde{s} - s\| && (7.35)\\
&\leq \|u_{\Lambda(u,v)}\| + \|\tilde{s}_{\Lambda(s',x')\cap(K\cup J)}\| + \|\tilde{s} - s\| && (7.36)\\
&\leq |I|\varepsilon_{\mathrm{sub}}\widetilde{\chi}_{(W^\perp)_I}(s_I, |I|\varepsilon_{\mathrm{sub}}\chi_{(W^\perp)_I}(s_I)) + 0 + \varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c) && (7.37)\\
&\leq |I|\varepsilon_{\mathrm{sub}}\widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)) + \varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c) && (7.38)\\
&\leq (|I| + 1)\varepsilon_{\mathrm{sub}}\widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c)) && (7.39)\\
&\leq n\varepsilon_{\mathrm{sub}}\widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\mathrm{sub}}\chi_{W^\perp}(c))\,. && (7.40)
\end{aligned}$$

Here, (7.37) follows from the feasibility of (v, u) to the recursive system Alg-Opt$(\pi_I(W), x_I, \tilde{s}_I, M, \varepsilon_{\mathrm{sub}})$ and (7.32). Further, (7.38) follows from (7.33). (7.39) follows from $\chi_{W^\perp}(c) \leq \widetilde{\chi}_{W^\perp}(c, 0)$.

Infeasibility certificate. Assume our algorithm resulted in a certificate of primal feasibility of LP$(\pi_I(W), x_I)$. Then by Fact 7.2.7, this certificate can canonically be lifted to a certificate of primal infeasibility of LP(W, d).

Running time. The runtime is dominated by the m recursive calls to InnerLoop and the calls to the Proximal-Opt-Oracle as well as the projection that we perform. Note that the lift in line 28 boils down to another projection. Hence, the runtime claim follows. □

Algorithm 7.3: InnerLoop

Input : $W \subseteq \mathbb{R}^n$, $c \in \mathbb{R}^n_+$, $d \in \mathbb{R}^n$, $M \geq 1$, error $\varepsilon_{\text{sub}} > 0$.
Output : A solution to Alg-Opt$(W, d, c, M, \varepsilon_{\text{sub}})$, an element in $\mathfrak{F}(W, d)$ or an element in $\mathfrak{C}(W, M)$.

if $(d, c) \in$ Alg-Opt(W, d, c, M) **then**
  return (d, c)
$\mathrm{T} \leftarrow Mn C_{\text{po}}^{\text{alg}}$, $\varepsilon \leftarrow \frac{\varepsilon_{\text{sub}}}{n^3 \mathrm{T}}$
$\bar{d} \leftarrow d$
if $\|d_{\Lambda(d,c)}\| > \max\left\{\|\Pi_{W^\perp}(d)\|, \frac{\|d\|}{2C_{\text{po}}}\right\}$ **then**
  $\bar{d} \leftarrow \Pi_{W^\perp}(\bar{d})$
$S \leftarrow \text{supp}(c)$
$\check{c} \leftarrow \Pi_{(W_S \times \mathbb{R}^{[n]\setminus S})}(c)$
$\diamond \leftarrow$ Proximal-Opt-Oracle$(W, \bar{d}, c, \check{c}, M, \varepsilon)$
switch $\diamond$ **do**
  case *(i)* $\diamond \in$ Prox-Opt$(W, \bar{d}, c, \check{c}, M, \varepsilon)$ **do**
    go to 18
  case *(ii)* $\diamond \in \mathfrak{F}(W, d)$ **do**
    return $\diamond$
  // Note that (iii) can not occur as a feasible dual is part of the input
  case *(iv)* $\diamond \in \mathfrak{C}(W, M)$ **do**
    return $\diamond$
$(x, s) \leftarrow \diamond$
$\tilde{s} \leftarrow \text{ess}\left(s, \frac{\varepsilon_{\text{sub}}}{\sqrt{n}}\|\check{c}\|\right)$
$K \leftarrow \left\{ i \in [n] : x_i \geq \mathrm{T}\|x_{\Lambda(x,\tilde{s})}\| \right\}$
$J \leftarrow \text{cl}(K) \setminus K$
$I \leftarrow [n] \setminus \text{cl}(K)$
$\spadesuit \leftarrow$ InnerLoop$(\pi_I(W), x_I, \tilde{s}_I, M, \varepsilon_{\text{sub}})$
switch $\spadesuit$ **do**
  case $\spadesuit \in$ Alg-Opt$(\pi_I(W), x_I, \tilde{s}_I, M)$ **do**
    $(v, u) \leftarrow \spadesuit$
    $\tilde{J} \leftarrow \Lambda(x, \tilde{s}) \cap J$
    $w \leftarrow L^W_{I\cup J}(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}})$
    if $\|w\|_\infty > M\|(v - x_I, -x_{\tilde{J}}, \mathbf{0}_{J\setminus\tilde{J}})\|_1$ **then**
      return generated certificate in $\mathfrak{C}(W, M)$
    $x' \leftarrow x + w$
    $s' \leftarrow (u, \tilde{s}_{J\cup K}) + s - \tilde{s}$
    return (x', s') // feasible to Alg-Opt(W, d, c, M)
  case $\spadesuit \in \mathfrak{F}(\pi_I(W), x_I)$ **do**
    return lifted certificate in $\mathfrak{F}(W, d)$ // using Fact 7.2.7
  case $\spadesuit \in \mathfrak{C}(\pi_I(W), M)$ **do**
    return lifted certficate in $\mathfrak{C}(W, M)$

System 7.12. *Alg-Opt*

Data: A tuple $(W, d, c, M, \varepsilon_{\text{sub}})$, where $W \subseteq \mathbb{R}^n$ is a subspace, $c, d \in \mathbb{R}^n$, $c \geq \mathbf{0}$, guess $M \geq 1$ and a dual error ε_{sub}.

find $x, s \in \mathbb{R}^n$

s.t.	$x \in W + d$	(primal subspace constraint),	(7.41a)
	$s \in W^\perp + c$	(dual subspace constraint),	(7.41b)
	$\|s_{\Lambda(s,x)}\| \leq n\varepsilon_{\text{sub}} \widetilde{\chi}_{W^\perp}(c, n\varepsilon_{\text{sub}} \chi_{W^\perp}(c))$	(apx. compl., dual feas.),	(7.41c)
	$\|x - d\| \leq C_{\text{po}}^{\text{alg}} \|d_{\Lambda(d,c)}\|$	(Primal proximity constraint),	(7.41d)
	$x \geq \mathbf{0}$	(Primal Non-negativity)	(7.41e)

where $C_{\text{po}}^{\text{alg}} := 8C_{\text{po}} = \text{poly}(M, n)$.

Note that the factor n shows up on the right-hand-side of Equation (7.41c) because we need to provide some slack for the recursive calls to work where the number of variables is smaller of n.

System 7.13. *Prox-Opt*

Data: A tuple $(W, d, c, \check{c}, M, \varepsilon)$, where $W \subseteq \mathbb{R}^n$ is a subspace, $c, \check{c}, d \in \mathbb{R}^n$, $c \geq \mathbf{0}$, guess $M \geq 1$, and $\varepsilon > 0$ s.t. $c - \check{c} \in W^\perp$ and $\operatorname{supp}(\check{c}) \subseteq \operatorname{supp}(c)$.

find $x, s \in \mathbb{R}^n$

s.t.

$$x \in W + d \quad \text{(Primal Subspace constraint)}, \tag{7.42a}$$
$$s \in W^\perp + c \quad \text{(Dual Subspace constraint)}, \tag{7.42b}$$
$$\|x \circ s\| \leq \varepsilon \|d_{\Lambda(d,c)}\| \|\check{c}\| \quad \text{(Apx. complementarity)}, \tag{7.42c}$$
$$\|x - d\| \leq C_{\text{po}} \|d_{\Lambda(d,c)}\| \quad \text{(Primal proximity)}, \tag{7.42d}$$
$$\|x^-\| \leq \varepsilon \|d_{\Lambda(d,c)}\| \quad \text{(Primal near non-negativity)}, \tag{7.42e}$$
$$\|s^-\| \leq \varepsilon \|\check{c}\| \quad \text{(Dual near non-negativity)}, \tag{7.42f}$$
$$\|s\| \leq C_{\text{o}} \|\check{c}\| \quad \text{(Dual norm bound)} \tag{7.42g}$$

where $C_{\text{po}} := 8 C_{\text{o}} M n^{3/2} = \operatorname{poly}(n, M)$.

Oracle 7.2. *Proximal-Opt-Oracle*

Input: Data $(W, d, c, \check{c}, M, \varepsilon)$ for Prox-Opt (System 7.13).

Output: One of the following:

(i) A solution (x, s) to Prox-Opt$(W, d, c, \check{c}, M, \varepsilon)$,

(ii) A Farkas certificate $y \in \mathfrak{F}(W, d)$,

(iii) A Farkas certificate $y \in \mathfrak{F}(W^\perp, c)$,

(iv) A lifting certificate $(I, p) \in \mathfrak{C}(W, M)$.

The implementation of Oracle 7.2 is given in Section 7.7.2, using Theorem 7.4.7.

Lemma 7.6.3 (Proof on p. 189). *There exists an implementation of* Proximal-Opt-Oracle *(Oracle 7.2) for data* $(W, d, c, \check{c}, M, \varepsilon)$ *with following running times, depending on the outcomes:*

Outcome	Running time
(i)	$\widetilde{O}\left(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\right)$
(ii), (iii), (iv)	$\widetilde{O}\left(\Psi(\mathbf{A}) \log(M\varepsilon^{-1})\right) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))$

7.7 Implementation of oracles and subroutines

All of the fast LP solvers we are going to use in a black-box manner are based on interior point methods. The papers [Bra20; CLS19; JSWZ21] use the standard log barrier, while the papers [BTSS20; LS19] are using a different central path based on leverage scores, or more generally Lewis weights. Another new result shows faster running times for matrices with small treewidth [DLY21]. Interior point methods rely on a primal-dual centered initialization point which is not easy to achieve for the original problem. Therefore, all the above-mentioned papers use an auxiliary

system for initialization, that has the property that near-optimal solutions to this system can be rounded to near-optimal and near-feasible solutions of the original system. We could apply their main theorems here, but we feel that the reader would benefit more from a general exposition of the main theorems in of the above-mentioned papers. The discussion in this paragraph is summarized in the following theorem:

Theorem 7.7.1 ([Bra+21; Bra20; DLY21; JSWZ21; LS19]). *Consider System 1.1 for $\mathbf{A} \in \mathbb{R}^{m\times n}$ with $\mathrm{rk}(\mathbf{A}) = m$. Assume that primal and dual are feasible and furthermore a primal-dual feasible pair $(x_0, s_0) \in \mathbb{R}^{n+n}$ with $\|x_0 \circ s_0 - \mu_0 \mathbf{1}_n\| < 0.5$ is given for some $\mu_0 > 0$. Then a primal-dual feasible pair $(x, s) \in \mathbb{R}^{n+n}$ with duality gap $\langle x, s\rangle \leq \mu$ can be found in time*

(1) In $\widetilde{O}((mn + m^{2.5}) \log(\mu_0/\mu))$ expected running time [Bra+21].

(2) In $\widetilde{O}\left(n^{\omega} \log(\mu_0/\mu)\right)$ deterministic running time, assuming $\omega \geq 13/6$ [Bra20]. The same expected running time is achievable assuming $\omega \geq 2 + 1/18$ [JSWZ21].

(3) In $\widetilde{O}\left((\mathrm{nnz}(\mathbf{A}) + m^2)\sqrt{m} \log(\mu_0/\mu)\right)$ expected running time [LS19], where $\mathrm{nnz}(\mathbf{A})$ denotes the number of nonzero entries in $\mathbf{A}$.

(4) In $\widetilde{O}(n\,\mathrm{tw}(\mathbf{A})^2 \log(\mu_0/\mu))$ expected running time, where $\mathrm{tw}(\mathbf{A})$ is the treewidth of the matrix $\mathbf{A}$.

Note that the all the implementations of Oracle 7.3 are interior point methods, which heavily makes use of the given initial point (x_0, s_0). On the other hand, other implementations of the oracle could be based on cutting plane methods or combinatorial algorithms that could just ignore the initially given solution (x_0, s_0).

Oracle 7.3. *Blackbox-Approx-Solver*

Input: An instance of $\mathrm{LP}(\mathbf{A}, d, c)$ with feasible primal-dual pair (x_0, s_0) such that $\|x_0 \circ s_0 - \mu_0 \mathbf{1}\| \leq \frac{1}{2}$ for some $\mu_0 \geq 0$ and $\mu > 0$.

Output: A feasible solution (x, s) to $\mathrm{LP}(\mathbf{A}, d, c)$ with $\langle x, s\rangle \leq \mu$.

7.7.1 Proofs of Theorem 7.4.6 and Theorem 7.4.7

We create the auxiliary System 7.14 which for a correct guess M (i.e., $M \geq \kappa_W$) maintains optimal solutions of the original instance, and which can be initialized close to the central path of the standard log-barrier. Note that κ and $\bar{\chi}$ can be significantly larger in System 7.14 than in the original System 1.1. Fortunately this does not matter, as we only aim for approximate solutions to the system. Importantly, this new system preserves sparsity and rank patterns, which is necessary to achieve the runtimes claimed in Theorem 7.7.1 under the mild Assumption 7.4.5. We state System 7.14 in both matrix formulation and subspace formulation. The matrix formulation confirms that we can use Assumption 7.4.5. The subspace formulation will be used in the remainder of this section.

System 7.14. *Initialization-LP*

Data: Data $(\mathbf{A}, d, c)$ for an instance of LP with data and values $M_P, M_D \geq 0$.

$$\begin{aligned} \min\ & \langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle & \max\ & \langle y, \mathbf{A}d\rangle - M_D\bar{s} \\ & \mathbf{A}x - \mathbf{A}\underline{x} = \mathbf{A}d & & \mathbf{A}^\top y + s - \bar{s}\mathbf{1} = c \\ & \sum_{i=1}^{n} x_i + \bar{x} = M_D & & -\mathbf{A}^\top y + \underline{s} = M_P\mathbf{1} - c \\ & x, \bar{x}, \underline{x} \geq 0 & & s, \bar{s}, \underline{s} \geq 0, \end{aligned}$$

$$\begin{aligned} \min\ & \langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle & \max\ & \langle c - s + \bar{s}\mathbf{1}, d\rangle - M_D\bar{s} \\ & x - \underline{x} \in W + d & & s - \bar{s}\mathbf{1} \in W^\perp + c \\ & \sum_{i=1}^{n} x_i + \bar{x} = M_D & & s - \bar{s}\mathbf{1} + \underline{s} = M_P\mathbf{1} \\ & x, \bar{x}, \underline{x} \geq 0 & & s, \bar{s}, \underline{s} \geq 0, \end{aligned}$$

Lemma 7.7.2 (Initialization). *If $M_P \geq C\|c\|$, $M_D \geq (n+1)C\|d\|$ and $C = 4$, then System 7.14 can be initialized near the central path induced by the standard log-barrier with parameter*

$$\mu_0 = O\left(\frac{M_P M_D}{n+1}\right) = O(\|c\|\|d\|). \tag{7.43}$$

Proof. For the primal side set $x = \frac{1}{n+1}M_D\mathbf{1}, \underline{x} = \frac{1}{n+1}(M_D\mathbf{1} - d), \bar{x} = \frac{1}{n+1}M_D$ and on the dual side $y = \mathbf{0}, s = M_P\mathbf{1}, \underline{s} = M_P\mathbf{1} + c, \bar{s} = M_P$. It is easy to see that $(x, \underline{x}, \bar{x}, y, s, \underline{s}, \bar{s})$ is a feasible solution to Initialization-LP (System 7.14). Further, for $\mu = \frac{1}{n+1}M_P M_D$ we have that

$$\begin{aligned} \|\mathbf{1} - \mu_0^{-1}(x \circ s, \underline{x} \circ \underline{s}, \bar{x} \circ \bar{s})\| &= \sqrt{\|\mathbf{1} - \mu^{-1}x \circ s\|^2 + \|\mathbf{1} - \mu^{-1}\underline{x} \circ \underline{s}\|^2 + \|\mathbf{1} - \mu^{-1}\bar{x} \circ \bar{s}\|^2} \\ &= \sqrt{0 + \|\mathbf{1} - \frac{1}{(n+1)\mu}(M_D\mathbf{1} - d) \circ (M_P\mathbf{1} + c)\|^2 + 0} \\ &= \frac{1}{M_P M_D}\|d \circ c - M_D\mathbf{1} \circ c + M_P\mathbf{1} \circ d\| \\ &= \left(\frac{1}{C^2(n+1)} + \frac{1}{C} + \frac{1}{C(n+1)}\right) \\ &\leq \frac{2}{C} \\ &\leq \frac{1}{2}, \end{aligned}$$

proving that the inital point is in the desired neighborhood of the central path. □

Proofs of Theorem 7.4.6 and Theorem 7.4.7

Theorem 7.4.6 (Restatement). *There exists a pair of algorithms (Algorithm 7.4 and Algorithm 7.5) that returns either of the following outcomes:*

(F1) Near feasible primal-dual solutions (x, s) such that

(F1.1) x is feasible to Apx-Feas(W, d, M, ε), and

Algorithm 7.4: Apx-Feasibility-Primal

Input : Instance of Primal with data (W, d), $\varepsilon > 0$,
Output : One in {(F1.1), (F2), (F4)}.

if $d = 0$ **then**
  return 0 ▷ Case (F1.1)
$c \leftarrow 0, M_P \leftarrow 1, M_D \leftarrow (C_f - 1)\|d\|, \mu \leftarrow \varepsilon\|d\|$
$(\underline{x}, x, \bar{x}, \underline{s}, s, \bar{s}) \leftarrow$ Blackbox-Approx-Solver(Initialization-LP$(M_P, M_D), \mu)$)
if $\langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s} \geq 0$ **then**
  return $x - \underline{x}$ ▷ Case (F1.1)
else
  $\Diamond \leftarrow \mathscr{h}^{n\times m}_{\text{im}}(\mathbf{A}^\top, \text{supp}((s - \bar{s}\mathbf{1})^-), J, M)$
  return $\Diamond$ ▷ Case (F2) or (F4)

Algorithm 7.5: Apx-Feasibility-Dual

Input : Instance of Dual with data (W, c), $\varepsilon > 0$,
Output : One in {(F1.2), (F3) , (F4)}.

if $c = 0$ **then**
  return 0 ▷ Case (F1.2)
$d \leftarrow 0, M_P \leftarrow n^{-1/2}C_f\|c\|, M_D \leftarrow 1, \mu \leftarrow \varepsilon\|c\|$
$(\underline{x}, x, \bar{x}, \underline{s}, s, \bar{s}) \leftarrow$ Blackbox-Approx-Solver(Initialization-LP$(M_P, M_D), \mu)$)
if $\langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle \geq 0$ **then**
  return $s - \bar{s}$ ▷ Case (F1.2)
else
  $\Diamond \leftarrow \mathscr{h}^{m\times n}_{\text{ker}}(\mathbf{A}, \text{supp}((x - \underline{x})^-), J, M)$
  return $\Diamond$ ▷ Case (F3) or (F4)

(F1.2) s is feasible to Apx-Feas$(W^\perp, c, M, \varepsilon)$.

(F2) A Farkas certificate of primal infeasibility: $y \in \mathfrak{F}(W, d)$,

(F3) A Farkas certificate of dual infeasibility: $y \in \mathfrak{F}(W^\perp, c)$,

(F4) A lifting certificate $(I, p) \in \mathfrak{C}(W, M)$.

The running time depends on the outcome and is presented in Table 7.1.

Outcome	Running time
(F1)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big)$
(F2)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{h}^{n\times m}_{\text{im}}(\mathbf{A}^\top))$
(F3)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{h}^{m\times n}_{\text{ker}}(\mathbf{A}))$
(F4)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{h}^{n\times m}_{\text{im}}(\mathbf{A}^\top)) + \mathcal{T}(\mathscr{h}^{m\times n}_{\text{ker}}(\mathbf{A}))$

Table 7.2: Running times to implement Theorem 7.4.6

Proof of Theorem 7.4.6. Let us first find a primal near-feasible solution via Algorithm 7.4. If $d = 0$, then $x = 0$ fulfills (F1.1). Otherwise, we apply Blackbox-Approx-Solver (Oracle 7.3) to System 7.14 (Initialization-LP) with data (W, d) as well as $(c, M_P, M_D) = (\mathbf{0}, 1, (C_f - 1)\|d\|)$ and initialize with $\mu_0 = O\big(\frac{1}{n+1}M_P M_D\big)$, noting that the assumptions of Lemma 7.7.2 are fulfilled. The terminating parameters for Blackbox-Approx-Solver are set to $\mu = \varepsilon\|d\|$. Note that $\log(\mu_0/\mu) = \widetilde{O}(\varepsilon^{-1}M)$ and so in time $\widetilde{O}(\Psi(\mathbf{A})\varepsilon^{-1}\log(M))$ we obtain solutions $(x, \underline{x}, \bar{x})$ and $(s, \underline{s}, \bar{s})$ with duality gap

$\langle x, s\rangle + \langle \underline{x}, \underline{s}\rangle + \langle \bar{x}, \bar{s}\rangle \leq \varepsilon\|d\|$. Let us distinguish between following cases based on the negated dual objective $\langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s}$ in System 7.14.

Case I. $\langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s} \geq 0$. In this case, the primal objective value in System 7.14 is

$$\langle \mathbf{1}, \underline{x}\rangle \leq \langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s} + \langle x, s\rangle + \langle \underline{x}, \underline{s}\rangle + \langle \bar{x}, \bar{s}\rangle \leq 0 + \varepsilon\|d\| = \varepsilon\|d\|. \tag{7.44}$$

Now, $\hat{x} \coloneqq x - \underline{x}$ is the required solution for (F1.1) as

$$\begin{aligned} \hat{x} &\in W + d, \\ \|\hat{x}^-\| &\leq \varepsilon\|d\|, \\ \|\hat{x}\| \leq \|\hat{x}\|_1 \leq \|x\|_1 + \|\underline{x}\|_1 &\leq (C_f - 1)\|d\| + \varepsilon\|d\| \leq C_f\|d\|\,. \end{aligned} \tag{7.45}$$

Case II. $\langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s} < 0$. Then $\hat{s} = s - \bar{s}\mathbf{1} \in W^\perp$ satisfies

$$\langle d, \hat{s}\rangle < -M_D\bar{s} = -(C_f - 1)\|d\|\bar{s} \leq -n^{-1}(C_f - 1)\|d\|\|\hat{s}^-\|_1\,. \tag{7.46}$$

Let $J \coloneqq \operatorname{supp}(\hat{s}^-)$. If $\diamond \coloneqq \mathscr{K}_{\mathrm{im}}^{n\times m}(\mathbf{A}^\top, \hat{s}, J, M)$ fullfils $\diamond \in \mathfrak{C}(W^\perp, M)$, then we terminate in case (F4). Else, let $z \coloneqq \diamond$. Then $z \in W^\perp$, is sign-consistent with $\hat{s}$, $z_J = 0$, and $\|z - \hat{s}\|_\infty \leq M\|\hat{s}^-\|_1$. 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 i.e. $z \in \mathfrak{F}(W, d)$ as in (F2). This follows since

$$\langle d, z\rangle = \langle d, \hat{s}\rangle + \langle d, z - \hat{s}\rangle \leq \langle d, \hat{s}\rangle + \|d\|_1\|z - \hat{s}\|_\infty \overset{(7.46)}{<} -n^{-1}C_f\|d\|\|\hat{s}^-\|_1 + M\|d\|_1\|\hat{s}^-\|_1 \leq 0, \tag{7.47}$$

where the last inequality follows by definition of C_f.

Analogously a near-feasible dual solution (or one of the alternatives) can be found with Algorithm 7.5. If $c = 0$, then $s = 0$ fulfills (F1.2). Otherwise, we apply Blackbox-Approx-Solver (Oracle 7.3) to System 7.14 (Initialization-LP) with data (W, c) as well as $(d, M_P, M_D) = (\mathbf{0}, n^{-1/2}C_f\|c\|, 1)$ and initialize with $\mu_0 = O\big(\frac{1}{n+1}M_PM_D\big)$, noting that the assumptions of Lemma 7.7.2 are fulfilled. The terminating parameters for Blackbox-Approx-Solver are set to $\mu = \varepsilon\|c\|$. Note that $\log(\mu_0/\mu) = \widetilde{O}(\varepsilon^{-1}M)$ and so in time $\widetilde{O}(\Psi(\mathbf{A})\varepsilon^{-1}\log(M))$ we obtain solutions $(x, \underline{x}, \bar{x})$ and $(s, \underline{s}, \bar{s})$ with duality gap $\langle x, s\rangle + \langle \underline{x}, \underline{s}\rangle + \langle \bar{x}, \bar{s}\rangle \leq \varepsilon\|c\|$. Let us consider following cases based on the primal objective $\langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle$.

Case I. $\langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle \geq 0$. In this case, the dual objective value is

$$-M_D\bar{s} > \langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle - (\langle x, s\rangle + \langle \underline{x}, \underline{s}\rangle + \langle \bar{x}, \bar{s}\rangle) \geq 0 - \varepsilon\|c\|\,. \tag{7.48}$$

Let $\hat{s} \coloneqq s - \bar{s}\mathbf{1}$. Note that by the constraint $\hat{s} + \underline{s} = M_P\mathbf{1}$ and $\underline{s} \geq$ we have that $\|\hat{s}^+\|_\infty \leq M_P$. Hence, we get the required solution

$$\begin{aligned} \hat{s} &\in W^\perp + c, \\ \|\hat{s}^-\| &\leq \sqrt{n}\bar{s} \leq \varepsilon\|c\|, \\ \|\hat{s}\| \leq \sqrt{n}\|\hat{s}\|_\infty \leq \sqrt{n}\max\{\|\hat{s}^+\|_\infty, \|\hat{s}^-\|_\infty\} &\leq \sqrt{n}\max\{M_P, \bar{s}\} \leq C_f\|c\| \end{aligned} \tag{7.49}$$

for (F1.2).

Case II. $\langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle < 0$. Then $\hat{x} \coloneqq x - \underline{x} \in W$ satisfies

$$\langle c, \hat{x}\rangle < -M_P\|\underline{x}\|_1 = -n^{-1/2}C_f\|c\|\|\underline{x}\|_1\,. \tag{7.50}$$

Let $J := \operatorname{supp}(\hat{x}^-)$. If $\diamond := \mathscr{h}^{m\times n}_{\ker}(\mathbf{A}, \hat{x}, J, M)$ fulfills $\diamond \in \mathfrak{C}(W, M)$, then we terminate in (F4). Else, let $w := \diamond$. Then $w \in W$ is sign-consistent with $\hat{x}$, $w_J = 0$ and $\|w - \hat{x}\|_\infty \le M\|\hat{x}^-\|_1$. By sign-consistency $w \ge 0$, since $\hat{x}_i \ge 0$ for all $i \in [n] \setminus J$. We show that $\langle c, w\rangle < 0$, and consequently, $w \in \mathfrak{F}(W^\perp, M)$ is a Farkas certificate as in (F3). This follows since

$$\langle c, w\rangle = \langle c, \hat{x}\rangle + \langle c, w - \hat{x}\rangle < -n^{-1/2}C_f\|c\|\|\hat{x}^-\|_1 + \sqrt{n}M\|\hat{x}^-\|_1\|c\| \le 0,$$

where the last inequality follows by choice of C_f. □

Algorithm 7.6: Apx-Optimal

Input : Instance of LP$(\mathbf{A}, d, c)$, $\varepsilon > 0$, guess M.
Output : One in {(M1), (M2), (M3), (M4)}

$\varepsilon_o \leftarrow \min\{(4C_o)^{-1}, \varepsilon\}$
$\varepsilon_f \leftarrow (4n)^{-1}\varepsilon_o$
$\diamond \leftarrow$ Apx-Feasibility-Primal(LP, ε_f)
if $\diamond$ *not in Case* (F1.1) **then**
  return $\diamond$
$\hat{x} \leftarrow \diamond$
$\diamond \leftarrow$ Apx-Feasibility-Dual(LP, ε_f)
if $\diamond$ *not in Case* (F1.2) **then**
  return $\diamond$
$\hat{s} \leftarrow \diamond$
$M_D \leftarrow 4Mn^{3/2}C_f\|d\|$, and, $M_P \leftarrow 4Mn^{1/2}C_f\|c\|$
$(\underline{x}, x, \bar{x}, \underline{s}, s, \bar{s}) \leftarrow$ Blackbox-Approx-Solver(
  Initialization-LP$(\mathbf{A}, d, c, M_P, M_D)$, ε_o))
$\tilde{x} \leftarrow x - \underline{x}, \tilde{s} \leftarrow s - \bar{s}\mathbf{1}$
if $\|\tilde{x}^-\| \le \varepsilon_o\|d\|$ **and** $\|\tilde{s}^-\| \le \varepsilon_o\|c\|$ **then**
  return $(\tilde{x}, \tilde{s})$
if $\|\tilde{x}^-\| > \varepsilon_o\|d\|$ **then**
  $\ell \leftarrow -\tilde{x} + (\mathbf{0} \wedge \hat{x})$
  $u \leftarrow \infty\mathbf{1}$
  $\diamond \leftarrow \mathscr{H}^{m\times n}_{\ker}(\mathbf{A}, \hat{x} - \tilde{x}, \ell, u, M)$
  if $\diamond \in \mathfrak{C}(W, M)$ **then**
    return v
  $v \leftarrow \diamond$
  $z \leftarrow \tilde{x} + v$
  $\bar{J} \leftarrow \operatorname{supp}((z - \hat{x})^-), \bar{I} \leftarrow [n] \setminus \bar{J}$
  $\bar{\ell}_{\bar{J}} \leftarrow z_{\bar{J}} - \hat{x}_{\bar{J}}, \bar{\ell}_{\bar{I}} \leftarrow 0, u \leftarrow z - \hat{x}$
  $\diamond \leftarrow \mathscr{H}^{m\times n}_{\ker}(\mathbf{A}, z - \hat{x}, \bar{\ell}, \bar{u}, M)$
  if $\diamond \in \mathfrak{C}(W, M)$ **then**
    return $\diamond$
  $\tilde{w} \leftarrow \diamond$
  $w \leftarrow z - \hat{x} - \tilde{w}$
  return w ▷ Case (F2) as we must have $\langle c, w\rangle < 0$
▷ We have $\|\tilde{s}^-\| > \varepsilon_o\|c\|$.
$\spadesuit \leftarrow \mathscr{H}^{n\times m}_{\operatorname{im}}(\mathbf{A}^\top, \tilde{s} - \hat{s}, \ell, u, M)$
return $\spadesuit$ ▷ Case (M4)

Theorem 7.4.7 (Restatement). *Let an instance of a linear program of the form System 1.2 and $M \ge 1$, $\varepsilon > 0$ be given. There exists an algorithm (Algorithm 7.6) that returns either of the following outcomes:*

(M1) A pair of primal and dual near-feasible and near-optimal solutions (x, s) as solutions to System 7.8, i.e., Apx-Opt$(W, d, c, M, \varepsilon)$.

(M2) A Farkas certificate of primal infeasibility $y \in \mathfrak{F}(W, d)$,

(M3) A Farkas certificate of dual infeasibility $y \in \mathfrak{F}(W^\perp, c)$, or

(M4) A lifting certificate $(I, p) \in \mathfrak{C}(W, M)$.

The running time depends on the outcome and is as follows:

Outcome	Running time
(M1)	$\widetilde{O}\left(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\right)$
(M2)	$\widetilde{O}\left(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\right) + \mathcal{T}(\mathscr{H}^{n\times m}_{\operatorname{im}}(\mathbf{A}^\top))$
(M3)	$\widetilde{O}\left(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\right) + \mathcal{T}(\mathscr{H}^{m\times n}_{\ker}(\mathbf{A}))$
(M4)	$\widetilde{O}\left(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\right) + \mathcal{T}(\mathscr{H}^{n\times m}_{\operatorname{im}}(\mathbf{A}^\top)) + \mathcal{T}(\mathscr{H}^{m\times n}_{\ker}(\mathbf{A}))$

Proof of Theorem 7.4.7. We let

$$\varepsilon_o := \min\{(4C_o)^{-1}, \varepsilon\}, \quad \varepsilon_f := (4n)^{-1}\varepsilon_o, \quad M_D := 4Mn^{3/2}C_f\|d\|, \quad \text{and} \quad M_P := 4Mn^{1/2}C_f\|c\|. \tag{7.51}$$

First, apply the algorithms Apx-Feasibility-Primal and Apx-Feasibility-Dual to data $(W, d, c, \varepsilon_f, M)$. Note that $\log({\varepsilon_f}^{-1}) = O(\log(Mn\varepsilon^{-1}))$, hence the asymptotic runtime bound is observed. If it returns in (F4) we return in (M4). If it returns in (F2), we return in (M2), if it returns in (F3), we return in (M3). Note that the claimed runtime bounds are observed. From now on assume that the call returned in (F1). Then we obtain vectors $\hat{x}$ and $\hat{s}$ such that

$$\begin{aligned} \hat{x} &\in W + d\,, \quad \|\hat{x}^-\| \leq \varepsilon_f\|d\|\,, \quad \|\hat{x}\| \leq C_f\|d\|\,, \\ \hat{s} &\in W^\perp + c\,, \quad \|\hat{s}^-\| \leq \varepsilon_f\|c\|\,, \quad \|\hat{s}\| \leq C_f\|c\|\,. \end{aligned} \tag{7.52}$$

Proceed by using Blackbox-Approx-Solver for System 7.14 with W, d, c, M_P, M_D and $\mu = \varepsilon_o\|c\|\|d\|$. In time $\widetilde{O}(\Psi(\mathbf{A})\log({\varepsilon_o}^{-1}M)))$ we can obtain solutions $(x, \underline{x}, \bar{x})$ and $(s, \underline{s}, \bar{s})$ with duality gap $\langle x, s\rangle + \langle \underline{x}, \underline{s}\rangle + \langle \bar{x}, \bar{s}\rangle \leq \varepsilon_o\|c\|\|d\|$.

We let $\tilde{x} = x - \underline{x} \in W + d$ and $\tilde{s} = s - \bar{s}\mathbf{1} \in W^\perp + c$ and distinguish three cases.

Case I is the successful case in which we can derive near-feasible and near-optimal dual solutions. In Case II we have large primal error in which case we will be able to conclude that either the guess of M was too low (certified by an element in $\mathfrak{C}(W, M)$) or that the primal is in fact infeasible, certified by a Farkas certificate in $\mathfrak{F}(W, M)$. Deriving one of these certificates is not straightforward as we are dealing with a modified linear program to get the near-feasible solution $\hat{x}$ than the program in which we aim to find the near-optimal solution as well. In the paragraph "Norm reduction" we explain how such a certificate can be obtained in Algorithm 7.6 nonetheless. The final case, Case III analogously finds a certificate of small guess M or obtains a certificate of dual infeasibility in $\mathfrak{F}(W^\perp, c)$. In this case the argumentation relying on norm reduction can be omitted.

Case I. $\|\tilde{x}^-\| \leq \varepsilon_o\|d\|$ and $\|\tilde{s}^-\| \leq \varepsilon_o\|c\|$.

In this case, we claim that $(\tilde{x}, \tilde{s})$ is a near-feasible near-optimal pair of solutions as required in (M1). It is straightforward, that $(\tilde{x}, \tilde{s}) \in (W + d) \times (W^\perp + c)$ fulfill the subspace constraints in System 7.8. Further, as $\varepsilon_o \leq \varepsilon$ the Primal apx. feasbility and Dual apx. feasibility constraints are fulfilled as well. Now , we have

$$\begin{aligned} -M_D\mathbf{1} \leq -\varepsilon_o\|d\|\mathbf{1} \leq -\tilde{x}^- \leq \tilde{x} = x - \underline{x} \leq M_D\mathbf{1}\,, \text{ and} \\ -M_P\mathbf{1} \leq -\varepsilon_o\|c\|\mathbf{1} \leq -\tilde{s}^- \leq \tilde{s} = s - \bar{s}\mathbf{1} \leq M_P\mathbf{1}, \end{aligned} \tag{7.53}$$

where all the inequalities follow by the assumption of Case I and the constraints in the subspace formulation in (7.14). In particular $\|\tilde{x}\|_\infty \leq M_D$ and $\|\tilde{s}\|_\infty \leq M_P$, implying $\|\tilde{x}\| \leq \sqrt{n}M_P\|d\| \leq C_o\|d\|$ (showing (7.15f)) and $\|\tilde{s}\| \leq \sqrt{n}M_D\|c\| \leq C_o\|c\|$ (showing (7.15g)). It remains to show (7.15e), i.e., the bound on $\|x \circ s\|$. With the norm bounds on $\tilde{x}$ and $\tilde{s}$ we obtain

$$\begin{aligned} \|\tilde{x} \circ \tilde{s}\| &\leq \|x \circ s\| + \|\tilde{x}^- \circ \tilde{s}^-\| + \|\tilde{x}^- \circ \tilde{s}\| + \|\tilde{x} \circ \tilde{s}^-\| \\ &\leq \left(\varepsilon_o + {\varepsilon_o}^2 + \varepsilon_o C_o + \varepsilon_o C_o\right)\|c\|\|d\| \\ &\leq 4\varepsilon_o C_o\|c\|\|d\| \\ &\leq \varepsilon\|c\|\|d\|. \end{aligned} \tag{7.54}$$

Case II. $\|\tilde{x}^-\| > \varepsilon_o\|d\|$.

In this case, we use the approximately feasible solution $\hat{x}$ to find a lifting certificate $(I, p) \in \mathfrak{C}(W, M)$ or show dual infeasibility. The intuition is that because of the high cost M_P on the variables in $\underline{x}$ the solution $\tilde{x}$ is actually a far worse solution than the almost feasible solution $\hat{x}$. On the other hand, $(\underline{x}, x, \bar{x})$ is almost optimal to System 7.14 as certified by the small duality gap with $(\underline{s}, s, \bar{s})$. We will turn this into a contradiction.

Let $I \coloneqq \operatorname{supp}(\hat{x}^-)$, $J = [n] \setminus I$ and let us define $\ell, u \in \overline{\mathbb{R}}^n$ as $\ell_J = -\tilde{x}_J$, $\ell_I = \hat{x}_I - \tilde{x}_I$, $u = \infty\mathbf{1}$. Then $\ell \le \hat{x} - \tilde{x} \le u$ and $\hat{x} - \tilde{x} \in W$. If $\diamond \coloneqq \mathscr{H}^{m\times n}_{\ker}(\mathbf{A}, \hat{x} - \tilde{x}, \ell, u, M) \in \mathfrak{C}(W, M)$ then we return in (M4). Otherwise, let $v \coloneqq \diamond$. Then, $v \in W, \ell \le v \le u$ and

$$\|v\|_\infty \le M\|\ell^+ + u^-\|_1 = M\|\ell^+\|_1 \le M\|\tilde{x}^-\|_1\,, \tag{7.55}$$

since $u \ge 0$, and $|\ell_i| \le \tilde{x}_i^-$ whenever $\ell_i > 0$. We let $z \coloneqq \tilde{x} + v$. Now,

$$z \in W + d, \quad -\hat{x}^- \le z, \quad \text{and} \quad \|z - \tilde{x}\|_\infty = \|v\|_\infty \le M\|\tilde{x}^-\|_1 \tag{7.56}$$

hold. In particular,

$$|z^-\|_1 \le \|\hat{x}^-\|_1 \le \sqrt{n}\|\hat{x}^-\| \le \sqrt{n}\varepsilon_f\|d\|, \tag{7.57}$$

where the last inequaility holds by (7.52).

Norm reduction If $\|z^+\|_1 > M_D$ we can not easily convert z to a feasible solution of (7.14) with the parameters M_P,M_D chosen above. Therefore, we reduce the ℓ_1-norm of z by sign-consistently moving it towards $\hat{x}$, without decreasing the objective value with respect to c.

To this end, let $\bar{J} \coloneqq \operatorname{supp}\big((z - \hat{x})^-\big)$, $\bar{I} \coloneqq [n] \setminus \bar{J}$, $\bar{\ell}_{\bar{J}} = z_{\bar{J}} - \hat{x}_{\bar{J}}$, $\bar{\ell}_{\bar{I}} = \mathbf{0}$, $u = z - \hat{x}$ and $\diamond \coloneqq \mathscr{H}^{m\times n}_{\ker}(\mathbf{A}, z - \hat{x}, \bar{\ell}, \bar{u}, M)$. If $\diamond \in \mathfrak{C}(W, M)$, then return in case (M4). Else, let $\tilde{w} \coloneqq \diamond$ and $w \coloneqq z - \hat{x} - \tilde{w} \in W$ is sign-consistent with $z - \hat{x}, w_{\bar{J}} = \mathbf{0}$ and

$$\|z - \hat{x} - w\|_\infty = \|\tilde{w}\|_\infty \le M\|\bar{\ell}^+ + \bar{u}^-\|_1 = M\|z_{\bar{J}} - \hat{x}_{\bar{J}}\|_1 \le M\big(\|z^-\|_1 + \|\hat{x}\|_1\big). \tag{7.58}$$

By choice of $\bar{\ell}$ and $\bar{u}$ we further have $w \ge 0$. If $\langle c, w\rangle < 0$, then w is a Farkas certificate of dual infeasibility $w \in \mathfrak{F}(W^\perp, c)$ and we return in (M3). Else $\langle c, w\rangle \ge 0$ and $\tilde{z} \coloneqq z - w$ fulfills

$$\tilde{z} \in W + d,\ \langle \tilde{z}, c\rangle = \langle z - w, c\rangle \le \langle z, c\rangle \tag{7.59}$$

and we have

$$\|\tilde{z}^-\|_1 = \|(z_{\bar{J}} - w_{\bar{J}})^-\|_1 + \|(z_{\bar{I}} - w_{\bar{I}})^-\|_1 \le \|z^-\|_1 + \|(\hat{x} + \tilde{w})^-\|_1 \overset{\tilde{w}_{\bar{I}} \ge \bar{\ell}_{\bar{I}} = 0}{\le} \|z^-\|_1 + \|\hat{x}^-\|_1 \overset{(7.57),(7.52)}{\le} 2\sqrt{n}\varepsilon_f\|d\|, \tag{7.60}$$

Further note that

$$\begin{aligned}\|\tilde{z}\|_\infty = \|z - w\|_\infty &\overset{(7.58)}{\le} M(\|z^-\|_1 + \|\hat{x}\|_1) + \|\hat{x}\|_\infty \le (M+1)\|\hat{x}\|_1 + M\|z^-\|_1 \\ &\overset{(7.52),(7.57)}{\le} (M+1)\sqrt{n}C_f\|d\| + \sqrt{n}\varepsilon_f M\|d\| \le 4M\sqrt{n}C_f\|d\| \overset{(7.51)}{\le} \frac{M_D}{n}.\end{aligned} \tag{7.61}$$

We are ready to show hat $\tilde{z}$ contradicts the small duality gap of $\tilde{x}$. By (7.61) we can map $\tilde{z}$ to a primal feasible solution

$$(z', \bar{z}', \underline{z}') \coloneqq (\tilde{z}^+, M_D - \|\tilde{z}^+\|_1, \tilde{z}^-) \ge \mathbf{0} \tag{7.62}$$

of (7.14). The objective value of this solution is

$$\begin{aligned}
\langle c, z' - \underline{z}' \rangle + M_P \|\underline{z}'\|_1 &= \langle c, \tilde{z} \rangle + M_P \|\tilde{z}^-\|_1 \\
&\leq \langle c, z \rangle + 2\sqrt{n}\varepsilon_f M_P \|d\| && \text{(by (7.59), (7.60))} \\
&= \langle c, \tilde{x} \rangle + \langle c, z - \tilde{x} \rangle + 2\sqrt{n}\varepsilon_f M_P \|d\| \\
&\leq \langle c, \tilde{x} \rangle + \|c\|_1 \|z - \tilde{x}\|_\infty + 2\sqrt{n}\varepsilon_f M_P \|d\| \\
&\leq \langle c, \tilde{x} \rangle + M\|c\|_1 \|\tilde{x}^-\|_1 + 2\sqrt{n}\varepsilon_f M_P \|d\| && \text{(by (7.56))} \\
&< \langle c, \tilde{x} \rangle + \left(\sqrt{n}M\|c\| + 2n{\varepsilon_o}^{-1}\varepsilon_f M_P\right)\|\tilde{x}^-\|_1 && (\|\tilde{x}^-\| > \varepsilon_o\|d\|) \\
&< \langle c, \tilde{x} \rangle + \left(\sqrt{n}M\|c\| + 2n{\varepsilon_o}^{-1}\varepsilon_f M_P\right)\|\tilde{x}^-\|_1 + \|c\|(\|\tilde{x}^-\| - \varepsilon_o\|d\|) \\
&\leq \langle c, \tilde{x} \rangle + \left(\sqrt{n}(M+1)\|c\| + 2n{\varepsilon_o}^{-1}\varepsilon_f M_P\right)\|\tilde{x}^-\|_1 - \varepsilon_o\|c\|\|d\| \\
&\leq \langle c, x - \underline{x} \rangle + M_P\|\underline{x}\|_1 - \varepsilon_o\|c\|\|d\|, \\
& && \text{(by (7.51) } \varepsilon_f \leq (4n)^{-1}\varepsilon_o, M_P \geq 2(M+1)\sqrt{n}\|c\|)
\end{aligned}$$

This is a contradiction, since $\langle c, x - \underline{x} \rangle + M_P\|\underline{x}\|_1$ is the objective value of the solution $(x, \bar{x}, \underline{x})$ that was chosen to be within $\varepsilon_o\|c\|\|d\|$ of the optimum value.

Hence, such a feasible solution does not exist. So either we had that $\diamond \in \mathfrak{C}(W, M)$, (i.e. we were in case (M4)) or we terminated with the Farkas certificate $w \in \mathfrak{F}(W^\perp, c)$.

Case III. $\|\tilde{s}^-\| > \varepsilon_o\|c\|$.

The argument is similar to the one in Case II, but we do not need to perform the norm reducton.

Again, we use the approximately feasible solution $\hat{s}$ to find a lifting certificate in $\mathfrak{C}(W^\perp, M)$ or show primal infeasibility. Let us define $\ell := \tilde{s} - M_P\mathbf{1}$ and $u := \tilde{s} + \hat{s}^-$ and let $\spadesuit := \mathscr{H}^{n\times m}_{\text{im}}(\mathbf{A}^\top, \tilde{s} - \hat{s}, \ell, u, M)$. If $\spadesuit \in \mathfrak{C}(W^\perp, M)$ we terminate in case (M4). Else, let $v := \spadesuit$. Then, $v \in W^\perp$ and fulfils

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

since $\ell \leq \mathbf{0}$ and $|u_i| \leq \tilde{s}_i^-$ whenever $u_i < 0$. We let $z := \tilde{s} - v$. Now

$$z \in W^\perp + c\,, \quad -\hat{s}^- = \tilde{s} - u \leq \tilde{s} - v = z\,, \quad z = \tilde{s} - v \leq \tilde{s} - \ell = M_P\mathbf{1} \quad \text{and} \quad \|z - \tilde{s}\|_\infty = \|v\|_\infty \leq M\|\tilde{s}^-\|_1. \tag{7.63}$$

In particular,

$$\|z^-\|_1 \leq \|\hat{s}^-\|_1 \leq \sqrt{n}\varepsilon_f\|c\|. \tag{7.64}$$

The dual objective of System 7.14 can be rewritten as

$$\langle y, \mathbf{A}d \rangle - M_D\bar{s} = \left\langle \mathbf{A}^\top y, d \right\rangle - M_D\bar{s} = \langle c, d \rangle - \langle s - \bar{s}\mathbf{1}, d \rangle - M_D\bar{s},$$

as used in the subspace version of System 7.14. So, equivalently we could minimize $\langle s - \bar{s}\mathbf{1}, d \rangle + M_D\bar{s}$. The vector z can be transformed into a feasible dual solution as follows. Let $z_m := \min(0, \min_{i\in[n]} z_i)$. Note that $|z_m| = \|z^-\|_\infty \leq \sqrt{n}\varepsilon_f\|c\|$ by (7.64). We can define a dual feasible solution $(z', \bar{z}', \underline{z}')$ to the System 7.14 with W, d, c, M_P and M_D as $(z', \bar{z}', \underline{z}') := (z -$

$z_m\mathbf{1}, -z_m, M_P\mathbf{1} - z) \geq \mathbf{0}$ by (7.63). We proceed in similar manner as for Case II and show that this feasible solution contradicts near-optimality of $(y, s, \underline{s}, \bar{s})$ for the dual of System 7.14. We have

$$\begin{aligned}
\langle d, z' - \bar{z}'\mathbf{1}\rangle + M_D\bar{z}' &= \langle d, z\rangle + M_D|z_m| \\
&\leq \langle d, \tilde{s}\rangle + \langle d, z - \tilde{s}\rangle + \sqrt{n}\varepsilon_f M_D\|c\| && (|z_m| \leq \sqrt{n}\varepsilon_f\|c\|, \text{see above.}) \\
&\leq \langle d, \tilde{s}\rangle + \|d\|_1\|z - \tilde{s}\|_\infty + \sqrt{n}\varepsilon_f M_D\|c\| \\
&\leq \langle d, \tilde{s}\rangle + M\|d\|_1\|\tilde{s}^-\|_1 + \sqrt{n}\varepsilon_f M_D\|c\| && (\text{by } (7.63)) \\
&\leq \langle d, \tilde{s}\rangle + \sqrt{n}M\|d\|\|\tilde{s}^-\|_1 + \sqrt{n}\varepsilon_f M_D\|c\| \\
&< \langle d, \tilde{s}\rangle + \left(\sqrt{n}M\|d\| + \sqrt{n}\varepsilon_f{\varepsilon_o}^{-1}M_D\right)\|\tilde{s}^-\|_1 && (\|\tilde{s}^-\| > \varepsilon_o\|c\|) \\
&< \langle d, \tilde{s}\rangle + \left(\sqrt{n}M\|d\| + \sqrt{n}\varepsilon_f{\varepsilon_o}^{-1}M_D\right)\|\tilde{s}^-\|_1 + \|d\|(\|\tilde{s}^-\| - \varepsilon_o\|c\|) \\
&< \langle d, \tilde{s}\rangle + \left(\sqrt{n}(M+1\|d\| + \sqrt{n}\varepsilon_f{\varepsilon_o}^{-1}M_D\right)\|\tilde{s}^-\|_1 - \varepsilon_o\|c\|\|d\| \\
&\leq \langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s} - \varepsilon_o\|c\|\|d\|, \\
& && (\text{by } (7.51)\ M_D \geq 2\sqrt{n}(M+1)\|d\|,\ \varepsilon_f \leq (2n)^{-1}\varepsilon_o)
\end{aligned}$$

This is a contradiction, since $\langle d, s - \bar{s}\mathbf{1}\rangle + M_D\bar{s}$ is the objective value of the solution $(s, \bar{s}, \underline{s})$ that was chosen to be within $\varepsilon_o\|c\|\|d\|$ of the optimum value.

Hence, such a feasible solution does not exist, and therefore, ♠ must have fulfilled $\spadesuit \in \mathfrak{C}(W^\perp, M)$, i.e., it is a lifting certificate (Case (M4)). □

7.7.2 Implementation of the oracles

We proceed to show Lemma 7.5.1 and Lemma 7.6.3 which correspond to the implementations of Oracle 7.1 and Oracle 7.2, respectively.

Lemma 7.5.1 (Restatement). *Given a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ and vector $d \in \mathbb{R}^n$. Let $W = \ker(\mathbf{A})$, and M be an estimate on κ_W. Further, let $0 < \varepsilon < 1$. There exists an implementation of Oracle 7.1 for data (W, d, M, ε) with following outcome-dependent running times:*

Outcome	Running time
(i)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big)$
(ii), *(iii)*	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))$

Lemma 7.6.3 (Restatement). *There exists an implementation of* Proximal-Opt-Oracle *(Oracle 7.2) for data $(W, d, c, \check{c}, M, \varepsilon)$ with following running times, depending on the outcomes:*

Outcome	Running time
(i)	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big)$
(ii), *(iii)*, *(iv)*	$\widetilde{O}\big(\Psi(\mathbf{A})\log(M\varepsilon^{-1})\big) + \mathcal{T}(\mathscr{H}_{\text{im}}(\mathbf{A}^\top))$

We note that Oracle 7.1 is a special case of Oracle 7.2; thus, Lemma 7.5.1 follows as a special case of Lemma 7.6.3. Before proving the latter we formulate another Lemma that will be required in the proof (Lemma 7.7.3).

Lemma 7.7.3 (Exchanging near optimal solutions). *For $W \subseteq \mathbb{R}^n, d, c \in \mathbb{R}^n$, guess $M \geq 1$ and $\varepsilon > 0$ let (x, s) and $(\bar{x}, \bar{s})$ be two solutions to* Apx-Opt*$(W, d, c, M, \varepsilon)$. Then, $(x, \bar{s})$ and $(\bar{x}, s)$ are both solutions to* Apx-Opt*$\big(W, d, c, M, 8\varepsilon\sqrt{n}C_o\big)$.*

Algorithm 7.7: Apx-Optimal-Proximal (Implementation of Oracle 7.2)

Input : Data $(W, d, c, \check{c}, M, \varepsilon)$ for Prox-Opt (System 7.13)
Output : One in {(i), (ii), (iii), (iv)}

Set $\hat{I}, \hat{J}, \bar{I}, \bar{J}, \bar{\varepsilon}, \hat{\varepsilon}$ as in (7.69)
// Step I
$\diamond \leftarrow$ Apx-Optimal$(\pi_{\hat{I}}(W), d_{\hat{I}}, c_{\hat{I}}, M, \hat{\varepsilon})$
if $\diamond \in \mathfrak{F}(\pi_{\hat{I}}(W), d_{\hat{I}})$ **then**
  return $(0_{\hat{J}}, \diamond)$ ▷ $\in \mathfrak{F}(W, d)$, see Fact 7.2.7
if $\diamond \in \mathfrak{C}(\pi_{\hat{I}}(W), M)$ **then**
  Lift $\diamond$ to an element $\tilde{z} \in \mathfrak{C}(W, M)$
  return $\tilde{z}$;
$(\hat{x}, \hat{s}) \leftarrow \diamond$
// Step II
$\spadesuit \leftarrow$ Apx-Optimal$(W, \bar{d}, c, M, \bar{\varepsilon})$
if $\spadesuit \in \mathfrak{F}(W, \bar{d})$ **then**
  return $\spadesuit$
if $\spadesuit \in \mathfrak{C}(W, M)$ **then**
  return $\spadesuit$
$(w, \bar{s}) \leftarrow \spadesuit$
$\bar{x} \leftarrow w + d - \bar{d}$
// Step III
$\tilde{s} \leftarrow (0_J, \hat{s})$
if $\|\bar{x} \circ \tilde{s}\| \le \delta \|d_{\Lambda(d,c)}\| \|\check{c}\|$ **then**
  return $(\bar{x}, \tilde{s})$
else
  $\ell \leftarrow -d_{\hat{I}} - \hat{x}^-$
  Let $u \in (\mathbb{R} \cup \{\infty\})^{\hat{I}}$ such that
  $u_i \leftarrow \begin{cases} \bar{x}_i - d_i & \text{if } i \in \operatorname{supp}(c), \\ \infty & \text{else.} \end{cases}$
  $\diamond \leftarrow \mathscr{H}^{m \times n}_{\ker}(\mathbf{A}, \bar{x} - d, \ell, u, M)$
  return $\diamond$ ▷ $\diamond \in \mathfrak{C}(W, M)$

Proof. All constraints but the duality gap (7.15e) are preserved. Here, note that $\langle x - \bar{x}, s - \bar{s} \rangle = 0$ as $x - \bar{x} \in W$ and $s - \bar{s} \in W^\perp$ and so

$$\langle x, \bar{s} \rangle + \langle \bar{x}, s \rangle = \langle x, s \rangle + \langle \bar{x}, \bar{s} \rangle \le \sqrt{n}(\|x \circ s\| + \|\bar{x} \circ \bar{s}\|) \le 2\sqrt{n}\varepsilon \|c\| \|d\|. \tag{7.65}$$

Furthermore, we have that $\langle x, s \rangle \ge -\sqrt{n}(\|x^-\| \|s\| + \|x\| \|s^-\|) \ge -2\sqrt{n}\varepsilon C_o \|d\| \|c\|$. By the same argumentation $\langle \bar{x}, \bar{s} \rangle \ge -2\sqrt{n}\varepsilon C_o \|d\| \|c\|$, and so

$$\begin{aligned} \langle x, \bar{s} \rangle &\le 2\varepsilon\sqrt{n}(C_o + 1)\|d\| \|c\|, \\ \langle \bar{x}, s \rangle &\le 2\varepsilon\sqrt{n}(C_o + 1)\|d\| \|c\| \end{aligned} \tag{7.66}$$

and thus

$$\begin{aligned} \|x \circ \bar{s}\| \le \|x \circ \bar{s}\|_1 &\le \langle x, \bar{s} \rangle + 2(\|x^-\|_1 \|\bar{s}\| + \|x\| \|\bar{s}^-\|_1) \\ &\le 2\varepsilon\sqrt{n}(C_o + 1)\|d\| \|c\| + 4\varepsilon\sqrt{n}C_o \|d\| \|c\| \\ &\le 8\varepsilon\sqrt{n}C_o \|d\| \|c\|. \end{aligned} \tag{7.67}$$

The bound for $\|\bar{x} \circ s\|$ follows by symmetry analogously. □

In Algorithm 7.7 and in its analysis via Lemma 7.6.3 we will require the following recurring terms.

$$\delta := \frac{\varepsilon^2}{\sqrt{n}}. \tag{7.68}$$

Furthermore, let

$$\begin{aligned} \hat{I} &:= \left\{ i \in [n] : d_i \le 2M\|d_{\Lambda(d,c)}\| \right\}, \quad \hat{J} := [n] \setminus \hat{I}, \quad \hat{\varepsilon} := \frac{\delta}{16MnC_o \cdot 8\sqrt{n}C_o \cdot 4Mn^{3/2}}, \\ \bar{I} &:= \left\{ i \in [n] : d_i \le 4Mn\|d_{\Lambda(d,c)}\| \right\}, \quad \bar{J} := [n] \setminus \bar{I}, \quad \bar{\varepsilon} := \frac{\delta}{8\sqrt{n}C_o \cdot 4Mn^{3/2}}. \end{aligned} \tag{7.69}$$

Proof of Lemma 7.6.3. We split the proof into three steps. In Step I we construct the dual solution the problem. In Step II we construct the primal solution. In Step III we show that if stitching

together the primal solution from Step II and dual solution from Step I fails, then we are able to construct a certificate of such failure.

Step I. *Providing the dual solution.*

We run Apx-Optimal (Algorithm 7.6) on the data $(\pi_{\hat{I}}(W), d_{\hat{I}}, \check{c}_{\hat{I}})$ with guess M and parameter $\hat{\varepsilon}$. The reason that we do not consider the elements in $[n] \setminus \hat{I}$ is that we need to bound the minimum-norm vector in the affine space in which we run the solver. Note that $\|d_{\hat{I}}\| \leq 2\sqrt{n}M\|d_{\Lambda(d,c)}\|$ and that $\operatorname{supp}(c) \subseteq \hat{I}$ as $c \geq \mathbf{0}$. 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 the original system (see Fact 7.2.7). (M3) can not happen as $c_{[n]\setminus\hat{I}} = \mathbf{0}$ by definition of $\hat{I}$ and so $c_{\hat{I}}$ is a feasible dual solution to $\mathrm{LP}(\pi_{\hat{I}}(W), d_{\hat{I}}, \check{c}_{\hat{I}})$, by further noting that $\check{c}_{\hat{I}} - c_{\hat{I}} \in \big(\pi_{\hat{I}}(W)\big)^{\perp}$. If in case (M4), we output the corresponding lifting certificate in $\mathfrak{C}(\pi_{\hat{I}}(W), M)$ and terminate. The remaining case is that we obtained approximately feasible and approximately optimal primal and dual solutions $(\hat{x}, \hat{s})$ as in (M1) fulfilling

$$\begin{aligned} \|\hat{x}^-\| &\leq \hat{\varepsilon}\|d_{\hat{I}}\| \leq \hat{\varepsilon} \cdot 2\sqrt{n}M\|d_{\Lambda(d,c)}\| \leq \varepsilon\|d_{\Lambda(d,c)}\|, \\ \|\hat{s}^-\| &\leq \hat{\varepsilon}\|\check{c}_{\hat{I}}\|, \\ \|\hat{x} \circ \hat{s}\| &\leq \hat{\varepsilon}\|d_{\hat{I}}\|\|\check{c}_{\hat{I}}\|, \\ \|\hat{x}\| &\leq C_o\|d_{\hat{I}}\|, \text{ and} \\ \|\hat{s}\| &\leq C_o\|\check{c}_{\hat{I}}\|. \end{aligned} \tag{7.70}$$

Step II. *Providing the primal solution.*

Let $\bar{d} \in \mathbb{R}^n$ be defined as

$$\bar{d}_i := \begin{cases} d_i & \text{if } i \in \bar{I}, \\ 2Mn\|d_{\Lambda(d,c)}\| & \text{otherwise.} \end{cases} \tag{7.71}$$

Essentially, $\bar{d}$ is d with a cut-off for variables large compared to the norm of $d_{\Lambda(d,c)}$. Consider $\mathrm{LP}(W, \bar{d}, \check{c})$ and apply Algorithm 7.6 with M and $\bar{\varepsilon}$. We analyze the different outcomes according to Theorem 7.4.7. As in Step I the case (M3) can not occur as c is a feasible dual solution. In case of primal infeasibility (M2), we show how to convert the certificate of primal infeasibility for the auxiliary system $\mathrm{LP}(W, \bar{d}, \check{c})$ into either a primal infeasibility certificate of the original system or find an element in $\mathfrak{C}(W, M)$. A certificate of primal infeasibility for $\mathrm{LP}(W, \bar{d}, c)$ is a vector $s \in \mathfrak{F}(W, \bar{d})$, i.e., $s \in W^{\perp}, s \geq 0$ such that $\langle \bar{d}, s \rangle < 0$. By definition of $\bar{d}$, the last condition is equivalent to

$$\langle d_{\bar{I}}, s_{\bar{I}} \rangle + \big\langle 4Mn\|d_{\Lambda(d,c)}\|\mathbf{1}_{\bar{J}}, s_{\bar{J}} \big\rangle = \langle \bar{d}, s \rangle < 0. \tag{7.72}$$

Let $\ell, u \in \mathbb{R}^n$ such that $\ell_{\bar{I}} = \mathbf{0}$, $\ell_{\bar{J}} = s_{\bar{J}}$ and $u = s$. Then compute $\diamond := \mathscr{H}^{n\times m}_{\mathrm{im}}(\mathbf{A}^{\top}, s, \ell, u, M)$. If $\diamond \in \mathfrak{C}(W^{\perp}, M)$ terminate with the Farkas certificate, otherwise let $y := \diamond$. Then $y \in W^{\perp}$ and $z := s - y$ fulfills $s \geq z \geq \mathbf{0}$ such that $z_{\bar{J}} = \mathbf{0}$ and $\|s - z\|_{\infty} = \|y\|_{\infty} \leq M\|u^- + \ell^+\|_1 \leq M\|s_{\bar{J}}\|_1$. But then

$$\begin{aligned} \langle d, z \rangle = \langle d_{\bar{I}}, z_{\bar{I}} \rangle &= \langle d_{\bar{I}}, s_{\bar{I}} \rangle + \langle d_{\bar{I}}, z_{\bar{I}} - s_{\bar{I}} \rangle \overset{(7.72)}{<} -\big\langle 4Mn\|d_{\Lambda(d,c)}\|\mathbf{1}_{\bar{J}}, s_{\bar{J}} \big\rangle + \|d^-_{\bar{I}}\|_1\|s - z\|_{\infty} \\ &\leq -\big\langle 4Mn\|d_{\Lambda(d,c)}\|\mathbf{1}_{\bar{J}}, s_{\bar{J}} \big\rangle + M\|d_{\Lambda(d,c)}\|_1\|s_{\bar{J}}\|_1 \leq 0, \end{aligned}$$

so z is a Farkas certificate of the original system $\mathrm{LP}(W, d, c)$, i.e., $z \in \mathfrak{F}(W, d)$.

It remains to consider the (successful) case (M1), i.e., we obtain primal-dual near-feasible and near-optimal solution $(\bar{w}, \bar{s})$ to $\mathrm{LP}(W, \bar{d}, \check{c})$. In this case, we can define $\bar{x} := \bar{w} + d - \bar{d}$ and get

$$\begin{aligned} \bar{x} &\in W + d, \\ \bar{s} &\in W^\perp + \check{c} = W^\perp + c, \\ \|\bar{x}^-\| &\leq \|\bar{w}^-\| \leq \bar{\varepsilon}\|\bar{d}\|, \\ \|\bar{s}^-\| &\leq \bar{\varepsilon}\|\check{c}\|, \\ \|\bar{w} \circ \bar{s}\| &\leq \bar{\varepsilon}\|\bar{d}\|\|\check{c}\| \leq 4\bar{\varepsilon}Mn^{3/2}\|d_{\Lambda(d,c)}\|\|\check{c}\|, \\ \|\bar{w}\| &\leq C_{\mathrm{o}}\|\bar{d}\| \leq 4C_{\mathrm{o}}Mn^{3/2}\|d_{\Lambda(d,c)}\|, \text{ and} \\ \|\bar{s}\| &\leq C_{\mathrm{o}}\|\check{c}\| \end{aligned} \tag{7.73}$$

Note that for the primal side we get the proximity and near-non-negativity from

$$\|\bar{x} - d\| = \|\bar{w} - \bar{d}\| \leq \|\bar{w}\| + \|\bar{d}\| \leq 2 \cdot 4C_{\mathrm{o}}Mn^{3/2}\|d_{\Lambda(d,c)}\|, \text{ and} \tag{7.74}$$

$$\|\bar{x}^-\| \leq \bar{\varepsilon}\|\bar{d}\| \leq 4\bar{\varepsilon}Mn^{3/2}\|d_{\Lambda(d,c)}\| \leq \frac{\varepsilon}{\sqrt{n}}\|d_{\Lambda(d,c)}\|, \tag{7.75}$$

Now, for $j \in \bar{J}$ note that

$$\bar{x}_j = w_j - \bar{d}_j + d_j \geq -\|w^-\| + (4-2)Mn\|d_{\Lambda(d,c)}\| \geq (-\varepsilon + 2Mn)\|d_{\Lambda(d,c)}\| \geq Mn\|d_{\Lambda(d,c)}\|, \tag{7.76}$$

Step III. *Combining primal and dual solution.*

Let us extend the dual solution $\hat{s}$ to the full space via $\tilde{s} := (\mathbf{0}_J, \hat{s})$. We now consider the primal-dual pair $(\bar{x}, \tilde{s})$ and distinguish cases based on $\|\bar{x} \circ \tilde{s}\|$.

We distinguish following two cases.

Case I. $\|\bar{x} \circ \tilde{s}\| \leq \delta\|d_{\Lambda(d,c)}\|\|\check{c}\|$.

We can now show that $(\bar{x}, \tilde{s})$ is the desired solution to System 7.13. As $\delta < \varepsilon$ the condition (7.42c) is fulfilled. The subspace conditions (7.42a) and (7.42b) are straightforward. The primal proximity constraint (7.42d) follows from (7.74) and the choice of C_{po}. From (7.75) follows near-feasibility (7.42e). Dual near-feasibility (7.42f) and norm bound (7.42g) follow from (7.70).

Case II. $\|\bar{x} \circ \tilde{s}\| > \delta\|d_{\Lambda(d,c)}\|\|\check{c}\|$.

We will show that in this case $M < \kappa_W$ must hold, which we will certify with an element in $\mathfrak{C}(W, M)$. We do so by showing that the primal solution $\hat{x}$ in Step I of the proof can not be used in conjunction with Hoffman proximity results to produce a proximal solution to d.

To this end, extend $\hat{x}$ arbitrarily to an element $\tilde{x} \in W + d$ such that $\tilde{x}_{\hat{I}} = \hat{x}$. 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 let $\ell, u \in \mathbb{R}^{\hat{I}}$ such that

$$\ell_{\hat{I}} := -d_{\hat{I}} - \hat{x}^- \quad \text{and} \quad u_i := \begin{cases} \tilde{x}_i - d_i & \text{if } i \in \mathrm{supp}(c), \\ \infty & \text{else.} \end{cases}$$

These vectors have the property

$$\|\ell^+ + u^-\|_1 \leq \|d_{\Lambda(d,c)}\|_1 + \|\hat{x}^-\|_1 \overset{(7.70)}{\leq} \|d_{\Lambda(d,c)}\|_1 + \bar{\varepsilon} \cdot 2M\sqrt{n}\|d_{\Lambda(d,c)}\|_1 \leq 2\|d_{\Lambda(d,c)}\|_1, \tag{7.77}$$

Let $\diamond := \mathscr{H}^{m\times n}_{\ker}(\mathbf{A}, \tilde{x} - d, \ell, u, M)$. In the remainder of the proof we show that $\diamond \in \mathfrak{C}(W, M)$, i.e., we terminate with a lifting certificate.

Assume not. Then $z := \diamond$ fulfills $z \in \mathbb{R}^n$ with $\|z\|_\infty \leq M\|\ell^+ + u^-\|_1 \leq 2M\|d_{\Lambda(d,c)}\|_1$ by (7.77). Defining $\check{x} := d + z$, we note that $\check{x}_{\hat{J}} \geq d_{\hat{J}} - \|z\|_\infty \mathbf{1} \geq \mathbf{0}$ and $\langle c, \check{x}\rangle = \langle c, d + z\rangle \leq \langle c, \tilde{x}\rangle$ hold as $c \geq \mathbf{0}$ and $\check{x}_i = d_i + z_i \leq d_i + u_i = \tilde{x}_i$ for $i \in \operatorname{supp}(c)$. Therefore, we get that

$$\langle \check{x}, \tilde{s}\rangle = \langle \check{x} - \tilde{x}, \tilde{s}\rangle + \langle \tilde{x}, \tilde{s}\rangle = \langle \check{x} - \tilde{x}, c\rangle + \langle \tilde{x}, \tilde{s}\rangle \leq \langle \tilde{x}, \tilde{s}\rangle = \langle \hat{x}, \hat{s}\rangle \leq \|\hat{x} \circ \hat{s}\| \overset{(7.70)}{\leq} \hat{\varepsilon}\|d_{\hat{J}}\|\|\check{c}_{\hat{J}}\|. \tag{7.78}$$

Using the bound on the norm of z we get the primal proximity as

$$\|\check{x} - d\|_\infty = \|z\|_\infty \leq 2M\|d_{\Lambda(d,c)}\|_1\,, \tag{7.79}$$

from which we conclude that

$$\|\check{x}_{\hat{J}}\| \leq \|\check{x}_{\hat{J}} - d_{\hat{J}}\| + \|d_{\hat{J}}\| \overset{(7.79)}{\leq} 2M\sqrt{n}\|d_{\Lambda(d,c)}\| + 2M\sqrt{n}\|d_{\Lambda(d,c)}\| \leq 4M\sqrt{n}\|d_{\Lambda(d,c)}\|. \tag{7.80}$$

Further, we have

$$\|\check{x}^-_{\hat{I}}\| \leq \|(d_{\hat{I}} + \ell)^-\| = \|\hat{x}^-\| \leq \hat{\varepsilon}\|d_{\hat{I}}\|. \tag{7.81}$$

Therefore, with (7.78), (7.79), (7.80) and (7.81), we have that

$$\begin{aligned}
\|\check{x} \circ \tilde{s}\| = \|\check{x}_{\hat{I}} \circ \tilde{s}_{\hat{I}}\| &\leq \langle \check{x}, \tilde{s}\rangle + 2\Big(\|\check{x}^-_{\hat{I}}\|_1\|\tilde{s}_{\hat{I}}\|_\infty + \|\check{x}_{\hat{I}}\|_\infty\|\tilde{s}^-_{\hat{I}}\|_1\Big) \\
&\leq \hat{\varepsilon}\|d_{\hat{I}}\|\|\check{c}_{\hat{I}}\| + 2\sqrt{n}\Big(\hat{\varepsilon}\|d_{\hat{I}}\| \cdot C_{\mathrm{o}}\|\check{c}\| + 4M\sqrt{n}\|d_{\Lambda(d,c)}\| \cdot \sqrt{n}\hat{\varepsilon}\|\check{c}\|\Big) \\
&\leq \Big(2M\sqrt{n} + 2\sqrt{n} \cdot 2M\sqrt{n} \cdot C_{\mathrm{o}} + 4Mn\Big)\hat{\varepsilon}\|d_{\Lambda(d,c)}\|\|\check{c}\| \\
&\leq 16MnC_{\mathrm{o}}\hat{\varepsilon}\|d_{\Lambda(d,c)}\|\|\check{c}\|.
\end{aligned} \tag{7.82}$$

Further

$$\check{x}_{\hat{J}} - d_{\hat{J}} + \bar{d}_{\hat{J}} \geq -\|\check{x}_{\hat{J}} - d_{\hat{J}}\|_\infty \mathbf{1} + \bar{d}_{\hat{J}} \overset{(7.79)}{\geq} -2M\sqrt{n}\|d_{\Lambda(d,c)}\|\mathbf{1} + 2Mn\|d_{\Lambda(d,c)}\|\mathbf{1} \geq \mathbf{0}. \tag{7.83}$$

Therefore, $(\check{x} + \bar{d} - d, \tilde{s})$ is feasible to Apx-Opt$(W, \bar{d}, \check{c}, \widetilde{M}, 16MnC_{\mathrm{o}}\hat{\varepsilon})$ for some $\widetilde{M} \geq M$. Further, we already knew that $(\bar{x} + \bar{d} - d, \bar{s}) = (\bar{w}, \bar{s})$ is feasible to Apx-Opt$(W, \bar{d}, \check{c}, M, \bar{\varepsilon})$. Hence, by Lemma 7.7.3 we have that $(\bar{x} + \bar{d} - d, \tilde{s})$ is feasible to Apx-Opt$(W, \bar{d}, \check{c}, \widetilde{M}, \max\{16MnC_{\mathrm{o}}\hat{\varepsilon}, \bar{\varepsilon}\} \cdot 8\sqrt{n}C_{\mathrm{o}})$. Further, note that $\operatorname{supp}(\bar{d} - d) \subseteq \bar{J}$ and that $\operatorname{supp}(\tilde{s}) \subseteq \hat{I}$ and so

$$\operatorname{supp}(\bar{d} - d) \cap \operatorname{supp}(\tilde{s}) \subseteq \bar{J} \cap \hat{I} = \emptyset. \tag{7.84}$$

Therefore,

$$\|\bar{x} \circ \tilde{s}\| \overset{(7.84)}{=} \big\|(\bar{x} + \bar{d} - d) \circ \tilde{s}\big\| \overset{(7.15e)}{\leq} \max\{16MnC_{\mathrm{o}}\hat{\varepsilon}, \bar{\varepsilon}\} \cdot 8\sqrt{n}C_{\mathrm{o}}\|\check{c}\|\|\bar{d}\| \tag{7.85}$$

$$\leq \max\{16MnC_{\mathrm{o}}\hat{\varepsilon}, \bar{\varepsilon}\} \cdot 8\sqrt{n}C_{\mathrm{o}}\|\check{c}\| \cdot 4Mn^{3/2}\|d_{\Lambda(d,c)}\| \tag{7.86}$$

$$\leq \delta\|d_{\Lambda(d,c)}\|\|\check{c}\|, \tag{7.87}$$

contradicting the assumption of Case II. □

7.8 Projection-free black-box algorithms

It is noticeable that the exact linear algebra performed in Algorithms 7.1 to 7.3 can dominate the overall running time for the approximate solvers. The main aim of this section is to describe methods to reduce the computational burden of the exact linear algebra. We could have stated the main algorithms and theorems of the previous sections in terms of the ideas presented in this section, but for ease of presentation opted to postpone their introduction.

We will be able to state the following improved runtimes for both feasibility and optimization algorithms:

Theorem 7.8.1 (Projection-free blackbox algorithms)**.** *Given data* $(\mathbf{A}, d, c)$, $\mathbf{A} \in \mathbb{R}^{m\times n}$, $W := \ker(\mathbf{A})$ *such that* LP$(\mathbf{A}, d, c)$ *has primal and dual feasible solutions, as well as a guess* $M \geq \kappa_{\mathbf{A}}$*. There are algorithms that solve following* LP *problems in the corresponding runtimes.*

Problem	Running time
$x \in \text{Primal}(W, d)$	$\widetilde{O}\big(m(\Psi(\mathbf{A})\log(M) + \mathfrak{L}(\mathbf{A}))\big)$
$s \in \text{Dual}(W^{\perp}, d)$	$\widetilde{O}\big((n-m)(\Psi(\mathbf{A})\log(M) + \mathfrak{L}(\mathbf{A}))\big)$
(x, s) optimal to $\text{LP}(W, d, c)$	$\widetilde{O}\big(m(n-m)(\Psi(\mathbf{A})\log(M) + \mathfrak{L}(\mathbf{A}))\big)$

Note that while the black-box framework itself does not rely on projections, the black-box algorithm themselves might perform such expensive operations.

In Theorem 7.8.1, *approximate* solvers to a *harder* problem (LP) go hand in hand with *exact* solvers to an *easier* problem (Linear System Solve (LSS)). This does not come as a surprise. In real-world problems, approximate solutions to LP can be found efficiently with IPM. The drawback in comparison to Simplex based algorithms is that once an approximately optimal solution is found, the bottleneck is to turn these solutions into optimal basic feasible solutions, see e.g., [GWXY21].

Fast exact linear system solvers. The reduction of the exact linear algebra tasks to linear systems solvers raises the question of how such solvers compare to approximate solvers for LP. Assuming that we are given a basis B, so that $\mathbf{A}_B \in \mathbb{R}^{m\times m}$ is invertible, a long-standing bound has been $\widetilde{O}(m^{\omega})$. This corresponds to just inverting the matrix $\mathbf{A}_B$ and then for a system $\mathbf{A}_B x = b$ solve $x = \mathbf{A}_B^{-1} b$. For sparse matrices this has been improved on in a breakthrough result by Peng and Vempala [PV21], but their result depends on the condition number of $\mathbf{A}$. Techniques that solve linear system in dependency on parameters related to treewidth have been developed in [AY13; LRT79].

7.8.1 Operating in subspaces

The main results of the previous sections are formulated in subspace language. While the initial constraint matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ is given where $W = \ker(\mathbf{A})$,e we operate on subproblems where for some $I \subseteq [n]$ we aim to solve (or approximate) an LP in the spaces W_I and $\pi_I(W)$ with vectors d_I and c_I. Such operations can also be concatenated, i.e, for $J \subseteq I$ we might consider $\pi_J(W_I)$ and $\pi_I(W)_J$. The formulations we demonstrate in this section will be applicable to such chained subspace operations as well. j The easier cases is to consider the space W_I. It is easy to see that this corresponds to the constraint matrix $\mathbf{A}_I$, i.e., $W_I = \ker(\mathbf{A}_I)$. With Assumption 7.4.5 we have $\Psi(\mathbf{A}_I) = O(\Psi(\mathbf{A}))$.

Next, consider the space $\Pi_I(\mathbf{A})$. We could perform matrix manipulation such as finding an invertible matrix $\mathbf{B} \in \mathbb{R}^{m\times m}$ such that the matrix $\overline{\mathbf{A}} := \mathbf{B}\mathbf{A}$ fulfills $\ker(\overline{\mathbf{A}}) = \ker(\mathbf{A})$ and such that $\overline{\mathbf{A}}$ contains a block $\overline{\mathbf{A}}_{K\times I}$ for some $K \subseteq [m]$ such that $\ker(\overline{\mathbf{A}}_{K\times I}) = \pi_I(W)$. But note that on the one

hand computing such a transformation can be expensive and on the other hand the entries of $\overline{\mathbf{A}}_{K\times I}$ might have much higher numerical complexity as well as $\operatorname{nnz}(\overline{\mathbf{A}}_{K\times I}) \gg \operatorname{nnz}(\mathbf{A})$. Furthermore, the treewidth of the matrix may be affected. In general, it could be that $\Psi(\overline{\mathbf{A}}_{K\times I}) \gg \Psi(\mathbf{A})$.

Instead, we opt to not modify the matrix and instead augment the matrix by negative copies of the columns not in I. Namely, for $J \coloneqq [n] \setminus I$, we set $\widehat{\mathbf{A}} \coloneqq \begin{bmatrix} -\mathbf{A}_J & \mathbf{A} \end{bmatrix}$, where with Assumption 7.4.5 we have $\Psi(\widehat{\mathbf{A}}) = O(\Psi(\mathbf{A}))$. We further extend the vectors d to $\widehat{d} \coloneqq (\mathbf{0}_J, d)$ and c to $\widehat{c} \coloneqq (-c_J, c)$. The output (x, s) of a black-box solver with data $(\widehat{\mathbf{A}}, \widehat{d}, \widehat{c})$ can then be converted to a corresponding pair (x, s) of the original system via the canonical coordinate projection on $(\widehat{x}, \widehat{s})$.

7.8.2 Approximating minimum-norm points

Another recurring task in our algorithms is the computation of minimum-norm points in subspaces. That is, for some subspace W with corresponding matrix $\mathbf{A}$ such that $W = \ker(\mathbf{A})$ and a vector d, we want to find $\Pi_{W^\perp}(d)$, i.e., the minimum-norm point in $W + d$. This task is in general expensive. In particular, solving it exactly may not be cheaper than approximately solving an LP. One way of addressing this issue is to reduce the task to LLS. Here, note that if $\operatorname{rk}(\mathbf{A}) = m$ we can express $\Pi_{W^\perp}(d) = \mathbf{A}^\top(\mathbf{A}\mathbf{A}^\top)^{-1}\mathbf{A}d$. Hence, with Assumption 7.4.5, the task of computing the projection of d onto $W^\perp$ can be reduced to a single LLS with the matrix $\mathbf{A}\mathbf{A}^\top$. Unfortunately, it could be in general that $\mathfrak{L}(\mathbf{A}\mathbf{A}^\top) \gg \mathfrak{L}(\mathbf{A})$. A different approach is the following. Note that everywhere were we apply projections, the exact projection is required. Instead, it would suffice to compute a vector $z \in W + d$ such that $\|z\| = \operatorname{poly}(M, n)\|\Pi_{W^\perp}(d)\|$. Algorithms 7.1 to 7.3 would still go through, would such a z provided instead of an exact projection, modulo some polynomial modifications in the values of $C_{\text{po}}^{\text{alg}}$, $C_{\text{pf}}^{\text{alg}}$ and ε. This is the motivation behind the definition of $\mathfrak{P}_{\approx}$. Recall that any basic vector $x^B \in W + d$ for some basis $B \in \mathcal{B}(W)$ has the property that $\|x^B\| = \|\mathbf{A}_B^{-1}\mathbf{A}d\| = \|\mathbf{A}_B^{-1}\mathbf{A}\Pi_{W^\perp}(d)\| \leq \|\mathbf{A}_B^{-1}\mathbf{A}\|\|\Pi_{W^\perp}(d)\| \leq \bar{\chi}_W\|\Pi_{W^\perp}(d)\|$ by Proposition 3.3.3. Hence, an algorithm that finds any basic solution in $W + d$ is an implementation of $\mathfrak{P}_{\approx}(\mathbf{A}, \bar{\chi})$ and therefore for $\mathfrak{P}_{\approx}(\mathbf{A}, n\kappa_W)$ (recall Theorem 3.3.8). Note, that analogously a basic dual solution can be found: Given a basis $B \in \mathcal{B}(W)$ and the affine space $W^\perp + c$, we can find the basic solution s^B corresponding to B via solving $s^B = c - \mathbf{A}^\top\mathbf{A}_B^{-\top}c_B$. If we assume that $\mathfrak{L}(\mathbf{A}) \geq \operatorname{nnz}(\mathbf{A})$, then x^B and s^B can both be computed in time $\mathfrak{L}(\mathbf{A})$.

We also have to take care of the tasks of computing approximate projections for subspaces $W_I + d_I$ and $\pi_I(W) + d_I$, that is finding vectors in the respective spaces whose norm approximates $\|\Pi_{[W_I]^\perp}(d_I)\|$ and $\Pi_{[\pi_I(W)]^\perp}(d_I)$. Again, reformulated as problem over bases, the first task boils down to finding a maximal independent set of variables in I, ignoring the variables in $J = [n] \setminus I$. The second task is the opposite: Find a maximal independent set in J and augment it to a basis in $\mathcal{B}(W)$ by iteratively adding independent columns in I. Now, basic solutions with these bases (and their coordinate projections onto I) correspond again to minimum-norm approximations with multiplicative error $\bar{\chi}$.

With these insights, we want to phrase this implementation of $\mathfrak{P}_{\approx}(\mathbf{A}, n\kappa_W)$ as a sequence of linear system solves. So, the aim is to express $\mathcal{T}(\mathfrak{P}_{\approx}(\mathbf{A}, n\kappa_W))$ as a function of $\mathcal{T}(\mathfrak{L}(\mathbf{A}))$.

If we are given a basis, then we can manage this approximation within a single linear system solve as described above. Otherwise, the bottleneck is to actually compute a basis. There are several algorithms of doing this, see e.g., [CKL13] and references therein. We describe another approach to be able to express the task of finding a basis as a sequence of linear system solves. The algorithm goes as follows: Initially, set $B = \{i\}$ for some arbitrary index i with $\mathbf{A}_i \neq \mathbf{0}$. Now, to augment B by one index, we set $N \coloneqq [n] \setminus B$, use a random vector $u \in R^N$ and solve $\mathbf{A}_B x = \mathbf{A}_N u$ for x. If the system is feasible, then B is a basis with high probability. Otherwise, there exists an

index $j \in N$ such that $\mathbf{A}_j \notin \operatorname{im}(\mathbf{A}_B)$. We aim to find such an index j and add it to B. We then repeat the process until B is a basis. To find such a j, we partition N into two parts of almost even size $N_1 \cup N_2 = N, |N_1| = |N_2| \pm 1$ and solve the two systems $\mathbf{A}_B x = \mathbf{A}_{N_1} u_{N_1}$ and $\mathbf{A}_B x = \mathbf{A}_{N_2} u_{N_2}$ both for x. We keep iterating with one of N_1 or N_2, depending on where the linear system solve failed. This recursion has depth $\log(n)$ and hence it requires $\widetilde{O}(\mathfrak{Q}(\mathbf{A}))$ to augment B by one element. We then repeat the process until B is a basis.

The algorithm described above works correctly with high probability and requires $\widetilde{O}(\operatorname{rk}(\mathbf{A}))$ many linear system solves. Hence, we have that $\mathcal{T}(\mathfrak{P}_{\approx}(\mathbf{A}, \kappa_W)) = O(\operatorname{rk}(\mathbf{A})\mathcal{T}(\mathfrak{Q}(\mathbf{A})))$.

Note that the bottleneck of the algorithm is to find the initial basis. Once the basis is found, we only require a single linear system solve to find an approximate miminum-norm point in the subspace. In the current formulation we compute for every approximate projection the basis from scratch. But the algorithms behave in a dynamic manner in which variables are projected out one-by-one. This can be exploited as we once we have a basis only have to perform a pivot. A pivot can be performed within a single linear system solve. This drops the amortized time per projection to just a single linear system solve.

The idea is the following. Assume we have computed an initial basis B and now in a recursive call to Algorithm 7.1 or Algorithm 7.3 we project onto the coordinates $I \subseteq [n]$. Let $J := [n] \setminus I$. As described above, the task is now to compute a new basis $\widetilde{B}$ such that $\widetilde{B} \cap J$ is maximized among all bases. The naive computation would take $O(|J|) = O(n)$ many linear system solves. We show how to do it in $\widetilde{O}(\operatorname{rk}(J))$ many system solves. First, with the procedure described above we can augment $J \cap B$ to a maximally independent set $K \subseteq J$ within $\widetilde{O}(\operatorname{rk}(\mathbf{A}_{J \setminus B}))$ many linear system solves. Then, for any $k \in K$ we solve the linear system $\mathbf{A}_{B \setminus K} x + \mathbf{A}_{K \setminus \{k\}} y = \mathbf{A}_k$ for x and y. Such a solution must exist, and all solutions fulfill $x \neq \mathbf{0}$. We can then drop any index $i \in \operatorname{supp}(x)$ from B. We repeat this procedure for all $k \in K$ which gives another total of $|K|$ linear system solves.

In particular note, note we have that $\dim(\pi_I(W)^\perp) = \dim(W^\perp) - |K|$. This yields overall $\widetilde{O}(m)$ many linear system solves in both Algorithm 7.1 and Algorithm 7.3.

We also have to be careful as we do not have access to the value of κ_W but only work with guesses M. But this can be handled easily. If the algorithm encounters vectors $x, x^B \in W + d$, where x^B is a basic solution, such that $\|x^B\| > nM\|x\|$, then the vector $x_B - x \in W$ can be used to generate a certificate in $\mathfrak{C}(W, M)$. If on the other hand no such pair of vectors is encountered, then all required guarantees we have for κ_W carry over to M.

Approximating the lifting map. We can easily extend the argumentation above to the lifting map. In the algorithms, we require to compute $L_I^W(w)$ for some $x \in \pi_I(W)$. Again, it suffices for $J := [n] \setminus I$ to find a vector $v \in \mathbb{R}^J$ such that $(v, w) \in W$ and $\|v\| \leq \operatorname{poly}(n, M)\|[L_I^W(w)]_J\|$. This can again be done by finding a maximal independent set $K \subseteq J$ and solve $\mathbf{A}_K x = \mathbf{A}_I w$ for x. As seen above, the naive static approach requires $\widetilde{O}(\operatorname{rk}(\mathbf{A}_J))$ many linear system solves for this task. But note, that whenever we perform such an operation in the algorithms, we are already given a basis B such that $B \cap J$ is maximized among all bases. Hence, $\mathbf{A}_{B \cap J}$ is already a such a full rank submatrix $\operatorname{rk}(\mathbf{A}_{B \cap J}) = \operatorname{rk}(\mathbf{A}_J)$. Hence, a single linear system solve suffices.

7.8.3 Avoiding closure computation

Another expensive task is the computation of the closusure cl of a subset of variables. Recall, that in InnerLoop and Feasibility-Algorithm we compute a set of large coordinates K, compute their closure $\operatorname{cl}(K)$ and recurse on $I := [n] \setminus \operatorname{cl}(K)$. This is to ensure that the dimension decreases on the subspace on which we recurse, more precisely, we ensure that $\dim(\pi_I(W))^\perp < \dim(W^\perp)$. As

the original space fulfilled $\dim(W^\perp) = m$ this gives us a recursion depth of just m instead of n. However, we can circumvent the expensive operation of computing the closure. Assume that we have picked a set of large variables K in either Algorithm 7.1 or Algorithm 7.3 and let $\bar{K} \coloneqq [n] \setminus K$ be the remaining variables. In the notation of the algorithms this would correspond to $K = I \cup J$. Instead of recursing on I via $\pi_I(W)$ we are going to recurse on a set P which will fulfill $I \subseteq P \subseteq K$ and the corresponding subspace $\pi_P(W)$. What does K' need to fulfill? We need that the set K' defined as the set we want to project our recursively while running on $\pi_P(W)$ contains an element that is not contained in $\mathrm{cl}(K)$. This is exactly the condition we require to obtain a recursive depth of at most m.

We will find P as follows. We sort the variables in $\bar{K} \coloneqq \{1, \dots, |\bar{K}|\}$ such that $d_1 \geq \cdots \geq d_{|K|}$ and find the unique index $j \in [\bar{K}]$ such that $\{1, \dots, j-1\} \subseteq \mathrm{cl}(K)$ and $j \notin \mathrm{cl}(K)$ or conclude that $\bar{K} \subseteq \mathrm{cl}(K)$ and set $j = |\bar{K}|$. This can be achieved with binary search with the same algorithm as in Section 7.8.2 in $\log(n)$ linear system solves. We now set $P \coloneqq \{j, \dots, |\bar{K}|\}$.

In the following we will describe why this procedure gives us the desired recursion depth. In the algorithms, we rely on one of two types of arguments why K' is non-empty. We now have to extend these arguments to $K' \setminus \mathrm{cl}(K)$, to show that indeed $K' \setminus \mathrm{cl}(K)$ is non-empty.

The first argument is a *minimum-norm vector* argument. We have, that every vector $z \in \pi_P(W) + d_P$ has norm $\|z\| \geq \|\Pi_{\pi_P(W)^\perp}(d_P)\|$ and by the guarantee of the algorithms this is enough to identify a variable $i \in P$ such that $z_i = \|z\|_\infty$ which is sufficiently large. Now, the variables in $K' \cap \mathrm{cl}(K)$ vanish on minimum-norm vector, that is $[\Pi_{\pi_P(W)^\perp}(d_P)]_{P \cap \mathrm{cl}(K)} = \mathbf{0}$. In particular, for $L \coloneqq P \cap \mathrm{cl}(K)$ and $M \coloneqq P \setminus L$ we have that $\pi_P(W) = \mathbb{R}^L \times \pi_M(W)$ and so we have $\|z_M\| \geq \|\Pi_{\pi_M(W)^\perp}(d_M)\| = \|\Pi_{\pi_P(W)^\perp}(d_P)\|$. Hence, we can identify a variable $i \in M$ such that $z_i = \|z_K\|_\infty \geq \|\Pi^\perp_{pi_I}(d_I)\|$ which is sufficiently large and is hence in K' and can be projected out in the next iteration.

The second argument is a *proximity argument*. The idea is that d_I has norm that is much larger than the guaranteed proximity of the recursive call to the algorithm. That is the black-box call in the algorithn produces $x \in W_P + d_P$ such that $\|x - d_P\|_\infty \ll \|d_P\|_\infty$. But note that by choice of P we have that $\|d_P\|_\infty$ is attained on a variable in $P \setminus \mathrm{cl}(K)$. So, we are able to identify a variable i such that $i \in K' \setminus \mathrm{cl}(K)$.

7.A Applying interior point methods as blackboxes

This section discusses how the recent breakthroughs in IPM for LP can be leveraged to implement Oracle 7.3.

The IPMs we use can be classified by the barrier function that is used.

7.A.1 The standard log-barrier

The papers [CLS19], [Bra20], [JSWZ21] and [DLY21] use the standard log-barrier. As we do in this chapter, all of these papers construct an auxiliary system to be able to initialize the IPM with a near-central point. Therefore, we are not able to use their main theorems directly, but have to extract the behavior of the algorithms in the respective auxiliary systems that they use. [CLS19] as well as [Bra20] uses feasible primal-dual points throughout. Hence, we can use their algorithms directly to obtain the bounds $\Psi(\mathbf{A}) = \widetilde{O}(n^\omega)$ randomized ([CLS19]) as well as deterministically [Bra20] as long as $\omega \geq 2 + 1/6$.

The same holds for the result in [DLY21]. The arxiv version contains Theorem A.1 which shows that an exact feasible point of the auxiliary system is returned. This gives us that $\Psi(\mathbf{A}) = \widetilde{O}(n\mathrm{tw}(\mathbf{A})^2)$.

Deviating from the other solvers, [JSWZ21] uses sketching techniques on the left and as a result their returned points are not feasible. They discuss in Appendix H of the arxiv that they can recover feasible points without decreasing the duality gap at computational cost that is dominated by the algorithm. Hence, we can also apply their results to obtain $\Psi(\mathbf{A}) = \widetilde{O}(n^\omega)$ for a randomized algorithm as long as $\omega \geq 2 + 1/18$.

7.A.2 The Lee-Sidford barrier

The papers [LS19] and [Bra+21] use the Lee-Sidford barrier. For [Bra+21] they maintain a feasible dual solution of the auxiliary program, but to reconstruct a feasible primal solution they require a linear system solve with matrix $\mathbf{ADA}^\top$, where $\mathbf{D}$ is some diagonal matrix and additional cost $\text{nnz}(\mathbf{A})$. This result is stated in Lemma 4.11 of the arxiv version. Note that $\Psi(\mathbf{A}) \geq \text{nnz}(\mathbf{A})$ by assumption, hence this term gets dominated anyway. For the term regarding the linear system solve, not that our overall runtime computes as many linear system solves as calls to the approximate solver. Hence, the reconstruction of a feasible primal solution has costs that are dominated by the black box linear system solves as well. The result of [Bra+21] gives us therefore an overall running time of $\Psi(\mathbf{A}) = \widetilde{O}(mn + m^{2.5})$ randomized.

For [LS19] we get similar results on reconstructing a primal-dual feasible solution with appropriate duality gap to the auxiliary system. Theorem 43 yields a feasible dual point to the system, while Theorem 1 gives us a feasible primal point. These theorems contain a running time parameter $\mathcal{T}_w$ that is similar to $\mathfrak{T}(\mathbf{A})$ defined in this chapter. An earlier paper by the same authors [LS15] shows how to amortize the linear system solves to obtain the claim amortized running time per linear system solve of $\widetilde{O}(\text{nnz}(\mathbf{A}) + m^2)$. Hence, we they obtain the overall randomized algorithm with $\Psi(\mathbf{A}) = \widetilde{O}(\sqrt{m}(\text{nnz}(\mathbf{A}) + m^2))$ that implements Oracle 7.3.

7.B A symmetric initialization system

In this section we want to discuss an alternative to our system for initialization System 7.14 (Initialization-LP). Recall that we chose Initialization-LP to ensure that Ψ for the modified matrix matches Ψ of the original matrix under Assumption 7.4.5. On the other hand, the proofs that we can implement the black-box solvers as required are quite complicated. In this section we present a different system, which turns out to be symmetric on primal and dual side, so guarantees on the primal carry over analogously to the dual side. Furthermore, the system preserves κ_W up to constant factors, which recall is not the case for Initialization-LP. On the downside, the number of constraints on the primal side increases from m to $m + n$, hence, we can not guarantee anymore that Ψ does not increase significantly.

In case that $m = \Theta(n)$ this does not change the asymptotic runtime. We also want to point out that the papers [Bra+21; DLY21] are able to handle additional lower and upper bounds without increasing the runtime.

Throughout, we let $\mathbf{A} \in \mathbb{R}^{m \times n}$, $W = \ker(\mathbf{A})$, and let $b \in \mathbb{R}^m$, $c, d \in \mathbb{R}^n$ such that $\mathbf{A}d = b$. In order to apply interior point methods to (1.1), 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 significantly 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.

$$M_P = 2\|c\|_1 M, \quad M_D = 2\|d\|_1 M\,. \tag{7.88}$$

The system is then defined as follows:

System 7.15. *Symmetric Initialization-LP*

Data: Data $(\mathbf{A}, d, c)$ for an instance of LP with data and values $M_P, M_D \geq 0$.

$$\begin{aligned} \min\ & \langle c, x - \underline{x}\rangle + M_P\langle \mathbf{1}, \underline{x}\rangle & \max\ & \langle y, b\rangle - M_D\langle \mathbf{1}, \bar{s}\rangle \\ & \mathbf{A}x - \mathbf{A}\underline{x} = b & & \mathbf{A}^\top y + s - \bar{s} = c \\ & x - \frac{1}{2}\underline{x} + \bar{x} = M_D\mathbf{1} & & -\mathbf{A}^\top y + \frac{1}{2}\bar{s} + \underline{s} = M_P\mathbf{1} - c \\ & x, \bar{x}, \underline{x} \geq \mathbf{0} & & s, \bar{s}, \underline{s} \geq \mathbf{0}. \end{aligned}$$

$$\begin{aligned} \min\ & \langle c, x - \underline{x}\rangle + M_P\langle \underline{x}, \mathbf{1}\rangle & \max\ & \langle d, c\rangle - \langle d, s - \bar{s}\rangle - M_D\langle \mathbf{1}, \bar{s}\rangle \\ & x - \underline{x} \in W + d & & s - \bar{s} \in W^\perp + c \\ & x - \frac{1}{2}\underline{x} + \bar{x} = M_D\mathbf{1} & & s - \frac{1}{2}\bar{s} + \underline{s} = M_P\mathbf{1} \\ & x, \bar{x}, \underline{x} \geq \mathbf{0} & & s, \bar{s}, \underline{s} \geq \mathbf{0}. \end{aligned}$$

which displays the desired symmetry via $x \sim s$, $\underline{x} \sim \bar{s}$ and $\bar{x} \sim \underline{s}$ in the subspace formulation of System 7.15. In the following we will show that the system can be initialized centrally and that the condition number $\bar{\chi}$ does not increase by too much. Let us begin with the latter.

We denote by $\hat{\mathbf{A}}$ the primal constraint matrix of System 7.15, that is

$$\hat{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & -\mathbf{A} & 0 \\ \mathbf{I} & -\frac{1}{2}\mathbf{I} & \mathbf{I} \end{bmatrix}. \tag{7.89}$$

Lemma 7.B.1. *Let* $\hat{W} := \ker(\hat{\mathbf{A}})$ *for* $\hat{\mathbf{A}}$ *as in* (7.89)*. 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 denotes the i-th vector in the standard basis. We have $g' \in \hat{W}$, and $\operatorname{supp}(g') \subseteq \operatorname{supp}(g)$. 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 $\operatorname{supp}(w') \subsetneq \operatorname{supp}(w)$, 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 $\operatorname{supp}(g') \subsetneq \operatorname{supp}(g)$, 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 $-\frac{1}{2}w_i$. □

Initial solutions While we use the black box results in Theorem 7.4.4, it is worth noting that for interior point methods, the system can be easily initialized near the central path with the following solutions.

$$\begin{aligned}(x, \underline{x}, \bar{x}) &= \frac{2}{3}M_D(\mathbf{1}_n, \mathbf{1}_n, \mathbf{1}_n) + (d, \mathbf{0}_n, -d)\\ (y, s, \underline{s}, \bar{s}) &= \frac{2}{3}M_P(\mathbf{0}_m, \mathbf{1}_n, \mathbf{1}_n, \mathbf{1}_n) + (\mathbf{0}_m, \mathbf{0}_n, -\frac{1}{2}c, -c)\end{aligned} \tag{7.90}$$

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

7.C Miscellaneous

Lemma 7.C.1. *Let $\mathbf{A} \in \mathbb{R}^{m\times n}$, $W = \ker(\mathbf{A})$ and $I \subseteq [n]$. Then, both of the following can be computed in time $O(\min\{m^2n, n^\omega\})$.*

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

(ii) $L_I^{W^\perp}(q)$, 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 $\mathbf{A}_{[n]\setminus I}x = -\mathbf{A}_I p$ 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 $W_{[n]\setminus I} = \ker(\mathbf{A}_{[n]\setminus I})$. So, for $\mathbf{B} \coloneqq \mathbf{A}_{[n]\setminus I}$ we have

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

where $\mathbf{B}\mathbf{B}^\top \in \mathbb{R}^{m\times m}$ 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 $W^\perp = \operatorname{im}(\mathbf{A}^\top)$ and so we find $y \in \mathbb{R}^m$ such that $(\mathbf{A}_I)^\top y = q$. This can again be done in time $O(\min\{m^2n, n^\omega\})$. Then, we can set $\hat{w} = \mathbf{A}^\top y \in W^\perp$. 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 $\hat{\mathbf{A}}$, 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

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

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

8 Circuit diameter

We study the circuit diameter of polyhedra, introduced by Borgwardt, Finhold, and Hemmecke [BFH15] as a relaxation of the combinatorial diameter. We show that the circuit diameter of a system $\{ x \in \mathbb{R}^n : \mathbf{A}x = b, \mathbf{0} \leq x \leq u \}$ (i.e., System 1.3) for $\mathbf{A} \in \mathbb{R}^{m \times n}$ is bounded as $O(m^2 \log(m + \kappa_{\mathbf{A}}) + n \log n)$, where $\kappa_{\mathbf{A}}$ is the circuit imbalance measure of the constraint matrix. This yields a strongly polynomial circuit diameter bound e.g., if all entries of $\mathbf{A}$ have polynomially bounded encoding length in n. Further, we present circuit augmentation algorithms for LPs using the minimum-ratio circuit cancelling rule. Even though the standard minimum-ratio circuit cancelling algorithm is not finite in general, our variant can solve an optimization LP in $O(n^3 \log(n + \kappa_{\mathbf{A}}))$ augmentation steps. In this tune, we also prove an improved bound on the convergence of steepest-descent augmentation.

This chapter is based on joint work with Daniel Dadush, Farbod Ekbatani, Zhuan Khye Koh, and László A. Végh [DKNV22; ENV22].

Contents

8.1 Introduction

For a system $\{ x \in \mathbb{R}^d : \mathbf{M}x \leq d \}$ with integer constraint matrix $\mathbf{M}$, polynomial diameter bounds were given in terms of the maximum subdeterminant $\Delta_{\mathbf{M}}$ [Bon+14; BR13; DH16; EV17]. These arguments can be strengthened to using a parametrization by a 'discrete curvature measure' $\delta_{\mathbf{M}} \geq 1/(n\Delta_{\mathbf{M}}^2)$ (see Chapter 3). The best such bound was given by Dadush and Hähnle [DH16] as $O(d^3 \log(d/\delta_{\mathbf{M}})/\delta_{\mathbf{M}})$, using a shadow vertex simplex algorithm.

As a natural relaxation of the combinatorial diameter, Borgwardt, Finhold, and Hemmecke [BFH15] initiated the study of circuit diameters. Consider a polytope in the standard equality form

$$P = \{ x \in \mathbb{R}^n : \mathbf{A}x = b, x \geq \mathbf{0} \} \tag{P}$$

for $\mathbf{A} \in \mathbb{R}^{m \times n}$, $b \in \mathbb{R}^n$. This corresponds to the feasible point of Primal$(\mathbf{A}, b, \cdot)$. All edge directions of P are elementary vectors, and the set of elementary vectors $\mathcal{E}(\mathbf{A})$ equals the set of all possible edge directions of P in the form (P) for varying $b \in \mathbb{R}^n$ [ST97].

Recall the definition of a circuit walk as a set of consecutive points $x^{(1)}, x^{(2)}, \ldots, x^{(k+1)} \in P$ such that for each $i = 1, \ldots, k$, $x^{(i+1)} = x^{(i)} + g^{(i)}$ for $g^{(i)} \in \mathcal{E}(\mathbf{A})$, and further, $x^{(i)} + (1 + \varepsilon)g^{(i)} \notin P$ for any $\varepsilon > 0$, i.e., each consecutive circuit step is *maximal*. The *circuit diameter* of P is the minimum length of a circuit walk between any two vertices $x, y \in P$. Note that, in contrast to walks in the vertex-edge graph, circuit walks are non-reversible and the minimum length from x to y may be different from the one from y to x; this is due to the maximality requirement. The circuit-analogue of the Hirsch conjecture, formulated in [BFH15], asserts that the circuit diameter of a polytope in d dimensions with n facets is at most $n - d$; this may be true even for unbounded polyhedra, see [BSY18]. For P in the form (P), $d = n - m$ and hence the conjectured bound is m.

Circuit diameter bounds have been shown for some combinatorial polytopes such as dual transportation polyhedra [BFH15], matching, travelling salesman, and fractional stable set polytopes [KPS19]. The paper [BDF16] introduced several other variants of the circuit diameter, and explored the relation between them.

Circuit augmentation algorithms Circuit diameter bounds are inherently related to circuit augmentation algorithms. This is a general algorithmic scheme to solve System 1.1

$$\min \langle c, x \rangle \quad \text{s.t.} \quad \mathbf{A}x = b \,,\; x \geq \mathbf{0} \,. \tag{LP}$$

The algorithm proceeds through a sequence of feasible solutions $x^{(t)}$. An initial feasible $x^{(0)}$ is required in the input. For $t = 0, 1, \ldots,$ the current $x^{(t)}$ is updated to $x^{(t+1)} = x^{(t)} + \alpha g$ for some $g \in \mathcal{E}(\mathbf{A})$ such that $\langle c, g \rangle \leq 0$, and $\alpha > 0$ such that $x^{(t)} + \alpha g$ is feasible. The elementary vector g is an *augmenting direction* if $\langle c, g \rangle < 0$ and such an $\alpha > 0$ exists; by LP duality, $x^{(t)}$ is optimal if and only if no augmenting direction exists. The augmentation is *maximal* if $x^{(t)} + \alpha' g$ is infeasible for any $\alpha' > \alpha$; α is called the maximal stepsize for $x^{(t)}$ and g. An upper bound on the number of steps of a circuit augmentation algorithm with maximal augmentations for arbitrary cost c and starting point $x^{(0)}$ yields an upper bound on the circuit diameter.

The simplex algorithm can be seen as a circuit augmentation algorithm that is restricted to using special elementary vectors corresponding to edges of the polyhedron.[1] For the general framework, the iterates $x^{(k)}$ may not be vertices. However, in case of maximal augmentations, they must all lie on the boundary of the polyhedron.

In unpublished work, Bland [Bla76] extended the Edmonds–Karp–Dinic algorithm [Din70; EK72] algorithm for general LP, see also [Lee89, Proposition 3.1]. Circuit augmentation algorithms were revisited by De Loera, Hemmecke, and Lee in 2015 [DHL15], analyzing different augmentation rules and extending them to integer programming. We give an overview of their results first for linear programming. In particular, they studied three augmentation rules that use maximal

[1] Simplex may contain degenerate pivots when the basic solution remains the same; we do not count these as augmentation steps.

augmentation. Let $x^{(t)}$ be the current feasible solution, and we aim to select an augmenting direction g as follows.

- *Dantzig-descent direction:* Select g such that $-\langle c, g\rangle$ is maximized, where $g = g^C$ is the elementary vector with $\operatorname{lcm}(g^C) = 1$ for a circuit $C \in \mathcal{C}_W$.
- *Deepest-descent direction:* Select g such that $-\alpha\langle c, g\rangle$ is maximized, where α is the maximal stepsize for $x^{(t)}$ and g.
- *Steepest-descent direction:* Select g such that $-\langle c, g\rangle/\|g\|_1$ is maximized.

Computing Dantzig- and deepest-descent directions is in general NP-hard, see [LKS19] and as detailed below. The steepest-descent direction can be formulated by an LP; but without any restrictions on the input problem, this may not be simpler than the original one. However, it could be easier to solve in practice; Borgwardt and Viss [BV20] exhibits an implementation of a steepest-descent circuit augmentation algorithm with encouraging computational results.

Augmenting directions for flow problems

It is instructive to consider these algorithms for the special case of *minimum-cost flows*. Given a directed graph $D = (V, E)$ with capacities $u \in \mathbb{R}^E$, costs $c \in \mathbb{R}^E$, and node demands $b \in \mathbb{R}^V$ with $b(V) = \sum_{i\in V} b_i = 0$. The objective is to find the minimum cost flow x that satisfies the capacity constraints: $0 \le x \le u$, and the node demands: for each node $i \in V$, the total incoming minus the total outgoing flow equals b_i. This can be written in the form Capacitated-LP with $\mathbf{A}$ as the node-arc incidence matrix of D, a TU matrix. Let us define the *residual graph* $D_x = (V, E_x)$, where for $(i, j) \in E_x$ we let $(i, j) \in E$ if $x_{ij} < u_{ij}$ and $(j, i) \in E$ if $x_{ij} > 0$. The cost of a reverse arc will be defined as $c_{ji} = -c_{ij}$. We will also refer to the *residual capacities* of arcs; these are $u_{ij} - x_{ij}$ in the first case and x_{ij} in the second.

Let us observe that the augmenting directions correspond to directed cycles in the residual graph. Circuit augmentation algorithms for the primal and dual problems yield the rich classes of cycle cancelling and cut cancelling algorithms, see the survey [SIM00].

The *maximum flow problem* between a source s and sink t can be formulated as a special case as follows. We add a new arc (t, s) with capacity ∞, set the demands $b \equiv 0$, and costs as $c_{ts} = -1$ and $c_{ij} = 0$ otherwise. Bland's [Bla76] observation was that the steepest-descent direction for this problem corresponds to finding a shortest residual s-t path, as chosen in the Edmonds–Karp–Dinic algorithm.

More generally, a steepest-descent direction amounts to finding a residual cycle $C \subseteq E_x$ that minimizes the mean cycle cost $c(C)/|C|$. Thus, the steepest descent algorithm for minimum-cost flows corresponds to the classical Goldberg–Tarjan algorithm [GT89] that is strongly polynomial with running time $O(|V| \cdot |E|^2)$ [RG94].

Let us now consider the other two variants. A Dantzig-descent direction in this context asks for the most negative cycle, i.e., a cycle maximizing $-c(C)$. A deepest-descent direction asks for a cycle C of arcs that maximizes $-\alpha c(C)$, where α is the residual capacity of C. Computing both these directions exactly is NP-complete, since they generalize the Hamiltonian-cycle problem: for every directed graph, we can set up a flow problem where E_x coincides with the input graph, all residual capacities are equal to 1, and all costs are -1. We note that De Loera, Kafer, and Sanità [LKS19] showed that computing the Dantzig- and deepest-descent directions is also NP-hard for the fractional matching polytope.

Nevertheless, the deepest-descent direction can be suitably approximated. Wallacher [Wal89] proposed selecting a *minimum ratio cycle* in the residual graph. This is a cycle in E_x that minimizes

$c(C)/d(C)$, where $d_e = 1/u_e$ for every residual arc $e \in E_x$; such a cycle can be found in strongly polynomial time. It is easy to show that this cycle approximates the deepest descent direction within a factor $|E_x|$. Wallacher's algorithm can be naturally extended to linear programming [MS00], and has found several combinatorial applications, e.g. [Way02; WZ99], and has also been used in the context of integer programming [SW99]. We discuss an improved new variant in Section 8.6. A different relaxation of the deepest-descent algorithm was given by Barahona and Tardos [BT89], based on Weintraub's algorithm [Wei74].

Convergence bounds

We now state the convergence bounds from [DHL15]. The original statement refers to subdeterminant bounds; we paraphrase them in terms of finding approximately optimal solutions.

Theorem 8.1.1 (De Loera, Hemmecke, Lee [DHL15]). *Consider a linear program in the form Capacitated-LP. Assume we are given an initial feasible solution $x^{(0)}$, and let OPT denote the optimum value. By an ε-optimal solution we mean an iterate $x^{(t)}$ such that $\langle c, x^{(t)} \rangle \leq \mathrm{OPT} + \varepsilon$.*

(a) For given $\varepsilon > 0$, one can find an ε-optimal solution in $2n \log_2\left(\frac{\langle c,x^{(0)}\rangle - \mathrm{OPT}}{\varepsilon}\right)$ deepest-descent augmentations.

(b) For given $\varepsilon > 0$, one can find an ε-optimal solution in $\frac{2n^2\gamma}{\varepsilon} \log_2\left(\frac{\langle c,x^{(0)}\rangle - \mathrm{OPT}}{\varepsilon}\right)$ Dantzig-descent augmentations, where γ is an upper bound on the maximum entry in any feasible solution.

(c) One can find an exact optimal solution in $\min\{n|C_{\mathbf{A}}|, \ell_{\mathbf{A}}\}$ steepest-descent augmentations, where $\ell_{\mathbf{A}}$ denotes the number of distinct values of $\langle c, g\rangle/\|g\|_1$ over $g \in \mathcal{E}(\mathbf{A})$.

In general, circuit augmentation algorithms may not even finitely terminate; see [MS00] for an example on Wallacher's rule for minimum cost flows. In parts (a) and (b), assume that all basic solutions are $1/k$-integral for some $k \in \mathbb{Z}$ and cost function is $c \in \mathbb{Z}^n$. If $x^{(t)}$ is a ε-optimal solution for $\varepsilon < 1/k$, then we can identify an optimal vertex of the face containing $x^{(t)}$ using a Carathéodory decomposition argument, this can be implemented by a sequence of $\leq n$ circuit augmentations (see [DHL15, Lemma 5]).

According to part (c), steepest descent terminates with an optimal solution in a finite number of iterations; moreover, the bound only depends on the linear space $\ker(A)$ and c, and not on the parameters b and u. However, the bound can be exponentially large.

Bland's original observation was that $\ell_{\mathbf{A}}$ is strongly polynomially bounded for the maximum flow problem. Recall that all elementary vectors g correspond to cycles in the auxiliary graph. Normalizing such that $g_i \in \{0, \pm 1\}$, $-\langle c, g\rangle = 1$ for every augmenting cycle (as these must use the (t, s) arc), and $\|g\|_1$ is between 1 and $|E|$. In fact, the crucial argument by Edmonds and Karp [EK72] and Dinic [Din70] is showing that the length of the shortest augmenting path is non-decreasing, and must strictly increase within $|E|$ consecutive iterations.

For an integer cost function $c \in \mathbb{Z}^n$, Lee [Lee89, Proposition 3.2] gave the following upper bound on $\ell_{\mathbf{A}}$:

Proposition 8.1.2. *If $\|c\|_1 \leq (n - m + 1)\|c\|_\infty$, then*

$$\ell_{\mathbf{A}} \leq \frac{1}{2}\|c\|_\infty (n - m + 1)\bar{\kappa}_{\mathbf{A}}((n - m + 1)\bar{\kappa}_{\mathbf{A}} + 1)\,.$$

In order to bound the circuit distance between vertices x and y let us use the following cost function. For the basis B defining y, let

$$c_i = \begin{cases} 0 & \text{if } i \in B\,, \\ 1 & \text{if } i \in [n] \setminus B, y_i = 0\,, \\ -1 & \text{if } i \in [n] \setminus B, y_i = u_i\,. \end{cases} \tag{8.1}$$

With this cost function, Theorem 8.1.1(c) and Proposition 8.1.2 yield a bound $O((n-m)^2\bar{\kappa}_{\mathbf{A}}^2)$ on the circuit diameter using the steepest descent algorithm.

Extending the analysis of the Goldberg-Tarjan algorithm [GT89], we present a new bound that only depends on the fractional circuit imbalance $\kappa_{\mathbf{A}}$, and is independent of c. The same bound was independently obtained by Gauthier and Derosiers [GD21]. The proof is given in Section 8.7.

Theorem 8.1.3. *For the problem Capacitated-LP with constraint matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, the steepest-descent algorithm terminates within $O(n^2 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$ augmentations starting from any feasible solution $x^{(0)}$.*

This improves on the above bound $O((n-m)^2\bar{\kappa}_{\mathbf{A}}^2)$ for most values of the parameters (recall that $\kappa_{\mathbf{A}} \le \bar{\kappa}_{\mathbf{A}}^2$). Moreover, this bounds the running time for steepest descent for an arbitrary cost function c, not necessarily of the form (8.1).

Both these bounds are independent of b, however, $\kappa_{\mathbf{A}}$ and $\bar{\kappa}_{\mathbf{A}}$ may be exponentially large in the encoding length $L_{\mathbf{A}}$ of the matrix $\mathbf{A}$. In contrast, Theorem 8.1.1(a) yields a polynomial bound $O(nL_{A,b})$ on the number of deepest-descent iterations, where $L_{A,b}$ is the encoding length of (A, b). In what follows, we review a new circuit augmentation algorithm from [DKNV22] that achieves a $\log \kappa_{\mathbf{A}}$ dependence; the running time is bounded as $O(n^3 L_{\mathbf{A}})$, independently from b.

Imbalance and diameter The combinatorial diameter bound $O(d^3 \log(d/\delta_{\mathbf{M}})/\delta_{\mathbf{M}})$ from [DH16] mentioned above translates to a bound $O((n-m)^3 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$ for the system in the form (P), see [ENV22]. For circuit diameters, the Goldberg-Tarjan minimum-mean cycle cancelling algorithm for minimum-cost flows [GT89] naturally extends to a circuit augmentation algorithm for general LP using the steepest-descent rule. This yields a circuit diameter bound $O(n^2 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}}+n))$ [ENV22], see also [GD21]. However, note that these bounds may be exponential in the bit-complexity of the input.

8.1.1 Our contributions

Our first main contribution improves the $\kappa_{\mathbf{A}}$ dependence to a $\log \kappa_{\mathbf{A}}$ dependence for circuit diameter bounds.

Theorem 8.1.4. *The circuit diameter of a system in the form (P) with constraint matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ is $O(m^2 \log(m + \kappa_{\mathbf{A}}))$.*

The proof in Section 8.3 is via a simple 'shoot towards the optimum' scheme. Recall the concept of conformal circuit decompositions.

Consider a feasible basis $B \subseteq [n]$ and $N = [n] \setminus B$ such that $x^* = (\mathbf{A}_B^{-1} b, \mathbf{0}_N) \ge \mathbf{0}_n$ is a basic feasible solution. This is the unique optimal solution to (LP) for the cost function $c = (\mathbf{0}_B, \mathbf{1}_N)$. For the current iterate $x^{(t)}$, let us consider a conformal circuit decomposition $x^* - x^{(t)} = \sum_{j=1}^{k} h^{(j)}$. Note that the existence of such a decomposition *does not* yield a circuit diameter bound n due to the maximality requirement in the definition of circuit walks. For each $j \in [k]$, $x^{(t)} + h^{(j)} \in P$, but there might be a larger augmentation $x^{(t)} + \alpha h^{(j)} \in P$ for $\alpha > 1$.

Still, one can use this decomposition to construct a circuit walk. Let us pick the most improving circuit from the decomposition, i.e., the one maximizing $-\langle c, h^{(j)} \rangle = \|h_N^{(j)}\|_1$, and obtain $x^{(t+1)} = x^{(t)} + \alpha^{(t)} h^{(j)}$ for the maximum stepsize $\alpha^{(t)} \geq 1$. The proof of Theorem 8.1.4 is based on analyzing this procedure. The first key observation is that $\langle c, x^{(t)} \rangle = \|x_N^{(t)}\|_1$ decreases geometrically. Then, we look at the set of indices $L_t = \left\{ i \in [n] : x_i^* > n\kappa_{\mathbf{A}} \|x_N^{(t)}\|_1 \right\}$ and $R_t = \left\{ i \in [n] : x_i^{(t)} \leq n x_i^* \right\}$, and show that indices may never leave these sets once they enter. Moreover, a new index is added to either set every $O(m \log(m + \kappa_{\mathbf{A}}))$ iterations. In Section 8.4, we extend this bound to the setting with upper bounds on the variables.

Theorem 8.1.5. *The circuit diameter of a system in the form* $\mathbf{A}x = b$, $\mathbf{0} \leq x \leq u$ *with constraint matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *is* $O(m^2 \log(m + \kappa_{\mathbf{A}}) + n \log n)$.

There is a straightforward reduction from the capacitated form to (P) by adding n slack variables; however, this would give an $O(n^2 \log(n + \kappa_{\mathbf{A}}))$ bound. For the stronger bound, we use a preprocessing that involves cancelling circuits in the support of the current solution; this eliminates all but $O(m)$ of the capacity bounds in $O(n \log n)$ iterations, independently from $\kappa_{\mathbf{A}}$.

For rational input, $\log(\kappa_{\mathbf{A}}) = O(L_{\mathbf{A}})$ where $L_{\mathbf{A}}$ denotes the total encoding length of $\mathbf{A}$ [DHNV20]. Hence, our result yields an $O(m^2 L_{\mathbf{A}} + n \log n)$ diameter bound on $\mathbf{A}x = b$, $0 \leq x \leq u$. This can be compared with the bound $O(n L_{\mathbf{A},b})$ using deepest descent augmentation steps in [DHL15], where $L_{\mathbf{A},b}$ is the encoding length of $(\mathbf{A}, b)$. (Such a bound holds for every augmentation rule that decreases the optimality gap geometrically, including the minimum-ratio circuit rule discussed below.) Besides our bound being independent of b, it is also applicable to systems given by irrational inputs.

In light of these results, the next important step towards the polynomial Hirsch conjecture might be to show a $\text{poly}(n, \log \kappa_{\mathbf{A}})$ bound on the combinatorial diameter of (P). Note that—in contrast with the circuit diameter—not even a $\text{poly}(n, L_{\mathbf{A},b})$ bound is known. In this context, the best known general bound is $O((n-m)^3 m \kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$ implied by [DH16].

Circuit augmentation algorithms The diameter bounds in Theorems 8.1.4 and 8.1.5 rely on knowing the optimal solution x^*; thus, they do not provide efficient LP algorithms. We next present circuit augmentation algorithms with $\text{poly}(n, m, \log \kappa_{\mathbf{A}})$ bounds on the number of iterations. Such algorithms require subroutines for finding augmenting circuits. In many cases, such subroutines are LPs themselves. However, they may be of a simpler form, and might be easier to solve in practice. Borgwardt and Viss [BV20] exhibits an implementation of a steepest-descent circuit augmentation algorithm with encouraging computational results.

Our main subroutine assumption Ratio-Circuit$(\mathbf{A}, c, w)$ is the well-known minimum-ratio circuit rule. It takes as input a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $c \in \mathbb{R}^n$, $w \in (\mathbb{R}_+ \cup \{\infty\})^n$, and returns a basic optimal solution to the following system; this can be equivalently written as an LP using auxiliary variables.

$$\min \langle c, z \rangle \quad \text{s.t.} \quad \mathbf{A}z = \mathbf{0}\,, \ \langle w, z^- \rangle \leq 1\,. \tag{8.2}$$

If bounded, a basic optimal solution will be a circuit z that minimizes $\langle c, z \rangle / \langle w, z^- \rangle$.

Given a feasible $x \in P$ for (P), we use weights $w_i = 1/x_i$ (and uses $w_i = \infty$ if $x_i = 0$). For minimum-cost flow problems, this rule was proposed by Wallacher [Wal89]; such a cycle can be found in strongly polynomial time for flows. The main advantage of this rule is that the optimality gap decreases by a factor $1 - 1/n$ in every iteration. This rule, along with the same convergence property, can be naturally extended to linear programming [MS00], and has found several combinatorial applications, e.g. [Way02; WZ99], and has also been used in the context of integer programming [SW99].

On the negative side, Wallacher's algorithm is *not* strongly polynomial: it does not even terminate finitely for minimum-cost flows, as shown in [MS00]. In contrast, our algorithms achieve a strongly polynomial running time whenever $\kappa \leq 2^{\text{poly}(n)}$. An important modification is the occasional use of a second type of circuit augmentation step Support-Circuit that removes circuits in the support of the current (non-basic) iterate $x^{(t)}$ (see Subroutine 8.1); this can be implemented using simple linear algebra. Our first result addresses the feasibility setting:

Theorem 8.1.6. *Consider an LP of the form (P) with cost function $c = (\mathbf{0}_{[n]\setminus N}, \mathbf{1}_N)$ for some $N \subseteq [n]$. There exists a circuit augmentation algorithm that either finds a solution x such that $x_N = 0$ or a dual certificate that no such solution exists, using $O(n^2 \log(n + \kappa_{\mathbf{A}}))$ Ratio-Circuit and n^2 Support-Circuit augmentation steps.*

Such problems typically arise in Phase I of the Simplex method when we add auxiliary variables in order to find a feasible solution. The algorithm is presented in Section 8.5. The analysis extend that of Theorem 8.1.4, tracking large coordinates $x_i^{(t)}$. Our second result considers general optimization:

Theorem 8.1.7. *Consider an LP of the form (LP). There exists a circuit augmentation algorithm that finds an optimal solution or concludes unboundedness using $O(n^3 \log(n + \kappa_{\mathbf{A}}))$ Ratio-Circuit and n^3 Support-Circuit augmentation steps.*

The proof is given in Section 8.6. The main subroutine identifies a new index $i \in [n]$ such that $x_i^{(t)} = 0$ in the current iteration and $x_i^* = 0$ in an optimal solution; we henceforth fix this variable to 0. To derive this conclusion, at the end of each phase the current iterate $x^{(t)}$ will be optimal to (LP) with a slightly modified cost function $\tilde{c}$; the conclusion follows using a proximity argument (Theorem 8.2.5). The overall algorithm repeats this subroutine n times. The subroutine is reminiscent of the feasibility algorithm (Theorem 8.1.6) with the following main difference: whenever we identify a new 'large' coordinate, we slightly perturb the cost function.

Comparison to black-box LP approaches An important milestone towards strongly polynomial linear programming was Tardos's 1986 paper [Tar86] on solving (LP) in time $\text{poly}(n, m, \log \Delta_{\mathbf{A}})$, where $\Delta_{\mathbf{A}}$ is the maximum subdeterminant of $\mathbf{A}$. Her algorithm makes $O(nm)$ calls to a weakly polynomial LP solver for instances with small integer capacities and costs, and uses proximity arguments to gradually learn the support of an optimal solution. This approach was extended to the real model of computation for an $\text{poly}(n, m, \log \kappa_{\mathbf{A}})$ bound [DNV20]. This result uses proximity arguments with circuit imbalances $\kappa_{\mathbf{A}}$, and eliminates all dependence on bit-complexity.

The proximity tool Theorem 8.2.5 derives from [DNV20], and our circuit augmentation algorithms are inspired by the feasibility and optimization algorithms in this paper. However, using circuit augmentation oracles instead of an approximate LP oracles changes the setup. Our arguments become simpler since we proceed through a sequence of feasible solutions, whereas much effort in [DNV20] is needed to deal with infeasibility of the solutions returned by the approximate solver. On the other hand, we need to be more careful as all steps must be implemented using circuit augmentations in the original system, in contrast to the higher degree of freedom in [DNV20] where we can make approximate solver calls to arbitrary projections and modifications of the input LP.

8.2 Preliminaries

For P as in (P), $x \in P$ and an elementary vector $g \in \mathcal{E}(\mathbf{A})$, we let $\mathrm{aug}_P(x, g) := x + \alpha g$ where $\alpha = \arg\max\{\, \bar{\alpha} : x + \bar{\alpha} g \in P \,\}$.

Recall that for a linear space $W \subseteq \mathbb{R}^n$ and $z \in W$, there exists a conformal circuit decomposition of z into at most n circuits. That is, we can write $z = \sum_{t=1}^{k} h^{(t)}$ such that $k \leq n$, and $h^{(t)} \in \mathcal{E}(W)$ and $h^{(t)} \sqsubseteq z$ for all $t \in [k]$ (Lemma 2.0.3).

8.2.1 Circuit oracles

In Sections 8.4 to 8.6, we use a simple circuit finding subroutine Support-Circuit$(\mathbf{A}, c, x, S)$ that will be used to identify circuits in the support of the solution. This can be implemented easily using Gaussian elimination.

Subroutine 8.1. *Support-Circuit*$(\mathbf{A}, c, x, S)$

For a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, vectors $c, x \in \mathbb{R}^n$ and $S \subseteq [n]$, the output is an elementary vector $z \in \mathcal{E}(\mathbf{A})$ with $\mathrm{supp}(z) \subseteq \mathrm{supp}(x)$, $\mathrm{supp}(z) \cap S \neq \emptyset$ with $\langle c, z \rangle \leq 0$, or concludes that no such circuit exists.

The circuit augmentation algorithms in Sections 8.5 and 8.6 will use the subroutine Ratio-Circuit$(\mathbf{A}, c, w)$.

Subroutine 8.2. *Ratio-Circuit*$(\mathbf{A}, c, w)$

The input is a matrix $\mathbf{A} \in \mathbb{R}^{m \times n}$, $c \in \mathbb{R}^n$, $w \in (\mathbb{R}_+ \cup \{\infty\})^n$, and returns a basic optimal solution to the following system:

$$\min \langle c, z \rangle \quad \text{s.t.} \quad \mathbf{A}z = \mathbf{0}\,,\ \langle w, z^- \rangle \leq 1\,. \tag{8.3}$$

And a basic optimal solution (y, s) to the following dual program:

$$\max\ -\lambda \quad \text{s.t.} \quad s = c + \mathbf{A}^\top y \quad \mathbf{0} \leq s \leq \lambda w \tag{8.4}$$

We remark that $w_i z_i^- = 0$ whenever $w_i = \infty$ and $z_i^- = 0$ in (8.3). Note that (8.3) can be reformulated as an LP using additional variables, and the dual LP can be equivalently written as (8.4). If (8.3) is bounded, then a basic optimal solution is an elementary vector $z \in \mathcal{E}(\mathbf{A})$ that minimizes $\langle c, z \rangle / \langle w, z^- \rangle$. Further, note that every feasible solution to (8.4) is also feasible to the dual of (LP). The following lemma is well-known, see e.g., [MS00, Lemma 2.2].

Lemma 8.2.1. *Let* OPT *denote the optimum value of* (LP). *Given a feasible solution* x *to* (LP), *let* g *be the elementary vector returned by* Ratio-Circuit$(A, c, 1/x)$, *and* $x' = \mathrm{aug}_P(x, g)$. *Then,*

$$\langle c, x' \rangle - \mathrm{OPT} \leq \left(1 - \frac{1}{n}\right)(\langle c, x \rangle - \mathrm{OPT})\,.$$

Furthermore, $\alpha \geq 1$ *for the augmentation step.*

Proof. Let x^* be an optimal solution to (LP), and let $z = (x^* - x)/n$. Then, z is feasible to (8.3) for $w = 1/x$. The claim easily follows by noting that $\langle c, g \rangle \leq \langle c, z \rangle = (\mathrm{OPT} - \langle c, x \rangle)/n$, and noting that $x + g \in P$ is implied by $\langle 1/x, g^- \rangle \leq 1$. □

8.2.2 Proximity results

The condition number $\kappa_{\mathbf{A}}$ is mainly used for proving norm bounds that can be interpreted as special forms of Hoffman-proximity results. We formulate such statements that will be needed for our analyses. These can be derived from more general results in [DNV20]; see also [ENV22]. The references also explain the background and similar results in previous literature, in particular, to proximity bounds via $\Delta_{\mathbf{A}}$ in e.g. [Tar86] and [CGST86]. For completeness, we include the proofs.

Lemma 8.2.2. *For $\mathbf{A} \in \mathbb{R}^{m\times n}$, let $z \in \ker(\mathbf{A})$, and let $N \subseteq [n]$ such that $\mathbf{A}_{[n]\setminus N}$ has full column rank. Then, $\|z\|_\infty \le \kappa_{\mathbf{A}} \|z_N\|_1$.*

Proof. Consider a conformal circuit decomposition $h^{(1)}, \dots, h^{(k)}$ of z. By the full column-rank assumption of $\mathbf{A}_{[n]\setminus N}$, for every $h^{(t)}$, $N \cap \operatorname{supp}(h^{(t)}) \neq \emptyset$. Consequently, $\|h^{(t)}\|_\infty \le \kappa_{\mathbf{A}} |h^{(t)}_j| \le \kappa_{\mathbf{A}} |z_j|$ for some $j \in N$. The claim on $\|z\|_\infty = \sum_{t=1}^k \|h^{(t)}\|_\infty$ follows from this using the conformity of the decomposition. □

Lemma 8.2.3. *For $\mathbf{A} \in \mathbb{R}^{m\times n}$, let $z - x \in \ker(\mathbf{A})$, and let $T \subseteq [n]$ such that $[n] \setminus T \subseteq \operatorname{supp}(x)$, and there is no circuit in $\operatorname{supp}(x)$ intersecting T. Then,*

$$(\kappa_{\mathbf{A}} + 1)\|z_T\|_1 \ge \|x_T\|_\infty \quad \text{and} \quad (n\kappa_{\mathbf{A}} + 1)\|z_T\|_1 \ge \|x_T\|_1 .$$

Proof. Consider a conformal circuit decomposition $h^{(1)}, \dots, h^{(k)}$ of $z - x$. By the assumption, for every $h^{(t)}$ with $T \cap \operatorname{supp}(h^{(t)}) \neq \emptyset$, there exists an index $j \in T \cap \operatorname{supp}(h^{(t)})$ such that $x_j = 0 < z_j$. This shows that $\|h^{(t)}_T\|_\infty \le \kappa_{\mathbf{A}} z_j$; moreover, by the conformity of the decomposition, $\|z_T - x_T\|_\infty = \sum_{t=1}^k \|h^{(t)}_T\|_\infty \le \kappa_{\mathbf{A}} \|z_T\|_1$. This implies the first claim. The second claim follows since $\|x_T\|_1 \le \|z_T\|_1 + \|z_T - x_T\|_1 \le \|z_T\|_1 + n\|z_T - x_T\|_\infty$. □

For $T = [n]$, we obtain the following corollary.

Corollary 8.2.4. *Let x be a basic (not necessarily feasible) solution to (LP). Then, for any z where $\mathbf{A}z = b$, we have $(\kappa_{\mathbf{A}} + 1)\|z\|_1 \ge \|x\|_\infty$.*

The following proximity theorem will be key to derive $x^*_i = 0$ for certain variables in our optimization algorithm; see [DNV20] and [ENV22, Theorem 6.5]. For $\tilde{c} \in \mathbb{R}^n$, we use $\mathrm{LP}(\tilde{c})$ to denote (LP) with cost vector $\tilde{c}$, and $\mathrm{OPT}(\tilde{c})$ as the optimal value of $\mathrm{LP}(\tilde{c})$.

Theorem 8.2.5. *Let $c, c' \in \mathbb{R}^n$ be two cost vectors, such that both $\mathrm{LP}(c)$ and $\mathrm{LP}(c')$ have finite optimum values. Let s' be a dual optimal solution to $\mathrm{LP}(c')$. If there exists an index $j \in [n]$ such that*

$$s'_j > (m+1)\kappa_{\mathbf{A}} \|c - c'\|_\infty ,$$

then $x^*_j = 0$ for every optimal solution x^* to $\mathrm{LP}(c)$.

Proof. We may assume that $c \neq c'$, as otherwise we are done by complementary slackness. Let x' be an optimal solution to $\mathrm{LP}(c')$. We have that $x'_j = 0$. For the purpose of contradiction, suppose that there exists an optimal solution x^* to $\mathrm{LP}(c)$ such that $x^*_j > 0$. Let $h^{(1)}, \dots, h^{(k)}$ be a conformal circuit decomposition of $x^* - x'$. Then, $h^{(t)}_j > 0$ for some $t \in [k]$, and therefore $\|h^{(j)}\|_1 \le (m+1)\|h^{(j)}\|_\infty \le (m+1)\kappa h^{(t)}_j$ Observe that for any $i \in [n]$ where $h^{(t)}_i < 0$, we have $s'_i = 0$ because $x'_i > x^*_i \ge 0$. Hence,

$$\begin{aligned}\left\langle c, h^{(t)} \right\rangle &= \left\langle c - c', h^{(t)} \right\rangle + \left\langle c', h^{(t)} \right\rangle \ge -\|c - c'\|_\infty \|h^{(t)}\|_1 + \left\langle s', h^{(t)} \right\rangle \\ &\ge -(m+1)\kappa_{\mathbf{A}} \|c - c'\|_\infty h^{(t)}_j + s'_j h^{(t)}_j > 0 .\end{aligned}$$

Since $x^* - h^{(t)}$ is feasible to LP(c), this contradicts the optimality of x^*. □

8.2.3 Estimating circuit imbalances

The circuit augmentation algorithms in Sections 8.5 and 8.6 explicitly use the circuit imbalance measure $\kappa_{\mathbf{A}}$. However, this is NP-hard to approximate within a factor $2^{O(n)}$, see [DHNV20; Tun99]. We circumvent this problem using a standard guessing procedure, see e.g. [DHNV20; VY96]. Instead of $\kappa_{\mathbf{A}}$, we use an estimate $\hat{\kappa}$, initialized as $\hat{\kappa} = n$. Running the algorithm with this estimate either finds the desired feasible or optimal solution (which we can verify), or fails. In case of failure, we conclude that $\hat{\kappa} < \kappa_{\mathbf{A}}$, and replace $\hat{\kappa}$ by $\hat{\kappa}^2$. Since the running time of the algorithms is linear in $\log(n + \hat{\kappa})$, the running time of all runs will be dominated by the last run, giving the desired bound. For simplicity, the algorithm descriptions use the explicit value $\kappa_{\mathbf{A}}$. Throughout, we use the shorthand $\kappa = \kappa_{\mathbf{A}}$ whenever A is clear from the context.

8.3 The Circuit Diameter Bound

In this section, we show Theorem 8.1.4, namely, a bound $O(m^2 \log(m + \kappa_A))$ on the circuit diameter of a polyhedron in standard form (P). As outlined in the Introduction, let $B \subseteq [n]$ be a feasible basis and $N = [n] \setminus B$ such that $x^* = (\mathbf{A}_B^{-1} b, \mathbf{0}_N)$ is a basic solution to (LP). We can assume $n \leq 2m$: the union of the supports of the starting vertex $x^{(0)}$ and the target vertex x^* is at most $2m$; we can fix all other variables to 0. The simple 'shoot towards the optimum' procedure is shown in Algorithm 8.1.

Algorithm 8.1: Diameter Bound

Input : A polytope P as in (P) a vector $x^{(0)} \in P$ and a vertex $x^* \in P$.
Output : A circuit walk from $x^{(0)}$ to x^*.

1 Start from $t = 0$ and $x^{(0)}$.
2 At each iteration t, let $h^{(1)}, \dots, h^{(k)}$ be a conformal circuit decomposition of $x^* - x^{(t)}$
3 Let $g^{(t)}$ be the circuit in the decomposition that maximizes $\|h_N^{(i)}\|_1$ for $i \in [k]$ and update $x^{(t+1)} = \mathrm{aug}_P(x^{(t)}, g^{(t)})$.
4 Terminate once $x^{(t+1)} = x^*$.

A priori, even finite termination is not clear. The first key lemma shows that $\|x_N^{(t)}\|_1$ decreases geometrically, and bounds the relative error to x^*.

Lemma 8.3.1. *For every iteration $t \geq 0$ in Algorithm 8.1, we have $\|x_N^{(t+1)}\|_1 \leq (1 - \frac{1}{n})\|x_N^{(t)}\|_1$ and for all $i \in [n]$ we have $|x_i^{(t+1)} - x_i^{(t)}| \leq n|x_i^* - x_i^{(t)}|$.*

Proof. Let $h^{(1)}, \dots, h^{(k)}$ with $k \leq n$ be the conformal circuit decomposition of $x^* - x^{(t)}$ used in Algorithm 8.1. Note that $h_N^{(i)} \leq \mathbf{0}_N$ for $i \in [k]$ as $x_N^* = \mathbf{0}_N$ and $x^{(t)} \geq \mathbf{0}$. Then

$$\begin{aligned} \|g_N^{(t)}\|_1 = \max_{i \in [k]} \|h_N^{(i)}\|_1 \geq \frac{1}{k} \sum_{i \in [k]} \|h_N^{(i)}\|_1 = \frac{1}{k}\|x_N^{(t)}\|_1, \quad \text{and so} \\ \|x_N^{(t+1)}\|_1 = \|\mathrm{aug}_P(x^{(t)}, g^{(t)})_N\|_1 \leq \|x_N^{(t)} + g_N^{(t)}\|_1 \leq \big(1 - \frac{1}{k}\big)\|x_N^{(t)}\|_1. \end{aligned} \tag{8.5}$$

Let $\alpha^{(t)}$ be such that $x^{(t+1)} = x^{(t)} + \alpha^{(t)} g^{(t)}$. Then, by conformity and (8.5)

$$\alpha^{(t)} = \frac{\|x_N^{(t+1)} - x_N^{(t)}\|_1}{\|g_N^{(t)}\|_1} \leq \frac{\|x_N^{(t)}\|_1}{\|g_N^{(t)}\|_1} \leq k,$$

and so for all i we have $|x_i^{(t+1)} - x_i^{(t)}| = \alpha^{(t)}|g_i^{(t)}| \leq k|g_i^{(t)}| \leq k|x_i^* - x_i^{(t)}|$. □

We analyze the sets

$$L_t = \left\{ i \in [n] : x_i^* > n\kappa_A \|x_N^{(t)}\|_1 \right\}, \quad T_t = [n] \setminus L_t, \quad R_t = \left\{ i \in [n] : x_i^{(t)} \leq n x_i^* \right\}. \tag{8.6}$$

Lemma 8.3.2. *For every iteration $t \geq 0$, we have $L_t \subseteq L_{t+1} \subseteq B$ and $R_t \subseteq R_{t+1}$.*

Proof. We have that $L_t \subseteq L_{t+1}$ as $\|x_N^{(t)}\|_1$ is monotonically decreasing by Lemma 8.3.1, and $L_t \subseteq B$ as $x_N^* = 0_N$. For $j \in R_t$ note that if $x_j^{(t)} \geq x_j^*$, then $x_j^{(t+1)} \leq x_j^{(t)}$ by conformity. Otherwise, by Lemma 8.3.1 we have $x^{(t+1)} \leq x_i^{(t)} + n|x_i^* - x_i^{(t)}| \leq n x_i^*$. In both cases we conclude $j \in R_{t+1}$. □

Lemma 8.3.3. *If $\|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty > 2mn^2\kappa^2 \left\|x_{T_t}^*\right\|_\infty$, then $R_t \subsetneq R_{t+1}$.*

Proof. Let $i \in \operatorname{supp}(x^{(t)}) \setminus \operatorname{supp}(x^{(t+1)})$; such a variable exists by the maximality of the augmentation. Lemma 8.2.2 for $x^{(t+1)} - x^* \in \ker(A)$ implies that

$$\|x^{(t+1)} - x^*\|_\infty \leq \kappa \|x_N^{(t+1)} - x_N^*\|_1 = \kappa \|x_N^{(t+1)}\|_1 < \kappa \|x_N^{(t)}\|_1, \tag{8.7}$$

and so $i \notin L_t$. Noting that $x^{(t+1)} - x^{(t)}$ is a circuit and $x_i^{(t+1)} = 0$, it follows that

$$\left\|x_N^{(t)} - x_N^{(t+1)}\right\|_1 \leq (m\kappa + 1)x_i^{(t)} \leq 2m\kappa x_i^{(t)}. \tag{8.8}$$

On the other hand, let $h^{(1)}, \ldots, h^{(k)}$ be the conformal circuit decomposition of $x^* - x^{(t)}$ used in iteration t in Algorithm 8.1. Let $j \in T_t$ such that $|x_j^{(t)} - x_j^*| = \|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty$. There exists $\tilde{h}$ in this decomposition such that $|\widetilde{h}_j| \geq \frac{1}{n}|x_j^{(t)} - x_j^*|$. As $B \subseteq [n]$ is independent one has $\operatorname{supp}(\widetilde{h}) \cap N \neq \emptyset$ and so

$$\|\widetilde{h}_N\|_1 \geq \frac{|\widetilde{h}_j|}{\kappa} \geq \frac{|x_j^{(t)} - x_j^*|}{n\kappa}. \tag{8.9}$$

From (8.8), (8.9) and noting that $\|\widetilde{h}_N\|_1 \leq \|g_N^{(t)}\|_1 \leq \|x_N^{(t)} - x_N^{(t+1)}\|_1$ we get

$$x_i^{(t)} \geq \frac{\|x_N^{(t)} - x_N^{(t+1)}\|_1}{2m\kappa} \geq \frac{\|\widetilde{h}_N\|_1}{2m\kappa} \geq \frac{\|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty}{2mn\kappa^2}. \tag{8.10}$$

In particular, if as in the assumption of the lemma $\|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty > 2mn^2\kappa^2\|x_{T_t}^*\|_\infty$, then $x_i^{(t)} > n\|x_{T_t}^*\|_\infty \geq n x_i^*$. We conclude that $i \notin R_t$ and $i \in R_{t+1}$ as $x_i^{(t+1)} = 0$. □

We are ready to give the convergence bound.

Proof of Theorem 8.1.4. In light of Lemma 8.3.2, it suffices to show that either L_t or R_t is extended in every $O(n \log(n + \kappa))$ iterations; recall the assumption $n \leq 2m$. By Lemma 8.3.3, if $\|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty > 2mn^2\kappa^2 \left\|x_{T_t}^*\right\|_\infty$, then $R_t \subsetneq R_{t+1}$ is extended.

Otherwise, $\|x_{T_t}^{(t)} - x_{T_t}^*\|_\infty \leq 2mn^2\kappa^2 \left\|x_{T_t}^*\right\|_\infty$. Assuming $\|x_N^{(t)}\|_1 > 0$, by Lemma 8.3.1, there is an iteration $r = t + O(n \log(n + \kappa))$ such that $n^2\kappa(2mn^2\kappa^2 + 1)\|x_N^{(r)}\|_1 < \|x_N^{(t)}\|_1$. In particular,

$$(2mn^2\kappa^2 + 1)\|x_{T_t}^*\|_\infty \geq \|x_{T_t}^{(t)}\|_\infty \geq \|x_N^{(t)}\|_\infty \geq \frac{1}{n}\|x_N^{(t)}\|_1 > n\kappa(2mn^2\kappa^2 + 1)\|x_N^{(r)}\|_1. \tag{8.11}$$

Therefore $\|x_{T_t}^*\|_\infty > n\kappa\|x_N^{(r)}\|_1$ and so $L_t \subsetneq L_r$. □

8.4 Diameter Bounds for the Capacitated Case

In this section we consider diameter bounds for systems of the form

$$P_u = \{ x \in \mathbb{R}^n : \mathbf{A}x = b, \mathbf{0} \le x \le u \} \tag{Bound-P}$$

The theory in Section 8.3 carries over to P_u at the cost of turning m into n via the standard reformulation

$$\widetilde{P}_u = \left\{ (x,y) \in \mathbb{R}^{n+n} : \begin{bmatrix} \mathbf{A} & \mathbf{0} \\ \mathbf{I} & \mathbf{I} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} b \\ u \end{bmatrix}, x, y \ge \mathbf{0} \right\}, \quad P_u = \left\{ x : (x,y) \in \widetilde{P}_u \right\}. \tag{8.12}$$

Corollary 8.4.1. *The circuit diameter of a system in the form* (P) *with constraint matrix* $\mathbf{A} \in \mathbb{R}^{m \times n}$ *is* $O(n^2 \log(n + \kappa_{\mathbf{A}}))$.

Proof. Follows straightforward from Theorem 8.1.4 together with the reformulation (8.12). It is easy to check that κ of the constraint matrix of (8.12) coincides with $\kappa_{\mathbf{A}}$. □

Intuitively, the polytope should not become more complex; related theory in [Bra+21] also shows how both-sided bounds can be incorporated in a linear program without significantly changing the complexity of solving the program.

We prove Theorem 8.1.5 via the following new procedure. A basic feasible point $x^* \in P_u$ is characterized by a partition $B \cup L \cup H = [n]$ where A_B is a basis (has full column rank), $x^*_L = \mathbf{0}_L$ and $x^*_H = u_H$. In $O(n \log n)$ iterations, we fix all but $2m$ variables to the same bound as in x^*; for the remaining system with $2m$ variables, we can use the standard reformulation.

Algorithm 8.2: Capacitated Diameter Bound

Input : A polytope P as in (Bound-P) a vector $x^{(0)} \in P$ and a vertex $x^* \in P$.

Let $B \cup L \cup H = [n]$ be the partition for x^*, i.e. $\mathbf{0}_B \le A_B^{-1} b \le u_B$, $x^*_L = \mathbf{0}_L$ and $x^*_H = u_H$.
Set the cost $c \in \mathbb{R}^n_+$ as $c_i = 0$ if $i \in B$, $c_i = 1/u_i$ if $i \in L$, and $c_i = -1/u_i$ if $i \in H$.
Start from $t = 0$ and some $x^{(0)} \in P_u$.
do
  if $\langle c, x^{(t)} \rangle \ge -|H| + 1$ **then**
    Let $h^{(1)}, \dots, h^{(k)}$ be a conformal circuit decomposition of $x^* - x^{(t)}$. Let $g^{(t)} \in \arg\min_{i \in [k]} \langle c, h^{(i)} \rangle$.
  else
    Let $g^{(t)}$ be the circuit returned by Support-Circuit$(\mathbf{A}_{S_t}, c_{S_t}, x^{(t)}_{S_t}, S_t)$, where
    $S_t := \left\{ i \in L \cup H : x_i^{(t)} \ne x_i^* \right\}$.
  $x^{(t+1)} \leftarrow \mathrm{aug}_P(x^{(t)}, g^{(t)})$.
  $t \leftarrow t + 1$
  $S_t \leftarrow \left\{ i \in L \cup H : x_i^{(t)} \ne x_i^* \right\}$
while $|S_t| > m$
Run Diameter Bound on $\widetilde{\mathbf{A}} := \begin{bmatrix} \mathbf{A}_{B \cup S_t} & 0 \\ \mathbf{I} & \mathbf{I} \end{bmatrix}$ and $\widetilde{b} = \begin{bmatrix} b \\ u \end{bmatrix}$.

Proof of Theorem 8.1.5. We show that Algorithm 8.2 has the claimed number of iterations. First, note that $\langle c, x^* \rangle = -|H|$ is the optimum value. Initially, $\left\langle c, x^{(0)} \right\rangle = -\sum_{i \in H} \frac{x_i^{(0)}}{u_i} + \sum_{i \in L} \frac{x_i^{(0)}}{u_i} \le n$. Similar to Lemma 8.3.1, due to our choice of $g^{(t)}$ from the conformal circuit decomposition, we have

$\langle c, x^{(t+1)}\rangle + |H| \le (1-\frac{1}{n})(\langle c, x^{(t)}\rangle + |H|)$. In particular, $O(n \log n)$ iterations suffice to find an iterate t such that $\langle c, x^{(t)}\rangle < -|H| + 1$.

Note that the calls to Support-Circuit do not increase $\langle c, x^{(t)}\rangle$, so from now we will never make use of the conformal circuit decomposition again. A call to Support-Circuit will set at least one variable $i \in \operatorname{supp}(g^{(t)})$ to either 0 or u_i. We claim that either $x_i^{(t+1)} = 0$ for some $i \in L$, or $x_i^{(t+1)} = u_i$ for some $i \in H$, that is, we set a variable to the 'right' boundary. To see this, note that if $x_i^{(t+1)}$ hits the wrong boundary, then the gap between $\langle c, x^{(t+1)}\rangle$ and $-|H|$ must be at least 1, a clear contradiction to $\langle c, x^{(t+1)}\rangle < -|H| + 1$.

Thus, after at most n calls to Support-Circuit, we get $|S_t| \le m$, at which point we call Algorithm 8.1 with $\le 2m$ variables, so the diameter bound of Theorem 8.1.4 applies. □

8.5 A Circuit-Augmentation Algorithm for Feasibility

In this section we prove Theorem 8.1.6: given a system (LP) with cost $c = (\mathbf{0}_{[n]\setminus N}, \mathbf{1}_N)$ for some $N \subseteq [n]$, find a solution x with $x_N = 0$, or show that no such solution exists.

As an application, assume we are looking for a feasible solution to the program (P). We can construct an auxiliary linear program, that has trivial feasible solutions and whose optimal solutions correspond to feasible solutions of the original program (P). This is very much in the same tune as Phase I simplex algorithms.

$$\min \langle \mathbf{1}_n, z\rangle \quad \text{s.t.} \quad \mathbf{A}y - \mathbf{A}z = b\,,\ y, z \ge \mathbf{0}\,. \tag{Aux-LP}$$

For the constraint matrix $\widetilde{\mathbf{A}} = \begin{bmatrix}\mathbf{A} & -\mathbf{A}\end{bmatrix}$ it is easy to see that $\kappa_{\widetilde{\mathbf{A}}} = \kappa_{\mathbf{A}}$ and that any solution $\mathbf{A}x = b$ can be converted into a feasible solution to (Aux-LP) via $(y, z) = (x^+, x^-)$. Hence, if the subroutines Support-Circuit and Ratio-Circuit are available for (Aux-LP), then we can find a feasible solution to (P) in $O(n^2 \log(n + \kappa_A))$ augmentation steps.

Our algorithm is presented in Algorithm 8.3. We maintain a set $\mathcal{L}_t \subseteq [n] \setminus N$, initialized as $\emptyset$. Whenever $x_i^{(t)} \ge 8n^3\kappa^2\|x_N^{(t)}\|_1$ for the current iterate $x^{(t)}$, we add i to $\mathcal{L}_t$. The key part of the analysis is to show that $\mathcal{L}_t$ is extended in every $O(n \log n)$ iterations.

We let $T_t = [n] \setminus \mathcal{L}_t$ denote the complement set. At each iteration when $\mathcal{L}_t$ is extended, we run a sequence of at most n Support-Circuit$(\mathbf{A}, c, x^{(t)}, T_t)$ steps. These are repeated as long as $\|x_{T_t}^{(t)}\|_\infty < 4n\kappa\|x_N^{(t)}\|_1$ and there are circuits in $\operatorname{supp}(x^{(t)})$ intersecting T_t. Afterwards, we run a sequence of Ratio-Circuit iterations until a new index is added to $\mathcal{L}_t$.

In the Support-Circuit iterations, we have an additional requirement that $g_k^{(t)} < 0$ for at least one $k \in T_t$ for the returned circuit. This is guaranteed whenever $\langle c, g^{(t)}\rangle < 0$. If the oracle would return a circuit with $\langle c, g^{(t)}\rangle = 0$ and $g_{T_t}^{(t)} \ge 0$, then we can replace $g^{(t)}$ by $-g^{(t)}$.

Proof of Theorem 8.1.6. Algorithm 8.3 performs at most n^2 Support-Circuit iterations; we show that $\mathcal{L}_t$ is extended after a sequence of $O(n \log(n + \kappa))$ Ratio-Circuit iterations; this implies the claim. Let us first analyze what happens at Ratio-Circuit iterations.

Claim 8.5.0.1. *If Ratio-Circuit is used in iteration t, then either $\|x_N^{(t+1)}\|_1 \le (1 - \frac{1}{n})\|x_N^{(t)}\|_1$, or the algorithm terminates with a dual certificate.*

Proof. The oracle returns $g^{(t)}$ that is optimal to (8.3) and $(y^{(t)}, s^{(t)})$ with optimum value $-\lambda$. Recall that we use weights $w_i = 1/x_i^{(t)}$. If $\langle b, y^{(t)}\rangle > 0$, the algorithm terminates. Otherwise, note that

$$\left\langle c, x^{(t)}\right\rangle = \left\langle b, y^{(t)}\right\rangle + \left\langle s^{(t)}, x^{(t)}\right\rangle \le \lambda\left\langle w, x^{(t)}\right\rangle = n\lambda\,,$$

Algorithm 8.3: Feasibility-Algorithm

Input : Linear program in standard form (LP) with cost $c = (\mathbf{0}_{[n]\setminus N}, \mathbf{1}_N)$ for some $N \subseteq [n]$, and initial feasible solution $x^{(0)}$.

Output : A solution x with $x_N = \mathbf{0}$, or a dual solution y with $\langle b, y\rangle > 0$.

$t \leftarrow 0$; $\mathcal{L}_{-1} \leftarrow 0$
while $x_N^{(t)} > 0$ **do**
  $\mathcal{L}_t \leftarrow \mathcal{L}_{t-1} \cup \{i \in [n] : x_i^{(t)} \geq 8n^3\kappa^2\|x_N^{(t)}\|_1\}$; $T_t \leftarrow [n] \setminus \mathcal{L}_t$
  if $t = 0$ *or* $\mathcal{L}_t \setminus \mathcal{L}_{t-1} \neq \emptyset$ **then**
    while $\|x_{T_t}^{(t)}\|_\infty < 4n\kappa\|x_N^{(t)}\|_1$ *and there is a circuit in* $\mathrm{supp}(x^{(t)})$ *intersecting* T **do**
      $g^{(t)} \leftarrow$ Support-Circuit$(\mathbf{A}, c, x^{(t)}, T_t)$ such that $g_k^{(t)} < 0$ for some $k \in T_t$
      $x^{(t+1)} \leftarrow \mathrm{aug}_P(x^{(t)}, g^{(t)})$; $t \leftarrow t+1$;
  $(g^{(t)}, y^{(t)}, s^{(t)}) \leftarrow$ Ratio-Circuit$(\mathbf{A}, c, 1/x^{(t)})$
  if $\left\langle b, y^{(t)}\right\rangle > 0$ **then**
    Terminate with infeasibility certificate
  $x^{(t+1)} \leftarrow \mathrm{aug}_P(x^{(t)}, g^{(t)})$; $t \leftarrow t+1$
return x

implying $\lambda \geq \langle c, x^{(t)}\rangle/n$, and therefore $\langle c, g^{(t)}\rangle = -\lambda \leq -\langle c, x^{(t)}\rangle/n$. This implies the claim, noting that

$$\|x_N^{(t+1)}\|_1 = \left\langle c, x^{(t+1)}\right\rangle \leq \left\langle c, x^{(t)}\right\rangle - \left\langle c, g^{(t)}\right\rangle \leq \left(1 - \frac{1}{n}\right)\|x_N^{(t)}\|_1 .$$

□

Further, assume that at a Ratio-Circuit iteration, for some index $j \in [n]$ we have $x_j^{(t)} \geq 4n\kappa\|x_N^{(t)}\|_1$. Then,

$$\begin{aligned}\frac{x_j^{(t+1)}}{\|x_N^{(t+1)}\|_1} &\geq \frac{x_j^{(t)} - \kappa\|x_N^{(t+1)} - x_N^{(t)}\|_1}{(1-\frac{1}{n})\|x_N^{(t)}\|_1} \geq \frac{x_j^{(t)} - 2\kappa\|x_N^{(t)}\|_1}{(1-\frac{1}{n})\|x_N^{(t)}\|_1} \\ &\geq \frac{x_j^{(t)} - x_j^{(t)}/(2n)}{(1-\frac{1}{n})\|x_N^{(t)}\|_1} = \left(1 + \frac{1}{2n-2}\right)\frac{x_j^{(t)}}{\|x_N^{(t)}\|_1} .\end{aligned} \tag{8.13}$$

Claim 8.5.0.2. *Throughout for every $j \in \mathcal{L}_t$, we have $x_j^{(t)} \geq 4n^3\kappa^2\|x_N^{(t)}\|_1$*

Proof. The proof is by induction on the number of iterations; it holds at the beginning. Assume that the property already holds at iteration t, and let us analyse what happens if Support-Circuit is called. This may happen only if $\|x_{T_t}^{(t)}\|_\infty < 4n\kappa\|x_N^{(t)}\|_1$. By the condition that $g_k^{(t)} < 0$ for some $k \in T_t$, and by the definition of κ, we have $\|g\|_\infty \leq \kappa|g_k^{(t)}|$. Together with $x_k^{(t)} \leq 4n\kappa\|x_N^{(t)}\|_1$ and the induction hypothesis, this guarantees that the augmentation must set a coordinate in T_t to 0, and consequently, $\|x^{(t+1)} - x^{(t)}\|_\infty \leq 4n\kappa^2\|x_N^{(t)}\|_1$.

To verify the claim for iteration $t+1$, consider any index $j \in \mathcal{L}_t$, and let $r \leq t$ be the iteration when j was added to $\mathcal{L}_r$; the claim holds at iteration r. We analyse the ratio $x_j^{(t')}/\|x_N^{(t')}\|_1$ for iterations $t' = r, \ldots, t+1$. At every iteration that performs Ratio-Circuit, if $x_j^{(t')}/\|x_N^{(t')}\|_1 \geq 4n^3\kappa^2$ then the ratio may only increase according to (8.13).

There are at most n sequences of at most n Support-Circuit augmentations throughout the algorithm. By the above argument, each augmentation may decrease $x_j^{(t')}$ by at most $4n\kappa^2\|x_N^{(t')}\|_1$; hence, the total decrease in $x_j^{(t')}/\|x_N^{(t')}\|_1$ throughout the algorithm is bounded by $4n^3\kappa^2$. Since the starting value was $\geq 8n^3\kappa^2$, it follows that the ratio may never drop below $4n^3\kappa^2$. □

After iteration $t = 0$ or in a later iteration after $\mathcal{L}_t$ got extended, we perform a (possibly empty) sequence of support circuit cancellations. Let us consider an iteration t right after we are done with the support circuit cancellations. We show that $\mathcal{L}_t$ is extended within $O(n \log(n + \kappa))$ consecutive calls to Ratio-Circuit; this completes the proof.

First, assume that the sequence of support circuit cancellations finished with $\|x^{(t)}_{T_t}\|_\infty > 4n\kappa\|x^{(t)}_N\|_1$, and let $j \in T_t$ such that $\|x^{(t)}_{T_t}\|_\infty = x^{(t)}_j$. By (8.13), we see that at iteration $r = t + O(n \log(n + \kappa))$ we must have $x^{(r)}_j / \|x^{(r)}_N\|_1 > 8n^3\kappa^2$, that is, j is added into $\mathcal{L}_r$.

Next, assume that after the sequence of support circuit cancellations $\operatorname{supp}(x^{(t)}) \cap T_t$ became linearly independent. Within $O(n \log(n + \kappa))$ consecutive Ratio-Circuit augmentations we reach an iterate $r = t + O(n \log(n + \kappa))$ such that $\|x^{(r)}_N\|_1 \leq (16n^5\kappa^3)^{-1}\|x^{(t)}_N\|_1$. Then, by Lemma 8.2.3,

$$\|x^{(r)}_{T_t}\|_1 \geq \frac{\|x^{(t)}_{T_t}\|_1}{n\kappa + 1} \geq \frac{\|x^{(t)}_N\|_1}{n\kappa + 1} > 8n^4\kappa^2\|x^{(r)}_N\|_1 ,$$

showing that some $j \in T_t$ must be included in $\mathcal{L}_r$. □

8.6 A Circuit-Augmentation Algorithm for Optimization

In this section, we give a circuit-augmentation algorithm for solving (LP), assuming an initial feasible solution $x^{(0)}$ is provided. At all times, the algorithm maintains a feasible primal solution $x^{(t)}$ to (LP), initialized with $x^{(0)}$. The goal is to augment $x^{(t)}$ using the subroutines Support-Circuit and Ratio-Circuit until the emergence of a set $\emptyset \neq N \subseteq [n]$ which satisfies $x^{(t)}_N = x^*_N = \mathbf{0}$ for every optimal solution x^* to (LP). When this happens, we have reached a lower dimensional face of the feasible region (P) that contains the optimal face. Hence, we can fix $x^{(t')}_N = \mathbf{0}$ in all subsequent iterations $t' \geq t$. In particular, the same procedure is repeated on a smaller LP with constraint matrix $\mathbf{A}_{[n]\setminus N}$, RHS vector b, and costs $c_{[n]\setminus N}$, initialized with the feasible solution $x^{(t)}_{[n]\setminus N}$. Since $\mathcal{E}(\mathbf{A}_{[n]\setminus N}) \subseteq \pi_{[n]\setminus N}(\mathcal{E}(\mathbf{A}))$, a circuit walk of the smaller LP is also a circuit walk of the original LP. This gives the overall circuit-augmentation algorithm.

In what follows, we focus on the aforementioned procedure (Algorithm 8.4), since the main algorithm just calls it at most n times. An instance of (LP) is given by $\mathbf{A} \in \mathbb{R}^{m\times n}$, $b \in \mathbb{R}^m$ and $c \in \mathbb{R}^n$. We fix parameters

$$\delta < \frac{1}{n^{3/2}(m+2)\kappa} , \qquad \Gamma = \frac{9\sqrt{n}\kappa^2 T}{\delta}, \qquad T = \left\lceil 4n \log\left(\frac{4mn\kappa}{\delta}\right) + 1 \right\rceil .$$

Throughout the procedure, $\mathbf{A}$ and b will be fixed, but we will sometimes modify the cost function c. Recall that for any $\tilde{c} \in \mathbb{R}^n$, we use LP($\tilde{c}$) to denote the problem with cost vector $\tilde{c}$, and the optimum value is OPT($\tilde{c}$).

At the start of Algorithm 8.4, we orthogonally project the original cost vector c to $\ker(\mathbf{A})$. This does not change the optimal face of (LP). If $c = 0$, then we terminate and return the current feasible solution $x^{(0)}$ as it is optimal. Otherwise, we scale the cost to $\|c\|_2 = 1$, and use Ratio-Circuit to obtain a basic dual feasible solution $s^{(-1)}$ to LP(c).

The majority of Algorithm 8.4 consists of repeated *phases*, ending when $\langle x^{(t)}, s^{(t-1)} \rangle = 0$. At the start of a phase, the set S of coordinates with large dual slack $s^{(t-1)}_i \geq \delta$ are identified. Based on this, a modified cost function $\tilde{c} \geq 0$ is derived from $s^{(t-1)}$ by truncating the entries not in S to zero. This modified cost $\tilde{c}$ will be used until the end of the phase. Next, we augment our current primal solution $x^{(t)}$ by calling Support-Circuit($\mathbf{A}, \tilde{c}, x^{(t)}, S$) to eliminate circuits in $\operatorname{supp}(x^{(t)})$ intersecting S

Algorithm 8.4: Variable-Fixing

Input : Linear program in standard form (LP), and initial feasible solution $x^{(0)}$.
Output : Either an optimal solution to (LP), or a feasible solution x with a nonempty set $N \subseteq [n]$ such that $x_N = x^*_N = \mathbf{0}$ for every optimal solution x^* to (LP).

$t \leftarrow 0$, $c \leftarrow \Pi_{\ker(\mathbf{A})}(c)$
if $c = 0$ **then**
  return $x^{(t)}$
$c \leftarrow c/\|c\|_2$
$(\cdot, s^{(t-1)}) \leftarrow$ Ratio-Circuit$(\mathbf{A}, c, \mathbf{1})$ ▷ Or any bfs to the dual of LP(c)
while $\left\langle x^{(t)}, s^{(t-1)} \right\rangle > 0$ **do** ▷ Start of a phase
  $S \leftarrow \left\{ i \in [n] : s_i^{(t-1)} \geq \delta \right\}$
  Set modified cost $\tilde{c} \in \mathbb{R}^n_+$ as $\tilde{c}_i \leftarrow s_i^{(t-1)}$ $\forall i \in S$, and $\tilde{c}_i \leftarrow 0$ $\forall i \notin S$
  while $\exists$ *a circuit in* supp$(x^{(t)})$ *intersecting* S *with nonpositive* $\tilde{c}$*-cost* **do**
    $g^{(t)} \leftarrow$ Support-Circuit$(\mathbf{A}, \tilde{c}, x^{(t)}, S)$ ▷ Iteration t
    $x^{(t+1)} \leftarrow \mathrm{aug}_P(x^{(t)}, g^{(t)})$, $s^{(t)} \leftarrow s^{(t-1)}$, $t \leftarrow t+1$
  for $i = 1$ **to** T **do**
    $(g^{(t)}, s^{(t)}) \leftarrow$ Ratio-Circuit$(\mathbf{A}, \tilde{c}, 1/x^{(t)})$ ▷ Iteration t
    $x^{(t+1)} \leftarrow \mathrm{aug}_P(x^{(t)}, g^{(t)})$, $t \leftarrow t+1$
    if $\left\langle \tilde{c}, g^{(t-1)} \right\rangle = 0$ **then**
      break
$N \leftarrow \left\{ i \in [n] : s_i^{(t-1)} > \kappa(m+1)n\delta \right\}$
return $(x^{(t)}, N)$

with nonpositive $\tilde{c}$-cost. Note that there are at most n such calls because each call sets a primal variable $x_i^{(t)}$ to zero.

In the remaining part of the phase, we augment $x^{(t)}$ using Ratio-Circuit$(\mathbf{A}, \tilde{c}, 1/x^{(t)})$ for T iterations. In every iteration, Ratio-Circuit$(\mathbf{A}, \tilde{c}, 1/x^{(t)})$ returns a minimum cost-to-weight ratio circuit $g^{(t)}$, where the choice of weights $1/x^{(t)}$ follows Wallacher [Wal89]. Recall that the oracle also gives a basic dual feasible solution $s^{(t)}$ to LP$(\tilde{c})$. If $g^{(t)}$ does not improve the current solution $x^{(t)}$, i.e., $\left\langle \tilde{c}, g^{(t)} \right\rangle = 0$, then we terminate the phase early as $x^{(t)}$ is already optimal to LP$(\tilde{c})$. In this case, $s^{(t)}$ is an optimal dual solution to LP$(\tilde{c})$ because $\left\langle x^{(t)}, s^{(t)} \right\rangle = 0$. This finishes the description of a phase.

Let $\tilde{c}^{(k)}$ denote the modified cost $\tilde{c}$ used in phase $k \geq 1$, with the convention $\tilde{c}^{(0)} = c$. Observe that $\tilde{c}^{(k)} \in \mathrm{im}(\mathbf{A}^\top) + c - r^{(k)}$ for some $r^{(k)} \in \mathbb{R}^n_+$ where $\|r^{(k)}\|_\infty \leq k\delta$. Thus, for any dual feasible solution s to LP$(\tilde{c}^{(k)})$, $s + r^{(k)}$ is dual feasible to LP(c). After the last phase k, we know that $(x^{(t)}, s^{(t-1)})$ is a pair of primal-dual optimal solutions to LP$(\tilde{c}^{(k)})$. As the analysis reveals, it turns out that there are at most n phases. Thus, the last step of the algorithm is to identify the coordinates with large dual slack $s_i^{(t-1)}$ relative to $n\delta$. Then, applying Theorem 8.2.5 for $c' = c - n\delta$ allows us to conclude that they can be fixed to zero.

To analyze the algorithm, let S_k denote the set S in phase $k \geq 1$. For every iteration $t \geq 0$, consider the sets

$$L_t := \{ i \in [n] : x_i^{(t)} > \Gamma \|x^{(t)}_{S_k}\|_1 \}, \quad M_t := [n] \setminus (S_k \cup L_t),$$

where k is the phase in which iteration t occured.

Lemma 8.6.1. *For $k < n$, let t and t' be the iteration in which* Ratio-Circuit *was first called in phase k and $k+1$ respectively. Then, $L_t \subsetneq L_{t'}$.*

Proof. Since the algorithm entered phase $k+1$, Ratio-Circuit was called T times during phase k. It follows that $s^{(t+T-1)} = s^{(t'-1)}$. First, we show that the duality gap decreases geometrically between iteration t and t':

$$\begin{aligned}
\left\langle \tilde{c}^{(k+1)}, x^{(t')} \right\rangle &\le \left\langle \tilde{c}^{(k+1)}, x^{(t+T)} \right\rangle && \text{(from Support-Circuit)}\\
&\le \left\langle s^{(t+T-1)}, x^{(t+T)} \right\rangle && \text{(because } \tilde{c}^{(k+1)} \le s^{(t'-1)}\text{)}\\
&\le \left\langle s^{(t+T-1)}, x^{(t+T-1)} \right\rangle && \text{(as } \left\langle s^{(t+T-1)}, g^{(t+T-1)} \right\rangle = \left\langle \tilde{c}^{(k)}, g^{(t+T-1)} \right\rangle \le 0\text{)}\\
&\le -n \left\langle \tilde{c}^{(k)}, g^{(t+T-1)} \right\rangle && \text{(due to the constraints in (8.4))}\\
&\le n \left(\left\langle \tilde{c}^{(k)}, x^{(t+T-1)} \right\rangle - \mathrm{OPT}(\tilde{c}^{(k)}) \right) && \text{(by step size } \alpha \ge 1 \text{ in Lemma 8.2.1)}\\
&\le \frac{n}{2^{(T-1)/n}} \left(\left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle - \mathrm{OPT}(\tilde{c}^{(k)}) \right) && \text{(by optimality gap decay in Lemma 8.2.1)}\\
&\le \frac{n}{2^{(T-1)/n}} \left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle.
\end{aligned}$$

Next, we claim that $\|\tilde{c}^{(k)}\|_\infty \le 4\sqrt{n}\kappa$. Since $\mathbf{0} \le \tilde{c}^{(k)} \le s^{(t-1)}$, it suffices to prove that $\|s^{(t-1)}\|_\infty \le 4\sqrt{n}\kappa$. We know that $s^{(t-1)}$ is a basic feasible solution to the dual of LP($\tilde{c}^{(k-1)}$). Moreover, by induction $\tilde{c}^{(k-1)} \in \mathrm{im}(A^\top) + c - r$ for some $\mathbf{0} \le r \le (k-1)\delta\mathbf{1}$. It follows that $s^{(t-1)} \in \mathrm{im}(\mathbf{A}^\top) + c - r$. Thus, by applying Corollary 8.2.4 to the dual LP, we see that

$$\|s^{(t-1)}\|_\infty \le (\kappa+1)\|c-r\|_1 \le (\kappa+1)(\|c\|_1 + \|r\|_1) \le (\kappa+1)\big(\sqrt{n} + n(k-1)\delta\big) \le 4\sqrt{n}\kappa$$

as desired. Combining this bound with the decay in duality gap yields a shrinkage between the 1-norms of $x^{(t)}_{S_k}$ and $x^{(t')}_{S_{k+1}}$:

$$\|x^{(t')}_{S_{k+1}}\|_1 \le \frac{1}{\delta}\left\langle \tilde{c}^{(k+1)}, x^{(t')} \right\rangle \le \frac{n}{2^{(T-1)/n}\delta}\left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle \le \frac{4n^{1.5}\kappa}{2^{(T-1)/n}\delta}\|x^{(t)}_{S_k}\|_1.$$

We are ready to prove that $L_t \subseteq L_{t'}$. Consider any iteration $t \le r < t+T$ in phase k. For every $i \in S_k$, we have $x^{(r)}_i \le \left\langle \tilde{c}^{(k)}, x^{(r)} \right\rangle/\delta \le \left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle/\delta$. Moreover, $\mathrm{supp}((g^{(r)})^-) \cap S_k \ne \emptyset$ because $\left\langle \tilde{c}^{(k)}, g^{(r)} \right\rangle < 0$. Hence, the total change in any coordinate $i \in [n]$ during the T calls to Ratio-Circuit in phase k is at most

$$|x^{(t+T)}_i - x^{(t)}_i| \le \frac{\kappa T}{\delta}\left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle \le \frac{4\sqrt{n}\kappa^2 T}{\delta}\|x^{(t)}_{S_k}\|_1.$$

By a similar argument, the total change in any coordinate $i \in [n]$ during the calls to Support-Circuit in phase $k+1$ is at most

$$|x^{(t')}_i - x^{(t+T)}_i| \le \frac{n\kappa}{\delta}\left\langle \tilde{c}^{(k+1)}, x^{(t+T)} \right\rangle \le \frac{n^2\kappa}{2^{(T-1)/n}\delta}\left\langle \tilde{c}^{(k)}, x^{(t)} \right\rangle \le \frac{4n^{2.5}\kappa^2}{2^{(T-1)/n}\delta}\|x^{(t)}_{S_k}\|_1 \le \|x^{(t)}_{S_k}\|_1.$$

Therefore, for every $i \in L_t$, we have

$$\begin{aligned} x_i^{(t')} \geq x_i^{(t)} - \left(\frac{4\sqrt{n}\kappa^2 T}{\delta} + 1\right)\|x_{S_k}^{(t)}\|_1 &\geq \left(\Gamma - \frac{4\sqrt{n}\kappa^2 T}{\delta} - 1\right)\|x_{S_k}^{(t)}\|_1 \\ &\geq \left(\Gamma - \frac{4\sqrt{n}\kappa^2 T}{\delta} - 1\right)\frac{2^{(T-1)/n}\delta}{4n^{1.5}\kappa}\|x_{S_{k+1}}^{(t')}\|_1 \\ &= \left(\frac{\delta}{4n^{1.5}\kappa}(\Gamma - 1) - \frac{\kappa T}{n}\right)2^{(T-1)/n}\|x_{S_{k+1}}^{(t')}\|_1 \\ &\geq \left(\frac{\kappa T}{n}\right)2^{(T-1)/n}\|x_{S_{k+1}}^{(t')}\|_1 \geq \Gamma\|x_{S_{k+1}}^{(t')}\|_1, \end{aligned}$$

where the last two inequalities follow from our choice of Γ and T.

To show strict containment $L_t \subsetneq L_{t'}$, it suffices to prove that $\|x_{S_k}^{(t)}\|_1 \leq mn\kappa\|x_{M_t \cup S_k}^{(t')}\|_\infty$. This would then imply the existence of a coordinate $i \in M_t \cup S_k$ such that

$$x_i^{(t')} \geq \frac{1}{mn\kappa}\|x_{S_k}^{(t)}\|_1 \geq \frac{2^{(T-1)/n}\delta}{4mn^{2.5}\kappa^2}\|x_{S_{k+1}}^{(t')}\|_1 \geq \Gamma\|x_{S_{k+1}}^{(t')}\|_1.$$

Let $h^{(1)}, h^{(2)}, \ldots, h^{(p)}$ be a conformal circuit decomposition of $x^{(t')} - x^{(t)}$. Consider the set $R := \{i \in [p] : \operatorname{supp}(h^{(i)}) \cap S_k \neq \emptyset\}$. For every $i \in R$, there exists a coordinate $j \in M_t \cup S_k$ such that $h_j^{(i)} > 0$. Otherwise, we have $h_{M_t \cup S_k}^{(i)} \leq 0$, so $\operatorname{supp}(h^{(i)}) \subseteq \operatorname{supp}(x^{(t)})$ and $\langle \tilde{c}^{(k)}, h^{(i)} \rangle \leq 0$. However, this is a contradiction as $h^{(i)}$ would have been found by Ratio-Circuit in phase k. Hence, we have

$$\|(x_{S_k}^{(t')} - x_{S_k}^{(t)})^-\|_1 = \sum_{i \in R}\|(h_{S_k}^{(i)})^-\|_1 \leq m\kappa \sum_{i \in R}\|(h_{M_t \cup S_k}^{(i)})^+\|_1 \leq m\kappa\|(x_{M_t \cup S_k}^{(t')} - x_{M_t \cup S_k}^{(t)})^+\|_1$$

which then gives us

$$\begin{aligned} \|x_{S_k}^{(t)}\|_1 &\leq \|x_{S_k}^{(t)} \wedge x_{S_k}^{(t')}\|_1 + \|(x_{S_k}^{(t')} - x_{S_k}^{(t)})^-\|_1 \\ &\leq \|x_{S_k}^{(t)} \wedge x_{S_k}^{(t')}\|_1 + m\kappa\|(x_{M_t \cup S_k}^{(t')} - x_{M_t \cup S_k}^{(t)})^+\|_1 \leq m\kappa\|x_{M_t \cup S_k}^{(t')}\|_1 \leq mn\kappa\|x_{M_t \cup S_k}^{(t')}\|_\infty. \end{aligned}$$

as desired. □

Before proving the main result of this section, we need the following lemma which guarantees the existence of a coordinate with large dual slack. It explains why we chose to work with a projected and normalized cost vector in Algorithm 8.4.

Lemma 8.6.2. *Let $c, c' \in \mathbb{R}^n$ be two cost vectors, and let s' be an optimal dual solution to $\mathrm{LP}(c')$. If $c \in \ker(\mathbf{A})$, $\|c\|_2 = 1$ and $\|c - c'\|_\infty < 1/(\sqrt{n}(m+2)\kappa)$ for some $\kappa \geq 1$, then there exists an index $j \in [n]$ such that*

$$s'_j > \frac{m+1}{\sqrt{n}(m+2)}.$$

Proof. Let $r = c - c'$. Note that $s' + r \in \operatorname{im}(\mathbf{A}^\top) + c$. Then,

$$\|s'\|_\infty + \|r\|_\infty \geq \|s' + r\|_\infty \geq \frac{1}{\sqrt{n}}\|s' + r\|_2 \geq \frac{1}{\sqrt{n}}\|c\|_2 = \frac{1}{\sqrt{n}},$$

where the last inequality is due to $s' + r - c$ and c being orthogonal. This gives us

$$\|s'\|_\infty \geq \frac{1}{\sqrt{n}} - \|r\|_\infty > \frac{(m+2)\kappa - 1}{\sqrt{n}(m+2)\kappa} \geq \frac{m+1}{\sqrt{n}(m+2)}$$

as desired because $\kappa \geq 1$. □

Proof of Theorem 8.1.7. We first prove correctness of Algorithm 8.4. By Lemma 8.6.1, there exists a phase $1 \leq k \leq n$ with its first Ratio-Circuit iteration t such that (i) $\operatorname{rk}(\mathbf{A}_{L_t}) = m$; or (ii) $\operatorname{rk}(\mathbf{A}_{L_t}) = m-1$ and $|L_t| = n-1$. We claim that $x^{(t)}$ is an optimal solution to LP($\tilde{c}^{(k)}$). Case (ii) is easy to see because no circuits intersect $S_k = [n] \setminus L_t$. For Case (i), suppose that there exists a coordinate $j \in \operatorname{supp}(x^{(t)}) \cap S_k$ for the sake of contradiction. Since $\mathbf{A}_{L_t}$ is full rank, we can pick any basis $B \subseteq L_t$ to form a fundamental circuit in $B \cup \{j\}$. However, the associated elementary vector g would have been found by Support-Circuit because $\operatorname{supp}(g) \subseteq \operatorname{supp}(x^{(t)})$ and $\left\langle \tilde{c}^{(k)}, g \right\rangle = \tilde{c}^{(k)}_j g_j \neq 0$. This is a contradiction.

Consider the output $(g^{(t)}, s^{(t)})$ of Ratio-Circuit in iteration t. Since $\left\langle \tilde{c}^{(k)}, g^{(t)} \right\rangle = 0$, we have $\left\langle x^{(t)}, s^{(t)} \right\rangle = 0$ from the constraints in (8.4). So, this is the last iteration of the algorithm. Furthermore, $s^{(t)}$ is an optimal dual solution to LP($\tilde{c}^{(k)}$). By induction on k, we know that $\tilde{c}^{(k)} \in \operatorname{im}(\mathbf{A}^\top) + c - r$ for some $\|r\|_\infty \leq n\delta$. Hence, $s^{(t)}$ is also an optimal dual solution to LP(c') where $c' := c - r$. Since $c \in \ker(\mathbf{A})$, $\|c\|_2 = 1$ and $\|c - c'\|_\infty \leq n\delta < 1/(\sqrt{n}(m+2)\kappa)$, by Lemma 8.6.2, there exists an index $j \in [n]$ such that $s^{(t)}_j > (m+1)\kappa n\delta \geq (m+1)\kappa\|c - c'\|_\infty$. Thus, Theorem 8.2.5 allows us to conclude that $x^{(t)} = x^*_j = 0$ for every optimal solution x^* to LP(c).

Runtime-wise, there are at most T calls to Ratio-Circuit per phase. On the other hand, there are at most n calls to Support-Circuit per phase because each call sets a primal variable to zero. Since the main circuit-augmentation algorithm consists of applying Algorithm 8.4 at most n times, we obtain the desired runtime. □

8.7 An improved bound for steepest-descent augmentation

We now prove Theorem 8.1.3. The proof follows the same lines as that of the Goldberg–Tarjan algorithm; see also [AMO93, Section 10.5] for the analysis. A factor $\log n$ improvement over the original bound was given in [RG94]. A key property in the original analysis is that for a flow around a cycle (i.e., an elementary vector), every edge carries at least $1/|V|$ fraction of the ℓ_1-norm of the flow. This can be naturally replaced by the argument that for every elementary flow g, the minimum nonzero value of $|g_i|$ is at least $\|g\|_1/(1 + (m-1)\kappa_{\mathbf{A}})$.

The Goldberg–Tarjan algorithm has been generalized to separable convex minimization with linear constraints by Karzanov and McCormick [KM97]. Instead of κ_W, they use the maximum entry in a Graver basis (see Section 3.6 below). Lemma 10.1 in their paper proves a weakly polynomial bound similar to Lemma 8.7.2 for the separable convex setting. However, no strongly polynomial analysis is given (which is in general not possible for the nonlinear setting).

Our arguments will be based on the dual of System 1.3:

$$\begin{aligned} \max\ & \langle y, b\rangle - \langle u, t\rangle \\ & \mathbf{A}^\top y + s - t = c \\ & s, t \geq \mathbf{0}\,. \end{aligned} \tag{8.14}$$

Recall the primal-dual slackness conditions from Section 3.5: if x is feasible to Capacitated-LP and $y \in \mathbb{R}^m$, they are primal and dual optimal solutions if and only if $\langle a_i, y\rangle \leq c_i$ if $x_i < u_i$ and $\langle a_i, y\rangle \geq c_i$ if $x_i > 0$.

Let us start by formulating the steepest-descent direction as an LP. Let

$$\bar{\mathbf{A}} = \begin{bmatrix} \mathbf{A} & -\mathbf{A} \end{bmatrix} \in \mathbb{R}^{m\times(2n)} \quad \text{and} \quad \bar{c} = \begin{bmatrix} c \\ -c \end{bmatrix} \in \mathbb{R}^{2n}. \tag{8.15}$$

It is easy to see that $\kappa_{\bar{\mathbf{A}}} = \kappa_{\mathbf{A}}$. For a feasible solution $x = x^{(t)}$ to Capacitated-LP, we define *residual variable set*

$$N(x) = \{i \in [n] : x_i < u_i\} \cup \{n + j : j \in [n] : x_j > 0\} \subseteq [2n],$$

and consider the system

$$\begin{aligned} \min \langle \bar{c}, z\rangle \\ \bar{\mathbf{A}} z &= \mathbf{0} \\ \langle \mathbf{1}_{2n}, z\rangle &= 1 \\ z_{[2n]\setminus N(x)} &= \mathbf{0} \\ z &\geq \mathbf{0}. \end{aligned} \tag{8.16}$$

We can map a solution $z \in \mathbb{R}^{2n}$ to $g \in \mathbb{R}^n$ by setting $g_i = z_i - z_{n+i}$. We will assume that z is chosen as a basic optimal solution. Observe that every basic feasible solution to this program maps to an elementary vector in $\ker(\bar{\mathbf{A}})$. The dual program can be equivalently written as

$$\begin{aligned} \min \varepsilon \\ \langle \bar{a}_i, y\rangle \leq \bar{c}_i + \varepsilon \quad \forall i \in N(x). \end{aligned} \tag{8.17}$$

For the solution x, we let $\varepsilon(x)$ denote the optimal solution to this dual problem; thus, the optimal solution to the primal is $-\varepsilon(x)$. If $\varepsilon = 0$, then x and y are complementary primal and dual optimal solutions to Capacitated-LP. We first show that this quantity is monotone (a key step also in the analysis in [DHL15]).

Lemma 8.7.1. *At every iteration of the circuit augmentation algorithm, $\varepsilon(x^{(t+1)}) \leq \varepsilon(x^{(t)})$.*

Proof. Let $\varepsilon = \varepsilon(x^{(t)})$ and let y be an optimal solution to (8.17) for $N(x^{(t)})$. We show that the same y is also feasible for $N(x^{(t+1)})$; the claim follows immediately. There is nothing to prove if $N(x^{(t+1)}) \subseteq N(x^{(t)})$, so let $i \in N(x^{(t+1)}) \setminus N(x^{(t)})$.

Assume first $i \in [n+1, 2n]$; let $i = n + j$. This means that $x_j^{(t)} = 0 < x_j^{(t+1)}$; therefore, the augmenting direction g has $g_j > 0$. Thus, for the optimal solution z to (8.16), we must have $z_j > 0$. By primal-dual slackness, $\langle a_j, y\rangle = c_j + \varepsilon$; thus,

$$\langle \bar{a}_i, y\rangle = \langle -a_j, y\rangle = -c_j - \varepsilon < -c_j = \bar{c}_j.$$

The case $i \in [n]$ is analogous. □

The next lemma shows that within every n iterations, $\varepsilon(x^{(t)})$ decreases by a factor depending on $\kappa_{\mathbf{A}}$.

Lemma 8.7.2. *For every iteration t, $\varepsilon(x^{(t+n)}) \leq \left(1 - \frac{1}{1+(m-1)\kappa_{\mathbf{A}}}\right)\varepsilon(x^{(t)})$.*

Proof. Let us set $N = N(x^{(t)})$, $\varepsilon = \varepsilon(x^{(t)})$, and let $y = y^{(t)}$ be an optimal dual solution to (8.17) for $x^{(t)}$. Let

$$T := \{\, i \in N : \langle \bar{a}_i, y\rangle > \bar{c}_i \,\} \subseteq [2n];$$

that is, if $i \in [n]$ then $\langle a_i, y\rangle > c_i$, and if $i \in [n+1, 2n]$, $i = n + j$, then $\langle a_j, y\rangle < c_j$. In particular, $|T \cap \{i, i+n\}| \leq 1$ for every $i \in [n]$. Let $z^{(t)}$ be the basic optimal solution to (8.16) for $x^{(t)}$. By complementary slackness, every $i \in \operatorname{supp}(z^{(t)})$ must have $\langle \bar{a}_i, y\rangle = \bar{c}_i + \varepsilon$, and thus, $\operatorname{supp}(z^{(t)}) \subseteq T$.

Claim 8.7.2.1. *Let us pick $k > t$ as the first iteration when for the basic optimal solution $z^{(k)}$ to (8.16), we have $\operatorname{supp}(z^{(k)}) \setminus T \neq \emptyset$. Then $k \leq t + n$, and the solution (y, ε) is still feasible for (8.17) for $x^{(k)}$.*

Proof. For $r \in [t, k-1]$, let $T^{(r)} = T \cap N(x^{(r)})$. We show that $T^{(r+1)} \subsetneq T^{(r)}$. Since $|T| \leq n$, this implies $k \leq t + n$. Let $z^{(r)}$ be the basic optimal solution for (8.16); recall that the augmenting direction is computed with $g_j = z^{(r)}_j - z^{(r)}_{n+j}$. By the choice of k, $\operatorname{supp}(z^{(r)}) \subseteq T^{(r)}$. Thus, we may only increase x_i for $i \in T \cap [n]$ and decrease it for $i = j - n$ for $j \in T \cap [n+1, 2n]$. Consequently, every index i entering $N(x^{(r+1)})$ has $\langle \bar{a}_i, y\rangle < \bar{c}_i$, and therefore (y, ε) remains feasible throughout.

We now turn to the proof of $T^{(r+1)} \subsetneq T^{(r)}$. Since we use a maximal augmentation, at least one index leaves $T^{(r)}$ at each iteration. We claim that $T^{(r+1)} \setminus T^{(r)} = \emptyset$. For a contradiction, assume there exists $i \in T^{(r+1)} \setminus T^{(r)}$. If $i \in [n]$, then $i + n$ must be in the support of $z^{(r)}$; in particular, $i + n \in T^{(r)}$. But this would mean that $\{i, i+n\} \subseteq T$, in contradiction with the definition of T. Similarly, for $i \in [n+1, 2n]$. □

Let us now consider the optimal solution $z = z^{(k)}$ to (8.16) at iteration k; by the above claim, (y, ε) is still a feasible dual solution. Select an index $j \in \operatorname{supp}(z) \setminus T$.

$$\langle -\bar{c}, z\rangle = \left\langle \bar{\mathbf{A}}^\top y - \bar{c}, z\right\rangle \leq \varepsilon \sum_{i \in \operatorname{supp}(z)\setminus\{j\}} z_i = (1 - z_j)\varepsilon \leq \left(1 - \frac{1}{1 + (m-1)\kappa_{\mathbf{A}}}\right)\varepsilon\,.$$

In the first inequality, we use that $\langle \bar{a}_i, y\rangle - \bar{c}_i \leq \varepsilon$ by the feasibility of (y, ε), and $\left\langle \bar{a}_j, y\right\rangle - \bar{c}_j \leq 0$ by the choice of $j \notin T$. In the second equality, we use the constraint $\sum_i z_i = 1$. The final inequality uses that z is a basic solution, and therefore, an elementary vector in $\ker(\bar{\mathbf{A}})$. In particular $|\operatorname{supp}(z)| \leq m$, and $z_i \leq \kappa_{\bar{\mathbf{A}}} z_j = \kappa_{\mathbf{A}} z_j$. Consequently, $z_j \geq 1/(1 + (m-1)\kappa_{\mathbf{A}})$. □

We say that the variable $j \in [2n]$ is *frozen* at iteration t, if $j \notin N(x^{(t')})$ for any $t' \geq t$. Thus, for $j \in [n]$, $x_j = u_j$, and for $j \in [n+1, 2n]$, $j = i + n$, $x_i = 0$ for all subsequent iterations. We show that a new frozen variable can be found in every $O(nm\kappa_{\mathbf{A}} \log(m\kappa_{\mathbf{A}}))$ iterations; this implies Theorem 8.1.3.

Lemma 8.7.3. *For every iteration $t \geq 1$, there is a variable $j \in N(x^{(t)})$ that is frozen at iteration k for $k = t + O(nm\kappa_{\mathbf{A}} \log(\kappa_{\mathbf{A}} + n))$.*

Proof. Let $\varepsilon = \varepsilon(x^{(t)})$. By Lemma 8.7.2, we can choose $k = t + O(nm\kappa_{\mathbf{A}} \log(n + \kappa_{\mathbf{A}}))$ such that $\varepsilon' = \varepsilon(x^{(k)}) < \varepsilon/(2n(\kappa_{\mathbf{A}} + 1))$. Consider the primal and dual optimal solutions (z, y, ε) to (8.16) and (8.17) at iteration t and (z', y', ε') at iteration k.

Claim 8.7.3.1. *There exists a $j \in \operatorname{supp}(z)$ such that $\left\langle \bar{a}_j, y'\right\rangle > \bar{c}_j + 2n(\kappa_{\mathbf{A}} + 1)\varepsilon'$.*

Proof. For a contradiction, assume that $\left\langle \bar{a}_j, y'\right\rangle - \bar{c}_j \leq 2n(\kappa_{\mathbf{A}} + 1)\varepsilon'$ for every $j \in \operatorname{supp}(z)$. Then,

$$\varepsilon = \langle \bar{c}, z\rangle = \left\langle \bar{\mathbf{A}}^\top y - \bar{c}, z\right\rangle \leq 2n(\kappa_{\mathbf{A}} + 1)\varepsilon' \sum_j z_j = 2n(\kappa_{\mathbf{A}} + 1)\varepsilon'\,,$$

contradicting the choice of ε'. □

We now show that all such indices are frozen at iteration k by making use of Theorem 3.5.7 on proximity. Let $x' = x^{(k)}$ and $x'' = x^{(k'')}$ for any $k'' > k$; let (y'', ε'') be optimal to (8.17) at iteration k''; we have $\varepsilon'' \leq \varepsilon'$ by Lemma 8.7.1.

Let us define the cost $c' \in \mathbb{R}^n$ by

$$c'_i := \begin{cases} \langle \bar{a}_i, y' \rangle & \text{if } 0 < x'_i < u_i \\ \max\{c_i, \langle \bar{a}_i, y' \rangle\} & \text{if } x'_i = u_i \\ \min\{c_i, \langle \bar{a}_i, y' \rangle\} & \text{if } x'_i = 0\,. \end{cases}$$

If we replace the cost c by c', then x' and y' satisfy complementary slackness, and hence are optimal to Capacitated-LP and (8.14). Moreover, the optimality of (y', ε') to (8.17) guarantees that $\|c' - c\|_\infty \le \varepsilon'$.

We similarly construct c'' for y'', and note that x'' and y'' are primal and dual optimal solutions for the costs c'', $\|c'' - c\|_\infty \le \varepsilon''$. Further,

$$\|c' - c''\|_1 \le n\|c' - c''\|_\infty \le n(\|c' - c\|_\infty + \|c'' - c\|_\infty) \le n(\varepsilon' + \varepsilon'') \le 2n\varepsilon'$$

We thus apply Theorem 3.5.7 for (x', y') for c' and (x'', y'') for c'', showing that every variable j as in Claim 8.7.3.1 must be frozen. □

9 Fractional Linear Programming

We present an accelerated, or *look-ahead* version of the Newton–Dinkelbach method, a well-known technique for solving fractional and parametric optimization problems. This acceleration halves the Bregman divergence between the current iterate and the optimal solution within every two iterations. Using the Bregman divergence as a potential in conjunction with combinatorial arguments, we obtain strongly polynomial algorithms in three applications domains: (i) For linear fractional combinatorial optimization, we show a convergence bound of $O(m \log m)$ iterations; the previous best bound was $O(m^2 \log m)$ by Wang et al. (2006). (ii) We obtain a strongly polynomial label-correcting algorithm for solving linear feasibility systems with Two Variables Per Inequality (2VPI). For a 2VPI system with n variables and m constraints, our algorithm runs in $O(mn)$ iterations. Every iteration takes $O(mn)$ time for general 2VPI systems, and $O(m + n \log n)$ time for the special case of Deterministic Markov Decision Processes (DMDP). This extends and strengthens a previous result by Madani (2002) that showed a weakly polynomial bound for a variant of the Newton–Dinkelbach method for solving DMDPs. (iii) We give a simplified variant of the parametric submodular function minimization result by Goemans et al. (2017). We only present (i) in full detail, see [DKNV21] for the full proofs of (ii) and (iii).

This chapter is based on joint work with Daniel Dadush, Zhuan Khye Koh, and László A. Végh [DKNV21]. Section 9.3 contains novel unpublished material.

Contents

9.1 Introduction

Linear fractional optimization problems are well-studied in combinatorial optimization. Given a closed domain $\mathcal{D} \subseteq \mathbb{R}^m$ and $c, d \in \mathbb{R}^m$ such that $\langle d, x \rangle > 0$ for all $x \in \mathcal{D}$, the problem is

$$\inf \frac{\langle c, x \rangle}{\langle d, x \rangle} \quad \text{s.t. } x \in \mathcal{D}\,. \tag{9.1}$$

The domain $\mathcal{D}$ could be either a convex set or a discrete set $\mathcal{D} \subseteq \{0,1\}^m$. Classical examples include finding minimum cost-to-time ratio cycles and minimum ratio spanning trees. One can equivalently formulate (9.1) as a parametric search problem. Let

$$f(\delta) \coloneqq \inf\{\, (c - \delta d)^\top x : x \in \mathcal{D} \,\}, \tag{9.2}$$

be a concave and decreasing function. Assuming (9.1) has a finite optimum δ, it corresponds to the unique root $f(\delta) = 0$.

A natural question is to investigate how the computational complexity of solving the minimum ratio problem (9.1) may depend on the complexity of the corresponding linear optimization problem $\min c^\top x$ s.t. $x \in \mathcal{D}$. Using the reformulation (9.2), one can reduce the fractional problem to the linear problem via binary search; however, the number of iterations needed to find an exact solution may depend on the bit complexity of the input. A particularly interesting question is: assuming there exists a strongly polynomial algorithm for linear optimization over a domain $\mathcal{D}$, can we find a strongly polynomial algorithm for linear fractional optimization over the same domain?

A seminal paper by Megiddo [Meg79] introduced the *parametric search* technique to solve linear fractional combinatorial optimization problems. He showed that if the linear optimization algorithm only uses $p(m)$ comparisons and $q(m)$ additions, then there exists an $O(p(m)(p(m)+q(m))$ algorithm for the linear fractional optimization problem. This in particular yielded the first strongly polynomial algorithm for the minimum cost-to-time ratio cycle problem. On a very high level, parametric search works by simulating the linear optimization algorithm for the parametric problem (9.2), with the parameter $\delta \in \mathbb{R}$ being indeterminate.

A natural alternative approach is to solve (9.2) using a standard root finding algorithm. Radzik [Rad92] showed that for a discrete domain $\mathcal{D} \subseteq \{0,1\}^m$, the *discrete Newton* method—in this context, also known as Dinkelbach's method [Din67]—terminates in a strongly polynomial number of iterations. In contrast to parametric search, there are no restrictions on the possible operations in the linear optimization algorithm. In certain settings, such as the maximum ratio cut problem, the discrete Newton method outperforms parametric search; we refer to the comprehensive survey by Radzik [Rad98] for details and comparison of the two methods.

9.1.1 Our Contributions

We introduce a new, accelerated variant of Newton's method for univariate functions. Let $f : \mathbb{R} \to \mathbb{R} \cup \{-\infty\}$ be a concave function. Under some mild assumptions on f, our goal is to either find the largest root, or show that no root exists. Let δ^* denote the largest root, or in case $f < 0$, let δ^* denote the largest maximizer of f. For simplicity, we now describe the method for differentiable functions. This will not hold in general: functions of the form (9.2) will be piecewise linear if $\mathcal{D}$ is finite or polyhedral. The algorithm description in Section 9.2 uses a form with supergradients (that can be choosen arbitrarily between the left and right derivatives).

The standard Newton method, also used by Radzik, proceeds through iterates $\delta^{(1)} > \delta^{(2)} > \ldots > \delta^{(t)}$ such that $f(\delta^{(i)}) \leq 0$, and updates $\delta^{(i+1)} = \delta^{(i)} - f(\delta^{(i)})/f'(\delta^{(i)})$.

Our new variant uses a more aggressive look-ahead technique. At each iteration, we compute $\delta = \delta^{(i)} - f(\delta^{(i)})/f'(\delta^{(i)})$, and jump ahead to $\delta' = 2\delta - \delta^{(i)}$. In case $f(\delta') \leq 0$ and $f'(\delta') < 0$, we update $\delta^{(i+1)} = \delta'$; otherwise, we continue with the standard iterate δ.

This modification leads to an improved and at the same time simplified analysis based on the Bregman divergence $D_f(\delta^*, \delta^{(i)}) = f(\delta^{(i)}) + f'(\delta^{(i)})(\delta^* - \delta^{(i)}) - f(\delta^*)$. We show that this decreases by a factor of two between any two iterations.

A salient feature of the algorithm is that it handles both feasible and infeasible outcomes in a unified framework. In the context of linear fractional optimization, this means that the assumption $d^\top x > 0$ for all $x \in \mathcal{D}$ in (9.1) can be waived. Instead, $d^\top x > 0$ is now added as a feasibility constraint to (9.1). This generalization is important when we use the algorithm to solve two variables per inequality systems.

This general result leads to improvements and simplifications of a number of algorithms using the discrete Newton method.

- For *linear fractional combinatorial optimization*, namely the setting (9.1) with $\mathcal{D} \subseteq \{0,1\}^m$, we obtain an $O(m \log m)$ bound on the number of iterations, a factor m improvement over the previous best bound $O(m^2 \log m)$ by Wang et al. [WYZ06] from 2006. We remark that Radzik's first analysis [Rad92] yielded a bound of $O(m^4 \log^2 m)$ iterations, improved to $O(m^2 \log^2 m)$ in [Rad98].
- Goemans et al. [GGJ17] used the discrete Newton method to obtain a strongly polynomial algorithm for parametric submodular function minimization. We give a simple new variant of this result with the same asymptotic running time, using the accelerated algorithm.
- For 2VPI systems, we obtain a strongly polynomial label-correcting algorithm. This will be discussed in more detail next.

9.1.2 Two Variables per Inequality Systems

A major open question in the theory of linear programming (LP) is whether there exists a strongly polynomial algorithm for LP. This problem is one of Smale's eighteen mathematical challenges for the twenty-first century [Sma98]. An LP algorithm is *strongly polynomial* if it only uses elementary arithmetic operations $(+, -, \times, /)$ and comparisons, and the number of such operations is polynomially bounded in the number of variables and constraints. Furthermore, the algorithm needs to be in PSPACE, i.e., the numbers occurring in the computations must remain polynomially bounded in the input size.

The notion of a strongly polynomial algorithm was formally introduced by Megiddo [Meg83] in 1983 (using the term *'genuinely polynomial'*), where he gave the first such algorithm for 2VPI systems. These are feasibility LPs where every inequality contains at most two variables. More formally, let $\mathcal{M}_2(n, m)$ be the set of $n \times m$ matrices with at most two nonzero entries per column. A 2VPI system is of the form $\mathbf{A}^\top y \leq c$ for $\mathbf{A} \in \mathcal{M}_2(n, m)$ and $c \in \mathbb{R}^m$.

If we further require that every inequality has at most one positive and at most one negative coefficient, it is called a (Monotone Two Variables Per Inequality (M2VPI)) system. A simple and efficient reduction is known from 2VPI systems with n variables and m inequalities to M2VPI systems with $2n$ variables and $\leq 2m$ inequalities [ERW89; HMNT93].

Connection between 2VPI and parametric optimization An M2VPI system has a natural graphical interpretation: after normalization, we can assume every constraint is of the form $y_u - \gamma_e y_v \leq c_e$. Such a constraint naturally maps to an arc $e = (u, v)$ with *gain factor* $\gamma_e > 0$ and cost c_e. Based on Shostak's work [Sho81] that characterized feasibility in terms of this graph, Aspvall and Shiloach [AS80] gave the first weakly polynomial algorithm for M2VPI systems.

We say that a directed cycle C is *flow absorbing* if $\prod_{e \in C} \gamma_e < 1$ and *flow generating* if $\prod_{e \in C} \gamma_e > 1$. Every flow absorbing cycle C implies an upper bound for every variable y_u incident to C; similarly, flow generating cycles imply lower bounds. The crux of Aspvall and Shiloach's algorithm is to find the tightest upper and lower bounds for each variable y_u.

Finding these bounds corresponds to solving fractional optimization problems of the form (9.1), where $\mathcal{D} \subseteq \mathbb{R}^m$ describes 'generalized flows' around cycles. The paper [AS80] introduced the *Grapevine* algorithm—a natural modification of the Bellman-Ford algorithm—to decide whether the optimum ratio is smaller or larger than a fixed value δ. The optimum value can be found using binary search on the parameter.

Megiddo's strongly polynomial algorithm [Meg83] replaced the binary search framework in Aspvall and Shiloach's algorithm by extending the parametric search technique in [Meg79]. Subsequently, Cohen and Megiddo [CM94] devised faster strongly polynomial algorithms for the problem. The current fastest strongly polynomial algorithm is given by Hochbaum and Naor [HN94], an efficient Fourier–Motzkin elimination with running time of $O(mn^2 \log m)$.

2VPI via Newton's method Since Newton's method proved to be an efficient and viable alternative to parametric search, a natural question is to see whether it can solve the parametric problems occurring in 2VPI systems. Radzik's fractional combinatorial optimization results [Rad92; Rad98] are not directly applicable, since the domain $\mathcal{D}$ in this setting is a polyhedron and not a discrete set.[1] Madani [Mad02] used a variant of the Newton–Dinkelbach method as a tool to analyze the convergence of policy iteration on DMDP, a special class of M2VPI systems (discussed later in more detail). He obtained a weakly polynomial convergence bound; it remained open whether such an algorithm could be strongly polynomial.

Our 2VPI algorithm We introduce a new type of strongly polynomial 2VPI algorithm by combining the accelerated Newton–Dinkelbach method with a *variable fixing* analysis. Variable fixing was first introduced in the seminal work of Tardos [Tar85] on minimum-cost flows, and has been a central idea of strongly polynomial algorithms, see in particular [GT89; RG94] for cycle cancelling minimum-cost flow algorithms, and [OV20; Vég17] for maximum generalized flows, a dual to the 2VPI problem.

We show that for every iterate $\delta^{(i)}$, there is a constraint that has been 'actively used' at $\delta^{(i)}$ but will not be used ever again after a strongly polynomial number of iterations. The analysis combines the decay in Bregman divergence shown in the general accelerated Newton–Dinkelbach analysis with a combinatorial *subpath monotonicity* property.

Our overall algorithm can be seen as an extension of Madani's DMDP algorithm. In particular, we adapt his 'unfreezing' idea: the variables y_u are admitted to the system one-by-one, and the accelerated Newton–Dinkelbach method is used to find the best 'cycle bound' attainable at the newly admitted y_u in the graph induced by the current variable set. This returns a feasible solution or reports infeasibility within $O(m)$ iterations. As every iteration takes $O(mn)$ time, our overall algorithm terminates in $O(m^2n^2)$ time. For the special setting of DMDP, the runtime per iteration improves to $O(m + n \log n)$, giving a total runtime of $O(mn(m + n \log n))$.

Even though our running time bound is worse than the state-of-the-art 2VPI algorithm [HN94], it is of a very different nature from all previous 2VPI algorithms. In fact, our algorithm is a *label-correcting*, naturally fitting to the family of algorithms used in other combinatorial optimization problems with constraint matrices from $\mathcal{M}_2(n, m)$ such as maximum flow, shortest paths, minimum-cost flow, and generalized flow problems. We next elaborate on this connection.

Label-correcting algorithms An important special case of M2VPI systems corresponds to the shortest paths problem: given a directed graph $G = (V, E)$ with target node $t \in V$ and arc costs $c \in \mathbb{R}^E$, we associate constraints $y_u - y_v \leq c_e$ for every arc $e = (u, v) \in E$ and $y_t = 0$. If the system

[1] The problem could be alternatively formulated with $\mathcal{D} \subseteq \{0, 1\}^m$ but with nonlinear functions instead of $c^\top x$ and $d^\top x$.

is feasible and bounded, the pointwise maximal solution corresponds to the shortest path labels to t; an infeasible system contains a negative cost cycle. A generic label-correcting algorithm maintains distance labels y that are upper bounds on the shortest path distances to t. The labels are decreased according to violated constraints. Namely, if $y_u - y_v > c_e$, then decreasing y_u to $c_e + y_v$ gives a smaller valid distance label at u. We terminate with the shortest path labels once all constraints are satisfied. The Bellman–Ford algorithm for the shortest paths problem is a particular implementation of the generic label-correcting algorithm; we refer the reader to [AMO93, Chapter 5] for more details.

It is a natural question if label-correcting algorithms can be extended to general M2VPI systems, where constraints are of the form $y_u - \gamma_e y_v \leq c_e$ for a 'gain/loss factor' $\gamma_e > 0$ associated with each arc. A fundamental property of M2VPI systems is that, whenever bounded, a unique pointwise maximal solution exists, i.e. a feasible solution y^* such that $y \leq y^*$ for every feasible solution y. A label-correcting algorithm for such a setting can be naturally defined as follows. Let us assume that the problem is bounded. The algorithm should proceed via a decreasing sequence $y^{(0)} \geq y^{(1)} \geq \ldots \geq y^{(t)}$ of labels that are all valid upper bounds on any feasible solution y to the system. The algorithm either terminates with the unique pointwise maximal solution $y^{(t)} = y^*$, or finds an infeasibility certificate.

The basic label-correcting operation is the 'arc update', decreasing y_u to $\min\{y_u, c_e + \gamma_e y_v\}$ for some arc $e = (u, v) \in E$. Such updates suffice in the shortest path setting. However, in the general setting arc operations only may not lead to finite termination. Consider a system with only two variables, y_u and y_v, and two constraints, $y_u - y_v \leq 0$, and $y_v - \frac{1}{2} y_u \leq -1$. The alternating sequence of arc updates converges to $(y_u^*, y_v^*) = (-2, -2)$, but does not finitely terminate. In this example, we can 'detect' the cycle formed by the two arcs, that implies the bound $y_u - \frac{1}{2} y_u \leq -1$.

Shostak's [Sho81] result demonstrates that arc updates, together with such 'cycle updates' should be sufficient for finite termination. Our M2VPI algorithm amounts to the first strongly polynomial label-correcting algorithm for general M2VPI systems, using arc updates and cycle updates.

Deterministic Markov decision processes A well-studied special case of M2VPI systems in which $\gamma \leq \mathbf{1}$ is known as DMDP. A *policy* corresponds to selecting an outgoing arc from every node, and the objective is to find a policy that minimizes the total discounted cost over an infinite time horizon. The pointwise maximal solution of this system corresponds to the optimal values of a policy.

The standard policy iteration, value iteration, and simplex algorithms can be all interpreted as variants of the label-correcting framework.[2] Value iteration can be seen as a generalization of the Bellman–Ford algorithm to the DMDP setting. As our previous example shows, value iteration may not be finite. One could still consider as the termination criterion the point where value iteration 'reveals' the optimal policy, i.e. updates are only performed using constraints that are tight in the optimal solution. If each discount factor γ_e is at most γ' for some $\gamma' > 0$, then it is well-known that value iteration converges at the rate $1/(1 - \gamma')$. This is in fact true more generally, for nondeterministic MDPs [LDK95]. However, if the discount factors can be arbitrarily close to 1, then Feinberg and Huang [FH14] showed that value iteration cannot reveal the optimal policy in strongly polynomial time even for DMDPs. Post and Ye [PY15] proved that simplex with the most negative reduced cost pivoting rule is strongly polynomial for DMDPs; this was later improved by Hansen et al. [HKZ14]. These papers heavily rely on the assumption $\gamma \leq \mathbf{1}$, and does not seem to extend to general M2VPI systems.

[2] The value sequence may violate monotonicity in certain cases of value iteration.

Madani's previously mentioned work [Mad02] used a variant of the Newton–Dinkelbach method as a tool to analyze the convergence of policy iteration on deterministic MDPs, and derived a weakly polynomial runtime bound.

One of our main results in [DKNV21] is the following.

Theorem 9.1.1 ([DKNV21]). *There exists a label-correcting algorithm based on the accelerated Newton–Dinkelbach methods that solves the feasibility of M2VPI linear systems in $O(m^2n^2)$ time.*

9.1.3 Submodular functions

The accelerated Newton Method for Line Search in Submodular Base Polytopes Let V be a set with n-elements and define $2^V := \{ S : S \subseteq V \}$ to be the set of all subsets of V. A function $h : 2^V \to \mathbb{R}$ is *submodular* if

$$h(S) + h(T) \geq h(S \cap T) + h(S \cup T) \quad \forall S, T \subseteq V .$$

Given a non-negative submodular function $h : 2^V \to \mathbb{R}_+$ and a vector $a \in \mathbb{R}^V$ satisfying $\max_{i \in V} a_i > 0$, we examine the problem of computing

$$\delta^* := \max \left\{ \delta : \min_{S \subseteq V} h(S) - \delta a(S) \geq 0 \right\}, \tag{9.3}$$

where $a(S) := \sum_{i \in S} a_i$. As the input model, we assume access to an evaluation oracle for h, which allows us to query $h(S)$ for any set $S \subseteq V$. The above problem models the line-search problem inside a submodular polyhedron and has been studied in [GGJ17; Nag07; Top78].

To connect to the root finding problem studied in previous sections, for $\delta \in \mathbb{R}$, we define

$$f(\delta) := \min_{S \subseteq V} h_\delta(S) := \min_{S \subseteq V} h(S) - \delta a(S).$$

Since f is the minimum of 2^n affine functions, f is a piecewise linear concave function. Noting that f is continuous, problem (9.3) can be equivalently restated as that of computing the largest root of f, i.e., the largest $\delta^* \in \mathbb{R}$ such that $f(\delta^*) = 0$. The assumption that h is non-negative ensures that $f(0) \geq 0$, and the assumption that $\max_{i \in V} a_i > 0$ ensures that δ^* exists and $\delta^* \geq 0$ (see the initialization section below). Given the root finding representation, we may apply the Newton–Dinkelbach method on f to compute δ^*. This approach was taken by Goemans, Gupta and Jaillet [GGJ17], who were motivated to give a more efficient alternative to the parametric search based algorithm of Nagano [Nag07]. Their main result is as follows:

Theorem 9.1.2 ([GGJ17]). *The Newton-Dinkelbach method requires at most $n^2 + O(n \log^2 n)$ iterations to solve (9.3).*

We provide in [DKNV21] a simplified potential function based proof of the above theorem using the accelerated Newton–Dinkelbach method (Algorithm 9.1), where we will give a slightly weaker $2n^2 + 2n + 4$ bound on the iteration count. Our analysis uses the same combinatorial ring family analysis as in [GGJ17], but the Bregman divergence enables considerable simplifications.

Chapter organization In Section 9.2, we present an accelerated Newton's method for univariate concave functions, and apply it to linear fractional combinatorial optimization and linear fractional programming. Section 9.3 contains our general of our results to general LP with dependency on the circuit imbalances κ.

9.2 An Accelerated Newton–Dinkelbach Method

Let $f\colon \mathbb{R} \to \overline{\mathbb{R}}$ be a proper concave function such that $f(\delta) \leq 0$ and $\partial f(\delta) \cap \mathbb{R}_{<0} \neq \emptyset$ for some $\delta \in \operatorname{dom}(f)$. Given a suitable starting point, as well as value and supergradient oracles of f, the Newton–Dinkelbach method either computes the largest root of f or declares that it does not have a root. In this chapter, we make the mild assumption that f has a root or attains its maximum. Consequently, the point

$$\delta^* := \max(\{\, \delta : f(\delta) = 0 \,\} \cup \arg\max f(\delta))$$

is well-defined. It is the largest root of f if f has a root. Otherwise, it is the largest maximizer of f. Therefore, the Newton–Dinkelbach method returns δ^* if f has a root, and certifies that $f(\delta^*) < 0$ otherwise.

The algorithm takes as input an initial point $\delta^{(1)} \in \operatorname{dom}(f)$ and a supergradient $g^{(1)} \in \partial f(\delta^{(1)})$ such that $f(\delta^{(1)}) \leq 0$ and $g^{(1)} < 0$. At the start of every iteration $i \geq 1$, it maintains a point $\delta^{(i)} \in \operatorname{dom}(f)$ and a supergradient $g^{(i)} \in \partial f(\delta^{(i)})$ where $f(\delta^{(i)}) \leq 0$. If $f(\delta^{(i)}) = 0$, then it returns $\delta^{(i)}$ as the largest root of f. Otherwise, a new point $\delta := \delta^{(i)} - f(\delta^{(i)})/g^{(i)}$ is generated. Now, there are two scenarios in which the algorithm terminates and reports that f does not have a root: (1) $f(\delta) = -\infty$; (2) $f(\delta) < 0$ and $g \geq 0$ where $g \in \partial f(\delta)$ is the supergradient given by the oracle. If both scenarios do not apply, the next point and supergradient is set to $\delta^{(i+1)} := \delta$ and $g^{(i+1)} := g$ respectively. Then, a new iteration begins.

According to this update rule, observe that $g^{(i)} < 0$ except possibly in the final iteration when $f(\delta^{(i)}) = 0$. This proves the correctness of the algorithm. Indeed, $\delta^{(i)} = \delta^*$ if $f(\delta^{(i)}) = 0$. On the other hand, if either of the aforementioned scenarios apply, then combining it with $f(\delta^{(i)}) < 0$ and $g^{(i)} < 0$ certifies that $f(\delta^*) < 0$.

The following lemma shows that $\delta^{(i)}$ is monotonically decreasing while $f(\delta^{(i)})$ is monotonically increasing. Furthermore, $g^{(i)}$ is monotonically increasing except in the final iteration where it may remain unchanged. The lemma also illustrates the useful property that $|f(\delta^{(i)})|$ or $|g^{(i)}|$ decreases geometrically. These are well-known facts and similar statements can be found in e.g. Radzik [Rad98, Lemmas 3.1 & 3.2].

Lemma 9.2.1. *For every iteration $i \geq 2$, we have $\delta^* \leq \delta^{(i)} < \delta^{(i-1)}$, $f(\delta^*) \geq f(\delta^{(i)}) > f(\delta^{(i-1)})$ and $g^{(i)} \geq g^{(i-1)}$, where the last inequality holds at equality if and only if $g^{(i)} = \inf_{g \in \partial f(\delta^{(i)})} g$, $g^{(i-1)} = \sup_{g \in \partial f(\delta^{(i-1)})} g$ and $f(\delta^{(i)}) = 0$. Moreover,*

$$\frac{f(\delta^{(i)})}{f(\delta^{(i-1)})} + \frac{g^{(i)}}{g^{(i-1)}} \leq 1\,.$$

Proof. Since $f(\delta^{(i)}) \leq 0$ and $g^{(i)} < 0$, by concavity of f we have that $f(\delta) \leq f(\delta^{(i)}) + g^{(i)}(\delta - \delta^{(i)}) < f(\delta^{(i)}) \leq 0$, for all $\delta > \delta^{(i)}$. Given this, we must have $\delta^* \leq \delta^{(i)}$ since either $f(\delta^*) = 0 \geq f(\delta^{(i)})$ or $0 > f(\delta^*) = \max_{z \in \mathbb{R}} f(z) \geq f(\delta^{(i)})$. As $\delta^{(i)} = \delta^{(i-1)} - \frac{f(\delta^{(i-1)})}{g^{(i-1)}} < \delta^{(i-1)}$, since $f(\delta^{(i-1)}), g^{(i-1)} < 0$, we have $f(\delta^{(i-1)}) < f(\delta^{(i)})$. Furthermore, $g^{(i)} \geq g^{(i-1)}$ is immediate from the concavity of f.

To understand when $g^{(i)} = g^{(i-1)}$, we see by concavity that

$$g^{(i)} \geq \inf_{g \in \partial f(\delta^{(i)})} g \geq \frac{f(\delta^{(i-1)}) - f(\delta^{(i)})}{\delta^{(i-1)} - \delta^{(i)}} \geq \sup_{g \in \partial f(\delta^{(i-1)})} g \geq g^{(i-1)}.$$

To have equality throughout, we must therefore have that $g^{(i)}$ and $g^{(i-1)}$ are equal to the respective infimum and supremum. We must also have $f(\delta^{(i)}) = 0$ since

$$\frac{f(\delta^{(i-1)}) - f(\delta^{(i)})}{\delta^{(i-1)} - \delta^{(i)}} = \frac{f(\delta^{(i-1)}) - f(\delta^{(i)})}{\frac{f(\delta^{(i-1)})}{g^{(i-1)}}} = g^{(i-1)}\left(1 - \frac{f(\delta^{(i)})}{f(\delta^{(i-1)})}\right)$$

To have equality throughout, we must therefore have that $g^{(i)}$ and $g^{(i-1)}$ are equal to the respective infimum and supremum and that $f(\delta^{(i)}) = 0$.

Lastly, since f is concave

$$f(\delta^{(i-1)}) \leq f(\delta^{(i)}) + g^{(i)}(\delta^{(i-1)} - \delta^{(i)}) = f(\delta^{(i)}) + g^{(i)}\frac{f(\delta^{(i-1)})}{g^{(i-1)}}.$$

The moreover now follows by dividing both sides by $f(\delta^{(i-1)}) < 0$. □

Our analysis of the Newton–Dinkelbach method utilizes the Bregman divergence associated with f as a potential. Even though the original definition requires f to be differentiable and strictly concave, it can be naturally extended to our setting in the following way.

Definition 9.2.2. *Given a proper concave function $f : \mathbb{R} \to \overline{\mathbb{R}}$, the Bregman divergence associated with f is defined as*

$$D_f(\delta', \delta) := \begin{cases} f(\delta) + \sup\limits_{g \in \partial f(\delta)} g(\delta' - \delta) - f(\delta') & \text{if } \delta \neq \delta', \\ 0 & \text{otherwise.} \end{cases} \tag{9.4}$$

for all $\delta, \delta' \in \operatorname{dom}(f)$ such that $\partial f(\delta) \neq \emptyset$.

Since f is concave, the Bregman divergence is nonnegative. The next lemma shows that $D_f(\delta^*, \delta^{(i)})$ is monotonically decreasing except in the final iteration where it may remain unchanged. The proof is given in Appendix A.

Lemma 9.2.3. *For every iteration $i \geq 2$, we have $D_f(\delta^*, \delta^{(i)}) \leq D_f(\delta^*, \delta^{(i-1)})$ which holds at equality if and only if $g^{(i-1)} = \inf_{g \in \partial f(\delta^{(i-1)})} g$ and $f(\delta^{(i)}) = 0$.*

Proof. By Lemma 9.2.1, we know that $\delta^* \leq \delta^{(i)} < \delta^{(i-1)}$ and $0 \geq f(\delta^{(i)}) > f(\delta^{(i-1)})$. Hence,

$$\begin{aligned} D_f(\delta^*, \delta^{(i-1)}) &= f(\delta^{(i-1)}) + \sup_{g \in \partial f(\delta^{(i-1)})} g(\delta^* - \delta^{(i-1)}) - f(\delta^*) \\ &\geq f(\delta^{(i-1)}) + g^{(i-1)}(\delta^{(i)} - \delta^{(i-1)}) + g^{(i-1)}(\delta^* - \delta^{(i)}) - f(\delta^*) \\ &= 0 + g^{(i-1)}(\delta^* - \delta^{(i)}) - f(\delta^*) \\ &\geq f(\delta^{(i)}) + g^{(i-1)}(\delta^* - \delta^{(i)}) - f(\delta^*) && \text{(by concavity of } f\text{)} \\ &\geq f(\delta^{(i)}) + \sup_{g \in \partial f(\delta^{(i)})} g(\delta^* - \delta^{(i)}) - f(\delta^*) \\ &= D_f(\delta^*, \delta^{(i)}). \end{aligned}$$

For the equality condition, note that the first two inequalities hold at equality precisely when $g^{(i-1)} = \inf_{g \in \partial f(\delta^{(i-1)})} g$ and $f(\delta^{(i)}) = 0$. If $f(\delta^{(i)}) = 0$, then $\delta^{(i)} = \delta^*$, and hence the third inequality holds at equality as well. □

To accelerate this classical method, we perform an aggressive guess $\delta' := 2\delta - \delta^{(i)} < \delta$ on the next point at the end of every iteration i. We call this procedure look-ahead, which is implemented

on Lines 7–10 of Algorithm 9.1. Let $g' \in \partial f(\delta')$ be the supergradient returned by the oracle. If $-\infty < f(\delta') < 0$ and $g' < 0$, then the next point and supergradient are set to $\delta^{(i+1)} := \delta'$ and $g^{(i+1)} := g'$ respectively as $\delta' \geq \delta^*$. In this case, we say that look-ahead is *successful* in iteration i. Otherwise, we proceed as usual by taking $\delta^{(i+1)} := \delta$ and $g^{(i+1)} := g$. It is easy to verify that Lemmas 9.2.1 and 9.2.3 also hold for Algorithm 9.1.

Algorithm 9.1: Look-ahead Newton

Input : Value and supergradient oracles for a proper concave function f, an initial point $\delta^{(1)} \in \mathrm{dom}(f)$ and supergradient $g^{(1)} \in \partial f(\delta^{(1)})$ where $f(\delta^{(1)}) \leq 0$ and $g^{(1)} < 0$.
Output : The largest root of f if it exists; report `NO ROOT` otherwise.

1 $i \leftarrow 1$
2 **while** $f(\delta^{(i)}) < 0$ **do**
3 $\delta \leftarrow \delta^{(i)} - f(\delta^{(i)})/g^{(i)}$
4 $g \in \partial f(\delta)$ /* Empty if $f(\delta) = -\infty$ */
5 **if** $f(\delta) = -\infty$ ***or*** $(f(\delta) < 0$ ***and*** $g \geq 0)$ **then**
6 **return** `NO ROOT`
7 $\delta' \leftarrow 2\delta - \delta^{(i)}$ /* Look-ahead guess */
8 $g' \in \partial f(\delta')$ /* Empty if $f(\delta') = -\infty$ */
9 **if** $-\infty < f(\delta') < 0$ ***and*** $g' < 0$ **then** /* Is the guess successful? */
10 $\delta \leftarrow \delta', g \leftarrow g'$
11 $\delta^{(i+1)} \leftarrow \delta, g^{(i+1)} \leftarrow g$
12 $i \leftarrow i + 1$
13 **return** $\delta^{(i)}$

If look-ahead is successful, then we have made significant progress. Otherwise, by our choice of δ', we learn that we are not too far away from δ^*. The next lemma demonstrates the advantage of using the look-ahead Newton–Dinkelbach method. It exploits the proximity to δ^* to produce a geometric decay in the Bregman divergence of $\delta^{(i)}$ and δ^*.

Lemma 9.2.4. *For every iteration $i > 2$ in Algorithm 9.1, we have $D_f(\delta^*, \delta^{(i)}) < \frac{1}{2} D_f(\delta^*, \delta^{(i-2)})$.*

Proof. Fix an iteration $i > 2$ of Algorithm 9.1. Let $g_+^{(i)} = \min_{g \in \partial f(\delta^{(i)})} g$ denote the right derivative of f at $\delta^{(i)}$. From Lemma 9.2.1, we know that $\delta^* \leq \delta^{(i)} < \delta^{(i-1)} < \delta^{(i-2)}$, $0 \geq f(\delta^*) \geq f(\delta^{(i)}) > f(\delta^{(i-1)}) > f(\delta^{(i-2)})$ and $0 > g_+^{(i)} \geq g^{(i-1)} > g^{(i-2)}$. Since $\delta^* \leq \delta^{(i)}$, we see that $D_f(\delta^*, \delta^{(i)}) = f(\delta^{(i)}) + g_+^{(i)}(\delta^* - \delta^{(i)}) - f(\delta^*)$.

Assume first that the look-ahead step in iteration $i-1$ was successful. We now claim that $0 < -2g_+^{(i)} \leq -g^{(i-1)}$. To see this, we have that

$$\begin{aligned} f(\delta^{(i-1)}) &\leq f(\delta^{(i)}) + g_+^{(i)}(\delta^{(i-1)} - \delta^{(i)}) && \text{(by concavity of } f) \\ &\leq g_+^{(i)}(\delta^{(i-1)} - \delta^{(i)}) && \text{(as } f(\delta^{(i)}) \leq 0) \\ &= 2g_+^{(i)} \frac{f(\delta^{(i-1)})}{g^{(i-1)}} && \text{(by definition of the accelerated step)} \end{aligned}$$

The desired inequality follows by multiplying through by $-\frac{g^{(i-1)}}{f(\delta^{(i-1)})} < 0$.

Using the above inequality, we compare Bregman divergences as follows:

$$\begin{aligned}
D_f(\delta^*, \delta^{(i-1)}) &\geq f(\delta^{(i-1)}) + g^{(i-1)}(\delta^* - \delta^{(i-1)}) - f(\delta^*) && (D_f \text{ is a maximum over supergradients}) \\
&> g^{(i-1)}(\delta^* - \delta^{(i)}) - f(\delta^*) && (f(\delta^{(i-1)}) + g^{(i-1)}(\delta^{(i)} - \delta^{(i-1)}) = -f(\delta^{(i-1)}) > 0) \\
&\geq g^{(i-1)}(\delta^* - \delta^{(i)}) && (-f(\delta^*) \geq 0) \\
&\geq 2g_+^{(i)}(\delta^* - \delta^{(i)}) && (-g^{(i-1)} \geq -2g_+^{(i)} \text{ and } \delta^{(i)} > \delta^*) \\
&\geq 2(f(\delta^{(i)}) + g_+^{(i)}(\delta^* - \delta^{(i)}) - f(\delta^*)) && (f(\delta^*) \geq f(\delta^{(i)})) \\
&= 2D_f(\delta^*, \delta^{(i)}). && (\text{by choice of } g_+^{(i)})
\end{aligned}$$

The desired inequality now follows from $D_f(\delta^*, \delta^{(i-2)}) > D_f(\delta^*, \delta^{(i-1)})$ by Lemma 9.2.3.

Now assume that the look-ahead step at iteration $i-1$ was unsuccessful. This implies that $2\delta^{(i)} - \delta^{(i-1)} \leq \delta^* \Leftrightarrow 2(\delta^{(i)} - \delta^*) \leq \delta^{(i-1)} - \delta^*$, i.e. that the look-ahead step "went past or exactly to" δ^*. We compare Bregman-divergences as follows:

$$\begin{aligned}
D_f(\delta^*, \delta^{(i-2)}) &\geq f(\delta^{(i-2)}) + g^{(i-2)}(\delta^* - \delta^{(i-2)}) - f(\delta^*) && (D_f \text{ is a maximum over supergradients}) \\
&\geq g^{(i-2)}(\delta^* - \delta^{(i-1)}) - f(\delta^*) && (f(\delta^{(i-2)}) + g^{(i-2)}(\delta^{(i-1)} - \delta^{(i-2)}) \geq 0) \\
&\geq g^{(i-2)}(\delta^* - \delta^{(i-1)}) && (-f(\delta^*) \geq 0) \\
&> g_+^{(i)}(\delta^* - \delta^{(i-1)}) && (0 > g_+^{(i)} > g^{(i-2)} \text{ and } \delta^{(i-1)} > \delta^*) \\
&\geq 2g_+^{(i)}(\delta^* - \delta^{(i)}) && (0 > g_+^{(i)} \text{ and } \delta^{(i-1)} - \delta^* \geq 2(\delta^{(i)} - \delta^*)) \\
&\geq 2(f(\delta^{(i)}) + g_+^{(i)}(\delta^* - \delta^{(i)}) - f(\delta^*)) && (\text{as } f(\delta^*) \geq f(\delta^{(i)})) \\
&= 2D_f(\delta^*, \delta^{(i)}). && (\text{by choice of } g_+^{(i)})
\end{aligned}$$

This concludes the proof. □

In the remaining of this section, we apply the accelerated Newton–Dinkelbach method to linear fractional combinatorial optimization and linear fractional programming. The application to parametric submodular function minimization can be found in full detail in [DKNV21].

9.2.1 Linear Fractional Combinatorial Optimization

The problem (9.1) with $\mathcal{D} \subseteq \{0,1\}^m$ is known as *linear fractional combinatorial optimization*. Radzik [Rad92] showed that the Newton–Dinkelbach method applied to the function $f(\delta)$ as in (9.2) terminates in a strongly polynomial number of iterations. Recall that $f(\delta) = \min_{x \in \mathcal{D}} \langle c - \delta d, x \rangle$. By the assumption $\langle d, x \rangle > 0$ for all $x \in \mathcal{D}$, this function is concave, strictly decreasing, finite and piecewise-linear. Hence, it has a unique root. Moreover, $f(\delta) < 0$ and $\partial f(\delta) \cap \mathbb{R}_{<0} \neq \emptyset$ for sufficiently large δ. To implement the value and supergradient oracles, we assume that a linear optimization oracle over $\mathcal{D}$ is available, i.e. it returns an element in $\arg\min_{x \in \mathcal{D}} (c - \delta d)^\top x$ for any $\delta \in \mathbb{R}$.

Our result for the accelerated variant improves the state-of-the-art bound $O(m^2 \log m)$ by Wang et al. [WYZ06] on the standard Newton–Dinkelbach method. We will need the following lemma, given by Radzik and credited to Goemans in [Rad98]. It gives a strongly polynomial bound on the length of a geometrically decreasing sequence of sums.

Lemma 9.2.5 ([Rad98]). *Let $c \in \mathbb{R}_+^m$ and $x^{(1)}, x^{(2)}, \ldots, x^{(k)} \in \{-1, 0, 1\}^m$. If $0 < c^\top x^{(i+1)} \leq \frac{1}{2} c^\top x^{(i)}$ for all $i < k$, then $k = O(m \log m)$.*

Theorem 9.2.6. *Algorithm 9.1 converges in $O(m \log m)$ iterations for linear fractional combinatorial optimization problems.*

Proof. Observe that Algorithm 9.1 terminates in a finite number of iterations because f is piecewise linear. Let $\delta^{(1)} > \delta^{(2)} > \cdots > \delta^{(k)} = \delta^*$ denote the sequence of iterates at the start of Algorithm 9.1. Since f is concave, we have $D_f(\delta^*, \delta^{(i)}) \geq 0$ for all $i \in [k]$. For each $i \in [k]$, pick $x^{(i)} \in \arg\min_{x \in \mathcal{D}} (c - \delta^{(i)} d)^\top x$ which maximizes $d^\top x$. This is well-defined because f is finite. Note that $-d^\top x^{(i)} = \min \partial f(\delta^{(i)})$. As $f(\delta^*) = 0$, the Bregman divergence of $\delta^{(i)}$ and δ^* can be written as

$$D_f(\delta^*, \delta^{(i)}) = f(\delta^{(i)}) + \max_{g \in \partial f(\delta^{(i)})} g(\delta^* - \delta^{(i)}) = (c - \delta^{(i)} d)^\top x^{(i)} - d^\top x^{(i)} (\delta^* - \delta^{(i)}) = (c - \delta^* d)^\top x^{(i)} .$$

According to Lemma 9.2.4, $(c - \delta^* d)^\top x^{(i)} = D_f(\delta^*, \delta^{(i)}) < \frac{1}{2} D_f(\delta^*, \delta^{(i-2)}) = \frac{1}{2}(c - \delta^* d)^\top x^{(i-2)}$ for all $3 \leq i \leq k$. By Lemma 9.2.3, we also know that $D_f(\delta^*, \delta^{(i)}) > 0$ for all $1 \leq i \leq k-2$. Thus, applying Lemma 9.2.5 yields $k = O(m \log m)$. □

9.2.2 Linear Fractional Programming

We next consider *linear fractional programming*, an extension of (9.1) with the assumption that the domain $\mathcal{D} \subseteq \mathbb{R}^m$ is a polyhedron, but removing the condition $\langle d, x \rangle > 0$ for $x \in \mathcal{D}$. For $c, d \in \mathbb{R}^m$, the problem is

$$\inf \frac{\langle c, x \rangle}{\langle d, x \rangle} \quad \text{s.t. } \langle d, x \rangle > 0, \ x \in \mathcal{D} . \tag{F}$$

For the problem to be meaningful, we assume that $\mathcal{D} \cap \{x : d^\top x > 0\} \neq \emptyset$. The common form in the literature assumes $\langle d, x \rangle > 0$ for all $x \in \mathcal{D}$ as in (9.1); we consider the more general setup for the purpose of solving M2VPI systems in the full version [DKNV21]. It is easy to see that any linear fractional combinatorial optimization problem on a domain $\mathcal{X} \subseteq \{0,1\}^m$ can be cast as a linear fractional program with the polytope $\mathcal{D} = \operatorname{conv}(\mathcal{X})$ because $c^\top \bar{x} / d^\top \bar{x} \geq \min_{x \in \mathcal{X}} c^\top x / d^\top x$ for all $\bar{x} \in \mathcal{D}$. The next theorem characterizes when (F) is unbounded.

Theorem 9.2.7. *If $\mathcal{D} \cap \{ x : d^\top x > 0 \} \neq \emptyset$, then the optimal value of (F) is $-\infty$ if and only if at least one of the following two conditions hold:*

1. *There exists $x \in \mathcal{D}$ such that $c^\top x < 0$ and $d^\top x = 0$;*
2. *There exists $r \in \mathbb{R}^m$ such that $c^\top r < 0$, $d^\top r = 0$ and $x + \lambda r \in \mathcal{D}$ for all $x \in \mathcal{D}, \lambda \geq 0$.*

Proof. By the Minkowski-Weyl Theorem, the polyhedron $\bar{\mathcal{D}} := \mathcal{D} \cap \{ x : d^\top x \geq 0 \}$ can be written as

$$\bar{\mathcal{D}} = \left\{ \sum_{i=1}^{k} \lambda_i g_i + \sum_{j=1}^{\ell} \nu_j h_j : \lambda \geq 0, \nu \geq 0, \|\lambda\|_1 = 1 \right\}$$

for some vectors $g_1, \ldots, g_k$ and $h_1, \ldots, h_\ell$. Note that $d^\top g_i \geq 0$ for all $i \in [k]$ and $d^\top h_j \geq 0$ for all $j \in [\ell]$. Let $x^\circ \in \mathcal{D} \cap \{x : d^\top x > 0\}$. If there exists $i \in [k]$ such that $c^\top g_i < 0$ and $d^\top g_i = 0$ or $j \in [\ell]$ such that $c^\top h_j < 0$ and $d^\top h_j = 0$, then,

$$\lim_{\lambda \nearrow 1} \frac{c^\top(\lambda g_i + (1-\lambda) x^\circ)}{d^\top(\lambda g_i + (1-\lambda) x^\circ)} = -\infty \quad \text{or} \quad \lim_{\lambda \to \infty} \frac{c^\top(x^\circ + \lambda h_j)}{d^\top(x^\circ + \lambda h_j)} = -\infty$$

as in Condition 1 or Condition 2.

Otherwise, the fractional value of any element in $\mathcal{D} \cap \{x : d^\top x > 0\}$ can be lower bounded by

$$\frac{c^\top(\sum_{i=1}^k \lambda_i g_i + \sum_{j=1}^\ell \nu_j h_j)}{d^\top(\sum_{i=1}^k \lambda_i g_i + \sum_{j=1}^\ell \nu_j h_j)} \geq \frac{\sum_{i\in[k],d^\top g_i>0} \lambda_i c^\top g_i + \sum_{j\in[\ell],d^\top h_j>0} \nu_j c^\top h_j}{\sum_{i\in[k],d^\top g_i>0} \lambda_i d^\top g_i + \sum_{j\in[\ell],d^\top h_j>0} \nu_j d^\top h_j}$$
$$\geq \min\left\{ \min_{i\in[k],d^\top g_i>0} \frac{c^\top g_i}{d^\top g_i}, \min_{j\in[\ell],d^\top h_j>0} \frac{c^\top h_j}{d^\top h_j} \right\},$$

where the last expression is finite by the assumption that $\mathcal{D} \cap \{ x : d^\top x > 0 \}$ is non-empty. □

Example 9.2.8. *Unlike in linear programming, the optimal value may not be attained even if it is finite. Consider the instance given by* $\inf(-x_1 + x_2)/(x_1 + x_2)$ *subject to* $x_1 + x_2 > 0$ *and* $-x_1 + x_2 = 1$. *The numerator is equal to 1 for any feasible solution, while the denominator can be made arbitrarily large. Hence, the optimal value of this program is 0, which is not attained in the feasible region.*

We use the Newton–Dinkelbach method for f as in (9.2), that is, $f(\delta) = \inf_{x\in\mathcal{D}}(c - \delta d)^\top x$. Since $\mathcal{D} \neq \emptyset$, $f(\delta) < \infty$ for all $\delta \in \mathbb{R}$. By the Minkowski–Weyl theorem, there exist finitely many points $P \subseteq \mathcal{D}$ such that $f(\delta) = \min_{x\in P}(c - \delta d)^\top x$ for all $\delta \in \operatorname{dom}(f)$. Hence, f is concave and piecewise linear. Observe that $f(\delta) > -\infty$ if and only if every ray r in the recession cone of $\mathcal{D}$ satisfies $(c - \delta d)^\top r \geq 0$. For f to be proper, we need to assume that Condition 2 in Theorem 9.2.7 does not hold. Moreover, we require the existence of a point $\delta' \in \operatorname{dom}(f)$ such that $f(\delta') = (c - \delta' d)^\top x' \leq 0$ for some $x' \in \mathcal{D}$ with $d^\top x' > 0$. It follows that f has a root or attains its maximum because $\operatorname{dom}(f)$ is closed. We are ready to characterize the optimal value of (F) using f.

Lemma 9.2.9. *Assume that there exists* $\delta' \in \operatorname{dom}(f)$ *such that* $f(\delta') = (c - \delta' d)^\top x' \leq 0$ *for some* $x' \in \mathcal{D}$ *with* $d^\top x' > 0$. *If f has a root, then the optimal value of* (F) *is equal to the largest root and is attained. Otherwise, the optimal value is* $-\infty$.

Proof. Recall the definition of $\delta^* = \max(\{\delta : f(\delta) = 0\} \cup \arg\max f(\delta))$. By our assumption on f, there exists $x^* \in \mathcal{D}$ such that $f(\delta^*) = (c - \delta^* d)^\top x^*$ and $d^\top x^* > 0$. If f has a root, then $f(\delta^*) = 0$. This implies that $c^\top x / d^\top x \geq \delta^* = c^\top x^* / d^\top x^*$ for all $x \in \mathcal{D}$ with $d^\top x > 0$ as desired. Next, assume that f does not have a root. Then $f(\delta^*) < 0$ and $0 \in \partial f(\delta^*)$. By convexity, there exists $\bar{x} \in \mathcal{D}$ such that $(c - \delta^* d)^\top \bar{x} = f(\delta^*) < 0$ and $d^\top \bar{x} = 0$. Then $c^\top \bar{x} < 0$, so $\bar{x}$ is a point as in Condition 1 of Theorem 9.2.7.

□

9.3 Fractional LP with circuit imbalances

In this section we consider the fractional analogue of LP

Recall from (9.2) that solving the problem Fractional LP is equivalent to finding the unique $\lambda^* \in \mathbb{R}$ such that the optimal solution to

$$\begin{aligned} \min_x \quad & \langle c, x\rangle - \lambda^* \langle q, x\rangle \\ \text{s.t.} \quad & \mathbf{A}x = b, \\ & x \geq \mathbf{0} \end{aligned} \tag{9.5}$$

has objective value 0. Solving (9.5) for a fixed value λ is equivalent to solving LP$(\mathbf{A}, b, c - \lambda^* q)$, a problem whose solutions we studied extensively in Chapters 4 and 7.

We now sketch how can be solved within $O(n \log \kappa)$ calls to any exact linear system solver. For simplicity let us assume boundedness, the argument will easily extend to unbounded instances.

We further assume that $c \in W$ and $q \in W$. Note that otherwise the problem is invariant under replacing c with $\Pi_W(c)$ and q with $\Pi_W(q)$. For an angle $\alpha \in [-\pi, \pi]$ we associate the value λ such that $\alpha = \sphericalangle(c, c - \lambda q)$. In abuse of notation we let $\sphericalangle$ range from $-\pi/2$ to $\pi/2$, which is well-defined as we operate in the 2-dimensional space spanned by c and q. Within $\tilde{O}(\log \kappa_{\mathbf{A}})$ iteration binary search we are able to find angles α^- and α^+ such that $\alpha^+ \geq \alpha^-$, $\alpha^+ - \alpha^- = \text{poly}(n)^{-\Theta(1)} \kappa_{\mathbf{A}}^{-\Theta(1)}$ and the corresponding values λ^- and λ^+ sandwich λ^*, i.e., $\lambda^- \leq \lambda^* \leq \lambda^+$.

With the objectives

$$c^- := \frac{c - \lambda^- q}{\|c - \lambda^- q\|}, \quad \text{and} \quad c^+ := \frac{c - \lambda^+ q}{\|c - \lambda^+ q\|} \tag{9.6}$$

and noting that $\sphericalangle(c^-, c^+) = \text{poly}(n)^{-\Theta(1)} \kappa_{\mathbf{A}}^{-\Theta(1)}$ and so in particular $\sphericalangle(c^-, c^*) = \text{poly}(n)^{-\Theta(1)} \kappa_{\mathbf{A}}^{-\Theta(1)}$ and $\sphericalangle(c^*, c^+) = \text{poly}(n)^{-\Theta(1)} \kappa_{\mathbf{A}}^{-\Theta(1)}$, where $c^* = c - \lambda^* q$. Note further that $c^-, c^*, c^+ \in W$ by the assumption that $c, q \in W$ and hence $\cos(\sphericalangle(c^-, W)) = \cos(\sphericalangle(c^+, W)) = \cos(\sphericalangle(c, W)) = 1$. We can apply Theorem 3.5.9 to obtain a primal-dual optimal solution $(\tilde{x}, \tilde{s})$ to $\text{LP}(\mathbf{A}, b, c^+)$. By the statement of the theorem we have that for the set

$$\begin{aligned} R &:= \left\{ i \in [n] : \frac{\tilde{s}_i}{\|\tilde{s}\|} > \frac{n(\kappa_W + 1)\sphericalangle(c, \tilde{c})}{\cos(\sphericalangle(\tilde{c}, W))} \right\} = \left\{ i \in [n] : \frac{\tilde{s}_i}{\|\tilde{s}\|} > n(\kappa_W + 1)\sphericalangle(c, \tilde{c})) \right\} \\ &\supseteq \left\{ i \in [n] : \frac{\tilde{s}_i}{\|\tilde{s}\|} > n(\kappa_W + 1)\text{poly}(n)^{-\Theta(1)} \kappa_{\mathbf{A}}^{-\Theta(1)} \right\}, \end{aligned}$$

the variables $x^*_R = \mathbf{0}$ in any optimal solution x^* to $\text{LP}(\mathbf{A}, b, c^*)$. R is non-empty if the constants in the exponents are chosen appropriately and so the primal variables in R can be deleted and one recurses on the variable set $[n] \setminus R$. This, together with the blackbox LP solvers gives following result:

Theorem 9.3.1. *$\text{LP}(\mathbf{A}, b, c, q)$ can be solved in $\widetilde{O}(n^2 m \Psi(\mathbf{A}) \log^2 \kappa_{\mathbf{A}})$ time.*

Acronyms

2VPI	Two Variables Per Inequality
AS	Affine Scaling
CI	circuit imbalance
CP	Central Path
DMDP	Deterministic Markov Decision Processes
gcd	Greatest Common Divisor
IP	Integer Programming
IPM	Interior-Point-Methods
lcm	Least Common Multiple
LLS	Layered-Least-Squares
LP	Linear Programming
LSS	Linear System Solve
M2VPI	Monotone Two Variables Per Inequality
MCP	Max Central Path
MDP	Markov Decision Processes
MTY	Mizuno-Todd-Ye
PCA	Principal Component Analysis
SCC	Strongly Connected Component
SLLS	Subspace LLS
SVD	Singular Value Decomposition
TU	Totally Unimodular
VY	Vavasis-Ye Algorithm in [VY96]

List of Symbols

Symbol	Introduction	Description
$\sphericalangle$	Equation (3.39) on Page 49	Angle between two vectors (or linear spaces)
aug	Section 8.2 on Page 207	circuit augmentation
$\mathcal{B}$	Chapter 2 on Page 18	set of bases
D	Equation (9.4) on Page 229	Bregman divergence
CP	Equation (5.23) on Page 115	segment of the central path between two inputs μ_1 and μ_0.
$\mathfrak{C}$	Definition 7.2.4 on Page 160	set of lifting certificates with costs $> M$
$\bar{\chi}$	Equation (3.13) on Page 34	condition number depending on subspace W
$\bar{\chi}^*$	Section 3.4 on Page 39	optimal condition number $\bar{\chi}$ under rescalings $\mathbf{D}W$ of W.
χ	Equation (7.3) on Page 159	condition number depending on subspace W and vector d
$\widetilde{\chi}$	Equation (7.5) on Page 160	condition number depending on subspace W, vector d and $\varepsilon > 0$.
$\mathcal{CD}$	Section 7.3 on Page 162	set of sign-consistent circuit decompositions
κ	Definition 3.1.1 on Page 21	condition number depending on subspace W.
κ^*	Section 3.4 on Page 39	optimal condition number $\kappa_{\mathbf{D}W}$ under rescalings $\mathbf{D}W$ of W.
$\mathcal{C}$	Chapter 2 on Page 18	set of circuits
cl	Chapter 2 on Page 18	closure of a set of variables in the matroidal sense
$\sqsubseteq$	Chapter 2 on Page 19	binary relation - conformality of two vectors.
Ψ	Section 7.4.2 on Page 165	runtime of approximate solvers
Υ	Equation (6.1) on Page 141	Sonnevend et al. curvature
δ	Definition 3.3.10 on Page 37	condition number corresponding to smallest non-zero angle of subspaces spanned by columns of a matrix
$\mathfrak{D}$	Chapter 2 on Page 17	diagonal positive definite matrices.
dom	Chapter 2 on Page 18	effective domain
$\mathcal{D}$	Equation (2.3) on Page 19	dual feasible region
$\mathcal{D}_{++}$	Equation (4.3) on Page 63	set of strictly feasible points in the dual $\mathcal{D}$
$\mathcal{E}$	Chapter 2 on Page 18	set of elementary vectors
$\mathfrak{F}$	Definition 7.2.5 on Page 160	set of Farkas certificates
Λ	Equation (3.35) on Page 47	set of variables that are negative in first argument of positive in second argument
$\mathcal{G}$	Definition 3.6.2 on Page 51	Graver basis
gcd	Chapter 2 on Page 18	greatest common divisor

Symbol	Introduction	Description
$\mathscr{H}$	Section 7.3 on Page 163	Algorithm implementing System 7.4
$\hbar$	Section 7.3 on Page 163	Algorithm implementing System 7.3
H	Section 3.5 on Page 47	Hoffman constant
im	Chapter 2 on Page 17	image
$\mathbb{Z}$	Chapter 2 on Page 17	integers
$\mathcal{I}$	Equation (6.2) on Page 141	curvature integral
ker	Chapter 2 on Page 17	kernel
lcm	Chapter 2 on Page 18	least common multiple
ℓ	Section 5.4 on Page 118	The lifting map for as used in Chapters 5 and 6
$\mathcal{M}_2$	Section 9.1.2 on Page 224	matrices with at most two nonzeros per column
$\mathcal{M}$	Chapter 2 on Page 19	induced matroid
MCP	Section 5.5 on Page 131	max central path
$\mathcal{N}^{-\infty}$	Equation (5.2) on Page 103	$-\infty$-neighbordhood of central path
$\mathcal{N}$	Section 4.3.1 on Page 63	ℓ_2-neighbordhood of central path
nnz	Item (3) on Page 165	number of non-zeros of a matrix
C_f	Section 7.4.2 on Page 166	parameter depending on guess M that gives a norm-bound for blackbox-feasibility
C_o	Section 7.4.2 on Page 167	parameter depending on guess M that gives a norm-bound for blackbox-optimization
$\overline{\mu}$	Section 5.1 on Page 102	Normalized duality gap of element near central path
ν	Equation (2.1) on Page 17	p adic number of an argument $n \in \mathbb{N}$, i.e., the maximal power α of p such that $p^\alpha \mid n$.
$\bar{d}$	Line 4 on Page 178	Perturbed d
γ	Definition 5.3.1 on Page 115	Polarization of central path
poly	Chapter 2 on Page 19	polynomial running time in arguments
$\mathbb{N}$	Chapter 2 on Page 17	positive integers
$\mathcal{P}$	Equation (2.3) on Page 19	primal feasible region
$\mathcal{P}_{++}$	Equation (4.3) on Page 63	set of strictly feasible points in the primal $\mathcal{P}$
Ψ	Section 4.4 on Page 78	overall potential in analysis.
range	Chapter 2 on Page 17	range of a matrix or an operator
rk	Chapter 2 on Page 17	rank
$\mathbb{Q}$	Chapter 2 on Page 17	rational numbers
$\mathbb{R}$	Chapter 2 on Page 17	real numbers
$\overline{\mathbb{R}}$	Chapter 2 on Page 17	extended real numbers $\mathbb{R} \cup \{\pm\infty\}$
$\mathbb{R}_+$	Chapter 2 on Page 17	nonnegative reals
$\mathbb{R}_{++}$	Chapter 2 on Page 17	positive reals
Rs	Section 4.3.2 on Page 65	dual residuals
ϱ	Equation (5.27) on Page 119	residuals in subspace lls
Rx	Section 4.3.2 on Page 65	primal residuals
ϱ	Section 4.4 on Page 78	potential between two variables in analysis.
δ	Section 4.3.2 on Page 65	error to dual variable $s(\mu)$ in Chapter 4
span	Chapter 2 on Page 18	linear span of the arguments.
Δ	Equation (3.1) on Page 23	maximum subdeterminant

Symbol	Introduction	Description
$\dot{\Lambda}$	Equation (3.1) on Page 23	lcm of subdeterminants
$\mathcal{S}$	Line 10 on Page 122	the set of i-dimensional subspaces, where i is given as argument.
supp	Chapter 2 on Page 17	support
τ	Equation (5.28) on Page 119	threshold for singular values to be chosen in space of *cheap* singular vectors.
tw	Definition 7.2.11 on Page 161	treewidth of a matrix

Index

Bibliography

[ABGJ18] X. Allamigeon, P. Benchimol, S. Gaubert, and M. Joswig. "Log-Barrier Interior Point Methods Are Not Strongly Polynomial". *SIAM Journal on Applied Algebra and Geometry* 2:1, 2018, pp. 140–178.

[ABGJ21] X. Allamigeon, P. Benchimol, S. Gaubert, and M. Joswig. "What Tropical Geometry Tells Us about the Complexity of Linear Programming". *SIAM Review* 63:1, 2021, pp. 123–164. URL: https://doi.org/10.1137/20m1380211.

[AGV22] X. Allamigeon, S. Gaubert, and N. Vandame. "No Self-Concordant Barrier Interior Point Method is Strongly Polynomial". In: *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2022. Association for Computing Machinery, Rome, Italy, 2022, pp. 515–528. ISBN: 9781450392648. URL: https://doi.org/10.1145/3519935.3519997.

[AK04] G. Appa and B. Kotnyek. "Rational and integral k-regular matrices". *Discrete Mathematics* 275:1-3, 2004, pp. 1–15.

[All+22] X. Allamigeon, D. Dadush, G. Loho, B. Natura, and L. A. Végh. *Interior point methods are not worse than Simplex*. To appear in Annual Symposium on Foundations of Computer Science (FOCS 2022). 2022. URL: https://arxiv.org/abs/2206.08810.

[AMO93] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., 1993.

[AMV22] K. Axiotis, A. Madry, and A. Vladu. "Faster Sparse Minimum Cost Flow by Electrical Flow Localization". In: *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*. 2022.

[AS80] B. Aspvall and Y. Shiloach. "A Polynomial Time Algorithm for Solving Systems of Linear Inequalities with Two Variables per Inequality". *SIAM J. Comput.* 9:4, 1980, pp. 827–845.

[AY13] N. Alon and R. Yuster. "Matrix Sparsification and Nested Dissection over Arbitrary Fields". *J. ACM* 60:4, 2013. ISSN: 0004-5411. URL: https://doi.org/10.1145/2508028.2505989.

[BDF16] S. Borgwardt, J. A. De Loera, and E. Finhold. "Edges versus circuits: a hierarchy of diameters in polyhedra". *Advances in Geometry* 16:4, 2016, pp. 511–530.

[BE15] S. Bubeck and R. Eldan. "The entropic barrier: a simple and optimal universal self-concordant barrier". In: *Proceedings of The 28th Conference on Learning Theory*. Ed. by P. Grünwald, E. Hazan, and S. Kale. Vol. 40. Proceedings of Machine Learning Research. PMLR, Paris, France, 2015, pp. 279–279. URL: https://proceedings.mlr.press/v40/Bubeck15b.html.

[BFH15] S. Borgwardt, E. Finhold, and R. Hemmecke. "On the circuit diameter of dual transportation polyhedra". *SIAM Journal on Discrete Mathematics* 29:1, 2015, pp. 113–121.

[BH74] J. R. Bunch and J. E. Hopcroft. "Triangular factorization and inversion by fast matrix multiplication". *Mathematics of Computation* 28:125, 1974, pp. 231–231. URL: https://doi.org/10.1090/s0025-5718-1974-0331751-8.

[Bla76] R. G. Bland. "On the generality of network flow theory". Presented at the ORSA/TIMS Joint National Meeting, Miami, FL. 1976.

[Bon+14] N. Bonifas, M. Di Summa, F. Eisenbrand, N. Hähnle, and M. Niemeier. "On sub-determinants and the diameter of polyhedra". *Discrete & Computational Geometry* 52:1, 2014, pp. 102–115.

[Bor12] K. H. Borgwardt. *The Simplex Method: a probabilistic analysis*. Vol. 1. Springer Science & Business Media, 2012.

[BR13] T. Brunsch and H. Röglin. "Finding short paths on polytopes by the shadow vertex algorithm". In: *Proceedings of the 40th International Colloquium on Automata, Languages, and Programming (ICALP)*. Springer. 2013, pp. 279–290.

[Bra] J. van den Brand. "Unifying Matrix Data Structures: Simplifying and Speeding up Iterative Algorithms". In: *Symposium on Simplicity in Algorithms (SOSA)*, pp. 1–13. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611976496.1.

[Bra+20] J. van den Brand, Y.-T. Lee, D. Nanongkai, R. Peng, T. Saranurak, A. Sidford, Z. Song, and D. Wang. "Bipartite matching in nearly-linear time on moderately dense graphs". In: *IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 919–930.

[Bra+21] J. van den Brand, Y. P. Liu, Y.-T. Lee, T. Saranurak, A. Sidford, Z. Song, and D. Wang. "Minimum Cost Flows, MDPs, and L1-Regression in Nearly Linear Time for Dense Instances". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing (STOC)*. 2021, pp. 859–869.

[Bra20] J. van den Brand. "A Deterministic Linear Program Solver in Current Matrix Multiplication Time". In: *Proceedings of the 2020 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2020.

[BSY18] S. Borgwardt, T. Stephen, and T. Yusun. "On the circuit diameter conjecture". *Discrete & Computational Geometry* 60:3, 2018, pp. 558–587.

[BT89] F. Barahona and É. Tardos. "Note on Weintraub's minimum-cost circulation algorithm". *SIAM Journal on Computing* 18:3, 1989, pp. 579–583.

[BTSS20] J. van den Brand, Y. Tat Lee, A. Sidford, and Z. Song. "Solving tall dense linear programs in nearly linear time". In: *Proceedings of the 52nd Annual ACM Symposium on Theory of Computing (STOC)*. 2020, pp. 775–788.

[BV20] S. Borgwardt and C. Viss. "An implementation of steepest-descent augmentation for linear programs". *Operations Research Letters* 48:3, 2020, pp. 323–328.

[Cam64] P. Camion. "Matrices Totalement Unimodulaires et Problemes Combinatoires". EUR 1632.1. PhD thesis. Communauté européenne de l'énergie atomique (EURATOM), 1964.

[Cam65] P. Camion. "Characterization of totally unimodular matrices". *Proceedings of the American Mathematical Society* 16:5, 1965, pp. 1068–1068. URL: https://doi.org/10.1090/s0002-9939-1965-0180568-2.

[Ced57] I. Cederbaum. "Matrices All of Whose Elements and Subdeterminants are 1, −1, or 0". *Journal of Mathematics and Physics* 36:1-4, 1957, pp. 351–361.

[CGST86] W. Cook, A. M. Gerards, A. Schrijver, and É. Tardos. "Sensitivity theorems in integer linear programming". *Mathematical Programming* 34:3, 1986, pp. 251–264.

[Che+22] L. Chen, R. Kyng, Y. P. Liu, R. Peng, M. P. Gutenberg, and S. Sachdeva. *Maximum Flow and Minimum-Cost Flow in Almost-Linear Time*. 2022. URL: https://arxiv.org/abs/2203.00671.

[Chu14] S. Chubanov. "A polynomial algorithm for linear optimization which is strongly polynomial under certain conditions on optimal solutions". http://www.optimization-online.org/DB_HTML/2014/12/4710.html. 2014.

[CKL13] H. Y. Cheung, T. C. Kwok, and L. C. Lau. "Fast Matrix Rank Algorithms and Applications". *J. ACM* 60:5, 2013. ISSN: 0004-5411. URL: https://doi.org/10.1145/2528404.

[CLS19] M. B. Cohen, Y. T. Lee, and Z. Song. "Solving linear programs in the current matrix multiplication time". In: *Proceedings of the 51st Annual ACM Symposium on Theory of Computing (STOC)*. 2019, pp. 938–942.

[CM94] E. Cohen and N. Megiddo. "Improved Algorithms for Linear Inequalities With Two Variables per Inequality". *SIAM J. Comput.* 23:6, 1994, pp. 1313–1347.

[DEFM14] M. Di Summa, F. Eisenbrand, Y. Faenza, and C. Moldenhauer. "On largest volume simplices and sub-determinants". In: *Proceedings of the 26th annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 2014, pp. 315–323.

[DF94] M. Dyer and A. Frieze. "Random walks, totally unimodular matrices, and a randomised dual simplex algorithm". *Mathematical Programming* 64:1, 1994, pp. 1–16.

[DH16] D. Dadush and N. Hähnle. "On the shadow simplex method for curved polyhedra". *Discrete & Computational Geometry* 56:4, 2016, pp. 882–909.

[DH20] D. Dadush and S. Huiberts. "Smoothed Analysis of the Simplex Method". In: *Beyond the Worst-Case Analysis of Algorithms*. Ed. by T. Roughgarden. Cambridge University Press, Cambridge, 2020.

[DHL15] J. A. De Loera, R. Hemmecke, and J. Lee. "On Augmentation Algorithms for Linear and Integer-Linear Programming: From Edmonds–Karp to Bland and Beyond". *SIAM Journal on Optimization* 25:4, 2015, pp. 2494–2511.

[DHNV20] D. Dadush, S. Huiberts, B. Natura, and L. 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)*. 2020, pp. 761–774.

[Dik67] I. Dikin. "Iterative solution of problems of linear and quadratic programming". *Doklady Akademii Nauk* 174:4, 1967, pp. 747–748.

[Din67] W. Dinkelbach. "On Nonlinear Fractional Programming". *Management Science* 13:7, 1967, pp. 492–498.

[Din70] E. A. Dinic. "Algorithm for solution of a problem of maximum flow in networks with power estimation". In: *Soviet Math. Doklady*. Vol. 11. 1970, pp. 1277–1280.

[DKNV21] D. Dadush, Z. K. Koh, B. Natura, and L. A. Végh. "An Accelerated Newton-Dinkelbach Method and Its Application to Two Variables per Inequality Systems". In: *29th Annual European Symposium on Algorithms (ESA 2021)*. Ed. by P. Mutzel, R. Pagh, and G. Herman. Vol. 204. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany, 2021, 36:1–36:15. ISBN: 978-3-95977-204-4. URL: https://drops.dagstuhl.de/opus/volltexte/2021/14617.

[DKNV22] D. Dadush, Z. K. Koh, B. Natura, and L. A. Végh. "On Circuit Diameter Bounds via Circuit Imbalances". In: *Integer Programming and Combinatorial Optimization*. Springer International Publishing, 2022, pp. 140–153. URL: https://doi.org/10.1007/978-3-031-06901-7_11.

[DLY21] S. Dong, Y. T. Lee, and G. Ye. "A Nearly-Linear Time Algorithm for Linear Programs with Small Treewidth: A Multiscale Representation of Robust Central Path". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. Association for Computing Machinery, New York, NY, USA, 2021, pp. 1784–1797. ISBN: 9781450380539. URL: https://doi.org/10.1145/3406325.3451056.

[DNT08] A. Deza, E. Nematollahi, and T. Terlaky. "How good are interior point methods? Klee–Minty cubes tighten iteration-complexity bounds". *Math. Program.* 113:1, 2008, pp. 1–14. URL: https://doi.org/10.1007/s10107-006-0044-x.

[DNV20] D. Dadush, B. Natura, and L. A. Végh. "Revisiting Tardos's framework for linear programming: Faster exact solutions using approximate solvers". In: *Proceedings of the 61st Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2020, pp. 931–942.

[DS08] S. I. Daitch and D. A. Spielman. "Faster approximate lossy generalized flow via interior point algorithms". In: *Proceedings of the 40th annual ACM symposium on Theory of Computing*. 2008, pp. 451–460.

[DVZ] D. Dadush, L. A. Végh, and G. Zambelli. "On finding exact solutions of linear programs in the oracle model". In: *Proceedings of the 2022 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 2700–2722. URL: https://epubs.siam.org/doi/abs/10.1137/1.9781611977073.106.

[DVZ18] D. Dadush, L. A. Végh, and G. Zambelli. "Geometric rescaling algorithms for submodular function minimization". In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2018, pp. 832–848.

[EHK18] F. Eisenbrand, C. Hunkenschröder, and K.-M. Klein. "Faster Algorithms for Integer Programs with Block Structure". In: *Proceedings of the 45th International Colloquium on Automata, Languages, and Programming (ICALP 2018)*. 2018, 49:1–13.

[Eis+19] F. Eisenbrand, C. Hunkenschröder, K. Klein, M. Koutecký, A. Levin, and S. Onn. "An Algorithmic Theory of Integer Programming". *CoRR* abs/1904.01361, 2019. URL: http://arxiv.org/abs/1904.01361.

[EJ70] J. Edmonds and E. L. Johnson. "Matching: A well-solved class of integer linear programs". In: *Combinatorial Structures and Their Applications*. Gordon and Breach, 1970, pp. 89–92.

[EK72] J. Edmonds and R. M. Karp. "Theoretical improvements in algorithmic efficiency for network flow problems". *Journal of the ACM (JACM)* 19:2, 1972, pp. 248–264.

[ENV22] F. Ekbatani, B. Natura, and L. A. Végh. "Circuit Imbalance Measures and Linear Programming". In: *Surveys in Combinatorics 2022*. London Mathematical Society Lecture Note Series. Cambridge University Press, 2022, pp. 64–114.

[ERW89] H. Edelsbrunner, G. Rote, and E. Welzl. "Testing the Necklace Condition for Shortest Tours and Optimal Factors in the Plane". *Theor. Comput. Sci.* 66:2, 1989, pp. 157–180.

[EV17] F. Eisenbrand and S. Vempala. "Geometric random edge". *Mathematical Programming* 164:1-2, 2017, pp. 325–339.

[FH14] E. A. Feinberg and J. Huang. "The value iteration algorithm is not strongly polynomial for discounted dynamic programming". *Oper. Res. Lett.* 42:2, 2014, pp. 130–131.

[Fra11] A. Frank. *Connections in Combinatorial Optimization*. Oxford Lecture Series in Mathematics and its Applications 38. Oxford University Press, 2011.

[FT87] A. Frank and É. Tardos. "An application of simultaneous Diophantine approximation in combinatorial optimization". *Combinatorica* 7:1, 1987, pp. 49–65.

[Ful68] D. Fulkerson. "Networks, frames, blocking systems". *Mathematics of the Decision Sciences, Part I, Lectures in Applied Mathematics* 2, 1968, pp. 303–334.

[GD21] J. B. Gauthier and J. Desrosiers. "The Minimum Mean Cycle-Canceling Algorithm for Linear Programs". *European Journal of Operational Research*, 2021. (in press).

[GGJ17] M. X. Goemans, S. Gupta, and P. Jaillet. "Discrete Newton's algorithm for parametric submodular function minimization". In: *Proceedings of the 19th International Conference on Integer Programming and Combinatorial Optimization*. Waterloo, ON, Canada, 2017, pp. 212–227.

[GHR95] O. Güler, A. J. Hoffman, and U. G. Rothblum. "Approximations to solutions to systems of linear inequalities". *SIAM Journal on Matrix Analysis and Applications* 16:2, 1995, pp. 688–696.

[GKS95] J. W. Grossman, D. M. Kulkarni, and I. E. Schochetman. "On the Minors of an Incidence Matrix and Its Smith Normal Form". en. *Linear Algebra and its Applications* 218, 1995, pp. 213–224. ISSN: 0024-3795.

[GL97] C. C. Gonzaga and H. J. Lara. "A note on properties of condition numbers". *Linear Algebra and its Applications* 261:1, 1997, pp. 269–273. ISSN: 0024-3795.

[GLP22] Y. Gao, Y. P. Liu, and R. Peng. "Fully dynamic electrical flows: Sparse maxflow faster than Goldberg–Rao". In: *62nd Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2022, pp. 516–527.

[GLS12] M. Grötschel, L. Lovász, and A. Schrijver. *Geometric algorithms and combinatorial optimization*. Vol. 2. Springer Science & Business Media, 2012.

[Gof80] J.-L. Goffin. "The relaxation method for solving systems of linear inequalities". *Mathematics of Operations Research* 5:3, 1980, pp. 388–414.

[Gon92] C. C. Gonzaga. "Path-following methods for linear programming". *SIAM review* 34:2, 1992, pp. 167–224.

[Gri72] R. C. Grinold. "The payment scheduling problem". *Naval Research Logistics Quarterly* 19, 1972, pp. 123–136.

[GS55] S. Gass and T. Saaty. "The computational algorithm for the parametric objective function". *Naval Res. Logist. Quart.* 2, 1955, pp. 39–45. ISSN: 0028-1441. URL: `http://dx.doi.org/10.1002/nav.3800020106`.

[GS86] A. M. Gerards and A. Schrijver. "Matrices with the Edmonds–Johnson property". *Combinatorica* 6:4, 1986, pp. 365–379.

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

[GWXY21] D. Ge, C. Wang, Z. Xiong, and Y. Ye. *From an Interior Point to a Corner Point: Smart Crossover*. 2021. URL: https://arxiv.org/abs/2102.09420.

[Hel57] I. Heller. "On linear systems with integral valued solutions." *Pacific Journal of Mathematics* 7:3, 1957, pp. 1351–1364.

[HK56] A. Hoffman and J. Kruskal. "Integral Boundary Points of Convex Polyhedra". In: *Linear Inequalities and Related Systems*. Princeton University Press, 1956, pp. 223–246.

[HKZ14] T. D. Hansen, H. Kaplan, and U. Zwick. "Dantzig's pivoting rule for shortest paths, deterministic MDPs, and minimum cost to time ratio cycles". In: *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete Algorithms*. Portland, OR, USA, 2014, pp. 847–860.

[HMNT93] D. S. Hochbaum, N. Megiddo, J. S. Naor, and A. Tamir. "Tight bounds and 2-approximation algorithms for integer programs with two variables per inequality". *Mathematical Programming* 62:1, 1993, pp. 69–83.

[HN94] D. S. Hochbaum and J. Naor. "Simple and Fast Algorithms for Linear and Integer Programs With Two Variables per Inequality". *SIAM J. Comput.* 23:6, 1994, pp. 1179–1192.

[Hof52] A. J. Hoffman. "On approximate solutions of systems of linear inequalities". *Journal of Research of the National Bureau of Standards* 49:4, 1952, pp. 263–265.

[HT02] J. C. Ho and L. Tunçel. "Reconciliation of Various Complexity and Condition Measures for Linear Programming Problems and a Generalization of Tardos' Theorem". In: *Foundations of Computational Mathematics*. World Scientific, 2002, pp. 93–147.

[HT56] I. Heller and C. Tompkins. "An extension of a theorem of Dantzig's". *Linear inequalities and related systems* 38, 1956, pp. 247–254.

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

[IMH82] O. H. Ibarra, S. Moran, and R. Hui. "A generalization of the fast LUP matrix decomposition algorithm and applications". *Journal of Algorithms* 3:1, 1982, pp. 45–56. URL: https://doi.org/10.1016/0196-6774(82)90007-4.

[JLSW20] H. Jiang, Y. T. Lee, Z. Song, and S. C.-w. Wong. "An Improved Cutting Plane Method for Convex Optimization, Convex-Concave Games, and Its Applications". In: *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*. Association for Computing Machinery, New York, NY, USA, 2020, pp. 944–953. ISBN: 9781450369794. URL: https://doi.org/10.1145/3357713.3384284.

[JSWZ21] S. Jiang, Z. Song, O. Weinstein, and H. Zhang. "A Faster Algorithm for Solving General LPs". In: *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*. STOC 2021. Association for Computing Machinery, Virtual, Italy, 2021, pp. 823–832. ISBN: 9781450380539. URL: https://doi.org/10.1145/3406325.3451058.

[Kal92] G. Kalai. "A subexponential randomized simplex algorithm". In: *Proceedings of the 24th annual ACM Symposium on Theory of Computing*. 1992, pp. 475–482.

[Kar84] N. Karmarkar. "A new polynomial-time algorithm for linear programming". In: *Proceedings of the 16th Annual ACM Symposium on Theory of Computing (STOC)*. 1984, pp. 302–311.

[Kha79] L. G. Khachiyan. "A polynomial algorithm in linear programming". In: *Doklady Academii Nauk SSSR*. Vol. 244. 1979, pp. 1093–1096.

[Kha95a] L. Khachiyan. "On the Complexity of Approximating Extremal Determinants in Matrices". *J. Complex.* 11:1, 1995, pp. 138–153. ISSN: 0885-064X. URL: `http://dx.doi.org/10.1006/jcom.1995.1005`.

[Kha95b] L. Khachiyan. "On the complexity of approximating extremal determinants in matrices". *Journal of Complexity* 11:1, 1995, pp. 138–153.

[KK92] G. Kalai and D. J. Kleitman. "A quasi-polynomial bound for the diameter of graphs of polyhedra". *Bulletin of the American Mathematical Society* 26:2, 1992, pp. 315–316.

[KM13] T. Kitahara and S. Mizuno. "A bound for the number of different basic solutions generated by the simplex method". *Mathematical Programming* 137:1-2, 2013, pp. 579–586.

[KM72] V. Klee and G. J. Minty. "How good is the simplex algorithm?" In: *Inequalities, III (Proc. Third Sympos., Univ. California, Los Angeles, Calif., 1969; dedicated to the memory of Theodore S. Motzkin)*. Academic Press, New York, 1972, pp. 159–175.

[KM97] A. V. Karzanov and S. T. McCormick. "Polynomial methods for separable convex optimization in unimodular linear spaces with applications". *SIAM Journal on Computing* 26:4, 1997, pp. 1245–1275.

[Knu85] D. E. Knuth. "Semi-optimal bases for linear dependencies". *Linear and Multilinear Algebra* 17:1, 1985, pp. 1–4.

[KOT13] S. Kakihara, A. O. Ohara, and T. Tsuchiya. "Information geometry and interior-point algorithms in semidefinite programs and symmetric cone programs". *Journal of Optimization Theory and Applications* 157, 2013, pp. 749–780.

[KOT14] S. Kakihara, A. O. Ohara, and T. Tsuchiya. "Curvature integrals and iteration complexities in SDP and symmetric cone programs". *Computational Optimization and Applications* 57, 2014, pp. 623–665.

[KPS19] S. Kafer, K. Pashkovich, and L. Sanità. "On the circuit diameter of some combinatorial polytopes". *SIAM Journal on Discrete Mathematics* 33:1, 2019, pp. 1–25.

[KT13] T. Kitahara and T. Tsuchiya. "A simple variant of the Mizuno–Todd–Ye predictor-corrector algorithm and its objective-function-free complexity". *SIAM Journal on Optimization* 23:3, 2013, pp. 1890–1903.

[KT95] D. Klatte and G. Thiere. "Error bounds for solutions of linear equations and inequalities". *Zeitschrift für Operations Research* 41:2, 1995, pp. 191–214.

[LDK95] M. L. Littman, T. L. Dean, and L. P. Kaelbling. "On the Complexity of Solving Markov Decision Problems". In: *Proceedings of the 11th Annual Conference on Uncertainty in Artificial Intelligence (UAI)*. Montreal, Quebec, Canada, 1995, pp. 394–402.

[Lee89] J. Lee. "Subspaces with well-scaled frames". *Linear Algebra and its Applications* 114, 1989, pp. 21–56.

[Lee90] J. Lee. "The incidence structure of subspaces with well-scaled frames". *Journal of Combinatorial Theory, Series B* 50:2, 1990, pp. 265–287.

[LHK12] J. D. Loera, R. Hemmecke, and M. Köppe. *Algebraic and Geometric Ideas in the Theory of Discrete Optimization*. Society for Industrial and Applied Mathematics, USA, 2012. ISBN: 1611972434.

[LKS19] J. A. D. Loera, S. Kafer, and L. Sanità. "Pivot Rules for Circuit-Augmentation Algorithms in Linear Optimization". *CoRR* abs/1909.12863, 2019. URL: `http://arxiv.org/abs/1909.12863`.

[LMT09] G. Lan, R. D. Monteiro, and T. Tsuchiya. "A polynomial predictor-corrector trust-region algorithm for linear programming". *SIAM Journal on Optimization* 19:4, 2009, pp. 1918–1946.

[LRT79] R. J. Lipton, D. J. Rose, and R. E. Tarjan. "Generalized Nested Dissection". *SIAM Journal on Numerical Analysis* 16:2, 1979, pp. 346–358. URL: `https://doi.org/10.1137/0716027`.

[LS14] Y. T. Lee and A. Sidford. "Path finding methods for linear programming: Solving linear programs in $\tilde{O}(\sqrt{\text{rank}})$ iterations and faster algorithms for maximum flow". In: *Proceedings of the 55th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. 2014, pp. 424–433.

[LS15] Y. T. Lee and A. Sidford. "Efficient Inverse Maintenance and Faster Algorithms for Linear Programming". In: *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. 2015, pp. 230–249.

[LS19] Y. T. Lee and A. Sidford. "Solving Linear Programs with $\tilde{O}(\sqrt{\text{rank}})$ Linear System Solves". *arXiv preprint 1910.08033*, 2019.

[LSW15] Y. T. Lee, A. Sidford, and S. C.-W. Wong. "A Faster Cutting Plane Method and its Implications for Combinatorial and Convex Optimization". In: *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*. 2015, pp. 1049–1065.

[LSZ19] Y. T. Lee, Z. Song, and Q. Zhang. "Solving Empirical Risk Minimization in the Current Matrix Multiplication Time". In: *COLT*. Vol. 99. Proceedings of Machine Learning Research. PMLR, Phoenix, USA, 2019, pp. 2140–2157. URL: `http://proceedings.mlr.press/v99/lee19a.html`.

[LY21] Y. T. Lee and M.-.-C. Yue. "Universal barrier is n-self-concordant". *Mathematics of Operations Research* 46:3, 2021, pp. 1129–1148.

[Mad02] O. Madani. "On Policy Iteration as a Newton's Method and Polynomial Policy Iteration Algorithms". In: *Proceedings of the 18th National Conference on Artificial Intelligence*. Edmonton, AB, Canada, 2002, pp. 273–278.

[Mad13] A. Madry. "Navigating Central Path with Electrical Flows: From Flows to Matchings, and Back". In: *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*. 2013, pp. 253–262.

[Meg79] N. Megiddo. "Combinatorial Optimization with Rational Objective Functions". *Math. Oper. Res.* 4:4, 1979, pp. 414–424.

[Meg82] N. Megiddo. "Is binary encoding appropriate for the problem-language relationship?" *Theoretical Computer Science* 19:3, 1982, pp. 337–341.

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

[Meh92] S. Mehrotra. "On the implementation of a primal-dual interior point method". *SIAM Journal on Optimization* 2:4, 1992, pp. 575–601.

[MMT98] N. Megiddo, S. Mizuno, and T. Tsuchiya. "A modified layered-step interior-point algorithm for linear programming". *Mathematical Programming* 82:3, 1998, pp. 339–355.

[MS00] S. T. McCormick and A. Shioura. "Minimum ratio canceling is oracle polynomial for linear programming, but not strongly polynomial, even for networks". *Operations Research Letters* 27:5, 2000, pp. 199–207.

[MT03] R. D. C. Monteiro and T. Tsuchiya. "A Variant of the Vavasis-Ye Layered-Step Interior-Point Algorithm for Linear Programming". *SIAM Journal on Optimization* 13:4, 2003, pp. 1054–1079.

[MT05] R. D. C. Monteiro and T. Tsuchiya. "A New Iteration-Complexity Bound for the MTY Predictor-Corrector Algorithm". *SIAM Journal on Optimization* 15:2, 2005, pp. 319–347.

[MT08] R. D. Monteiro and T. Tsuchiya. "A strong bound on the integral of the central path curvature and its relationship with the iteration-complexity of primal-dual path-following LP algorithms". *Mathematical Programming* 115:1, 2008, pp. 105–149.

[MTY93] S. Mizuno, M. Todd, and Y. Ye. "On Adaptive-Step Primal-Dual Interior-Point Algorithms for Linear Programming". *Mathematics of Operations Research - MOR* 18, 1993, pp. 964–981.

[Nag07] K. Nagano. "A strongly polynomial algorithm for line search in submodular polyhedra". *Discrete Optimization* 4:3-4, 2007, pp. 349–359.

[Nas00] J. C. Nash. "The (Dantzig) Simplex Method for Linear Programming". *Computing in Science & Engineering* 2:01, 2000, pp. 29–31. ISSN: 1558-366X.

[NN94] Y. Nesterov and A. Nemirovskii. *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

[OLe90] D. P. O'Leary. "On bounds for scaled projections and pseudoinverses". *Linear Algebra and its Applications* 132, 1990, pp. 115–117. URL: https://doi.org/10.1016/0024-3795(90)90056-i.

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

[OV20] N. Olver and L. A. Végh. "A Simpler and Faster Strongly Polynomial Algorithm for Generalized Flow Maximization". *J. ACM* 67:2, 2020. ISSN: 0004-5411. URL: https://doi.org/10.1145/3383454.

[PV21] R. Peng and S. Vempala. "Solving Sparse Linear Systems Faster than Matrix Multiplication". In: *Proceedings of the 2021 ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics, 2021, pp. 504–521. URL: https://doi.org/10.1137/1.9781611976465.31.

[PVZ20] J. Pena, J. C. Vera, and L. F. Zuluaga. "New characterizations of Hoffman constants for systems of linear constraints". *Mathematical Programming* 187, 2020, pp. 1–31.

[PY15] I. Post and Y. Ye. "The Simplex Method is Strongly Polynomial for Deterministic Markov Decision Processes". *Math. Oper. Res.* 40:4, 2015, pp. 859–868.

[Rad92] T. Radzik. "Newton's Method for Fractional Combinatorial Optimization". In: *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*. Pittsburgh, PA, USA, 1992, pp. 659–669.

[Rad98] T. Radzik. "Fractional Combinatorial Optimization". In: *Handbook of Combinatorial Optimization: Volume1–3*. Ed. by D.-Z. Du and P. M. Pardalos. Springer US, 1998, pp. 429–478.

[Ren88] J. Renegar. "A polynomial-time algorithm, based on Newton's method, for linear programming". *Mathematical Programming* 40:1-3, 1988, pp. 59–93.

[Ren94] J. Renegar. "Is it possible to know a problem instance is ill-posed?: some foundations for a general theory of condition numbers". *Journal of Complexity* 10:1, 1994, pp. 1–56.

[Ren95] J. Renegar. "Incorporating condition measures into the complexity theory of linear programming". *SIAM Journal on Optimization* 5:3, 1995, pp. 506–524.

[RG94] T. Radzik and A. V. Goldberg. "Tight bounds on the number of minimum-mean cycle cancellations and related results". *Algorithmica* 11:3, 1994, pp. 226–242.

[Roc69] R. T. Rockafellar. "The Elementary Vectors of a Subspace of R^N". In: *Combinatorial Mathematics and Its Applications: Proceedings North Carolina Conference, Chapel Hill, 1967*. The University of North Carolina Press, 1969, pp. 104–127.

[San12] F. Santos. "A counterexample to the Hirsch conjecture". *Annals of Mathematics*, 2012, pp. 383–412.

[Sch03] A. Schrijver. *Combinatorial Optimization – Polyhedra and Efficiency*. Springer, 2003.

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

[Sey80] P. Seymour. "Decomposition of regular matroids". *Journal of Combinatorial Theory, Series B* 28:3, 1980, pp. 305–359.

[Sey93] M. Seysen. "Simultaneous reduction of a lattice basis and its reciprocal basis". *Combinatorica* 13:3, 1993, pp. 363–376.

[Sho81] R. E. Shostak. "Deciding Linear Inequalities by Computing Loop Residues". *J. ACM* 28:4, 1981, pp. 769–779.

[SIM00] M. Shigeno, S. Iwata, and S. T. McCormick. "Relaxed most negative cycle and most positive cut canceling algorithms for minimum cost flow". *Mathematics of Operations Research* 25:1, 2000, pp. 76–104.

[Sma98] S. Smale. "Mathematical problems for the next century". *The Mathematical Intelligencer* 20, 1998, pp. 7–15.

[SSZ91] G. Sonnevend, J. Stoer, and G. Zhao. "On the complexity of following the central path of linear programs by linear extrapolation II". *Mathematical Programming* 52:1-3, 1991, pp. 527–553.

[ST04a] D. A. Spielman and S.-H. Teng. "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems". In: *Proceedings of the 36th Annual ACM Symposium on Theory of Computing (STOC)*. 2004.

[ST04b] D. A. Spielman and S.-H. Teng. "Smoothed analysis of algorithms: Why the simplex algorithm usually takes polynomial time". *Journal of the ACM (JACM)* 51:3, 2004, pp. 385–463.

[ST97] B. Sturmfels and R. R. Thomas. "Variation of cost functions in integer programming". *Mathematical Programming* 77:2, 1997, pp. 357–387.

[Ste89] G. Stewart. "On scaled projections and pseudoinverses". *Linear Algebra and its Applications* 112, 1989, pp. 189–193.

[Suk17] N. Sukegawa. "Improving bounds on the diameter of a polyhedron in high dimensions". *Discrete Mathematics* 340:9, 2017, pp. 2134–2142.

[SW99] A. S. Schulz and R. Weismantel. "An oracle-polynomial time augmentation algorithm for integer programming". In: *Proceedings of the 10th ACM-SIAM Symposium on Discrete Algorithms (SODA)*. 1999, pp. 967–968.

[Tar85] É. Tardos. "A strongly polynomial minimum cost circulation algorithm". *Combinatorica* 5:3, 1985, pp. 247–255.

[Tar86] É. Tardos. "A strongly polynomial algorithm to solve combinatorial linear programs". *Operations Research*, 1986, pp. 250–256.

[Tod90] M. J. Todd. "A Dantzig–Wolfe-Like Variant of Karmarkar's Interior-Point Linear Programming Algorithm". *Operations Research* 38:6, 1990, pp. 1006–1018. URL: `https://doi.org/10.1287/opre.38.6.1006`.

[Top78] D. M. Topkis. "Minimizing a submodular function on a lattice". *Operations research* 26:2, 1978, pp. 305–321.

[TTY01] M. J. Todd, L. Tunçel, and Y. Ye. "Characterizations, bounds, and probabilistic analysis of two complexity measures for linear programming problems". *Mathematical Programming* 90:1, 2001, pp. 59–69.

[Tun99] L. Tunçel. "Approximating the complexity measure of Vavasis-Ye algorithm is NP-hard". *Mathematical Programming* 86:1, 1999, pp. 219–223.

[Tut65] W. T. Tutte. "Lectures on matroids". *J. Research of the National Bureau of Standards (B)* 69, 1965, pp. 1–47.

[Vai89] P. M. Vaidya. "Speeding-up linear programming using fast matrix multiplication". In: *Proceedings of the 30th IEEE Annual Symposium on Foundations of Computer Science*. 1989, pp. 332–337.

[Vas12] V. Vassilevska Williams. "Multiplying matrices faster than Coppersmith-Winograd". In: *Proceedings of the 44th annual ACM symposium on Theory of computing*. 2012, pp. 887–898.

[Vav94] S. A. Vavasis. "Stable numerical algorithms for equilibrium systems". *SIAM Journal on Matrix Analysis and Applications* 15:4, 1994, pp. 1108–1131.

[Vég17] L. A. Végh. "A Strongly Polynomial Algorithm for Generalized Flow Maximization". *Mathematics of Operations Research* 42:2, 2017, pp. 179–211.

[VY95] S. A. Vavasis and Y. Ye. "Condition numbers for polyhedra with real number data". *Operations Research Letters* 17, 1995, pp. 209–214.

[VY96] S. A. Vavasis and Y. Ye. "A primal-dual interior point method whose running time depends only on the constraint matrix". *Mathematical Programming* 74:1, 1996, pp. 79–120.

[Wal89] C. Wallacher. "A generalization of the minimum-mean cycle selection rule in cycle canceling algorithms". unpublished manuscript, Institute für Angewandte Mathematik, Technische Universität Braunschweig. 1989.

[Way02] K. D. Wayne. "A polynomial combinatorial algorithm for generalized minimum cost flow". *Mathematics of Operations Research*, 2002, pp. 445–459.

[Wei74] A. Weintraub. "A primal algorithm to solve network flow problems with convex costs". *Management Science* 21:1, 1974, pp. 87–97.

[WYZ06] Q. Wang, X. Yang, and J. Zhang. "A Class of Inverse Dominant Problems under Weighted ℓ_∞ Norm and an Improved Complexity Bound for Radzik's Algorithm". *J. Global Optimization* 34:4, 2006, pp. 551–567.

[WZ99] C. Wallacher and U. T. Zimmermann. "A polynomial cycle canceling algorithm for submodular flows". *Mathematical programming* 86:1, 1999, pp. 1–15.

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

[Ye06] Y. Ye. "Improved Complexity Results on Solving Real-number Linear Feasibility Problems". *Mathematical Programming* 106:2, 2006, pp. 339–363. ISSN: 0025-5610.

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

[Ye97] Y. Ye. *Interior-Point Algorithms: Theory and Analysis*. John Wiley and Sons, New York, 1997.

[YTM94] Y. Ye, M. J. Todd, and S. Mizuno. "An $O(\sqrt{n}L)$-Iteration Homogeneous and Self-Dual Linear Programming Algorithm". *Mathematics of Operations Research* 19:1, 1994, pp. 53–67.

List of Tables

List of Figures

List of Algorithms

List of Oracles

List of Systems