2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)

Circulation Control for Faster Minimum Cost Flow in Unit-Capacity Graphs

Kyriakos Axiotis
MIT
Cambridge, MA, USA
kaxiotis@mit.edu

Aleksander Mądry
MIT
Cambridge, MA, USA
madry@mit.edu

Adrian Vladu
Boston University
Boston, MA
avladu@mit.edu

***Abstract*—We present an $m^{4/3+o(1)} \log W$-time algorithm for solving the minimum cost flow problem in graphs with unit capacity, where W is the maximum absolute value of any edge weight. For sparse graphs, this improves over the best known running time for this problem and, by well-known reductions, also implies improved running times for the shortest path problem with negative weights, minimum cost bipartite $\boldsymbol{b}$-matching when $\|\boldsymbol{b}\|_1 = O(m)$, and recovers the running time of the currently fastest algorithm for maximum flow in graphs with unit capacities (Liu-Sidford, 2020).**

Our algorithm relies on developing an interior point method–based framework which acts on the space of circulations in the underlying graph. From the combinatorial point of view, this framework can be viewed as iteratively improving the cost of a suboptimal solution by pushing flow around circulations. These circulations are derived by computing a regularized version of the standard Newton step, which is partially inspired by previous work on the unit-capacity maximum flow problem (Liu-Sidford, 2019), and subsequently refined based on the very recent progress on this problem (Liu-Sidford, 2020). The resulting step problem can then be computed efficiently using the recent work on ℓ_p-norm minimizing flows (Kyng-Peng-Sachdeva-Wang, 2019). We obtain our faster algorithm by combining this new step primitive with a customized preconditioning method, which aims to ensure that the graph on which these circulations are computed has sufficiently large conductance.

***Keywords*-minimum cost flow, shortest path, interior point method**

I. Introduction

Finding the least costly way to route a demand through a network is a fundamental algorithmic primitive. Within the context of algorithmic graph theory it is captured as the *minimum cost flow problem*, in which given a graph with costs on its arcs and a set of demands on its vertices, one needs to find a flow that routes the demand while minimizing its cost. This problem has received significant attention [1] and inspired the development of new algorithmic techniques. For example, Orlin's network simplex algorithm [2] offered an explanation of the excellent behavior that the simplex method exhibits in practice when applied to flow problems. More broadly, the recent progress on algorithms for the flow problems [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15] has been an instance of the general approach to graph algorithms that leverages the tools of continuous optimization, rather than classical combinatorial techniques. Also, there exist efficient reductions that enable us to leverage algorithms for the minimum cost flow problem to solve a host of other fundamental problems, including the maximum flow problem, the minimum cost bipartite matching problem, and the shortest path problem with negative weights.

A. Our Contributions

In this paper[1], we present an $m^{4/3+o(1)} \log W$-time algorithms for the minimum cost flow problem in graphs with unit capacities, where W denotes the bound on the magnitude of the arc costs. This improves upon the previously known $\widetilde{O}(m^{10/7} \log W)$ running time bound of Cohen et al. [11] and matches the running times of the recent algorithms due to Liu and Sidford [14], [15] for the unit capacity maximum flow problem.[2]

Similarly to most of the relevant prior work, our algorithm at its core relies on an interior point method, but the variant of the interior point method we design and employ is directly attuned to the combinatorial properties of the graph. In particular, in contrast to [11], we do not rely on a reduction to the bipartite perfect $\boldsymbol{b}$-matching problem (which requires a sophisticated analysis). Instead, our algorithm operates directly in the space of circulations of the original graph.

One can also draw an analogy between the network simplex method [2] and ours. The former navigated the corners of a feasible polytope and improved an existing suboptimal solution through pushing flow around cycles. In contrast, we iteratively improve our existing suboptimal solution by augmenting it with circulations, but navigate through the strict interior of the polytope, seeking to keep a specific condition called *centrality* satisfied. Also, while in the network simplex case, the key difficulty is in finding the right pivoting rule, our approach shifts the attention towards finding the right circulation to augment the flow with so as to maintain the centrality invariant.

[1] A full version of this paper is available as [16].

[2] The initial version of this paper obtained a running time of $m^{11/8+o(1)} \log W$, which matched the running time of the then-fastest unit-capacity maximum flow algorithm due to Liu and Sidford [14]. After this version was released [16], Liu and Sidford [15] developed an improved running time of $m^{4/3+o(1)}$ for the unit-capacity maximum flow problem. Their techniques turned out to be immediately adaptable to our minimum cost flow framework, and led to the current $m^{4/3+o(1)} \log W$ running time for the unit-capacity minimum cost flow problem.

2575-8454/20/$31.00 ©2020 IEEE
DOI 10.1109/FOCS46700.2020.00018

A key ingredient of our approach is a custom preconditioning method, which enables us to control the flows we use to update the solution in each iteration. We derive a new way to tie the conductance of the graph to a certain guarantee on the flows computed in the preconditioned graph. This allows us to perform a better, tighter analysis of the quality of the preconditioner we use.

On a more technical level, our work provides a number of insights into the underlying interior point method. In particular, in our $m^{11/8+o(1)} \log W$-time algorithm (that we develop first), the progress steps we perform in order to reduce the duality gap of our current solution are cast as a refinement procedure, which simply attempts to correct a residual. This procedure is very similar to iterative refinement—widely used in the more restricted case of minimizing convex quadratic functions [17], [18]. Also, in contrast to the classic approach for maintaining constraint feasibility during the interior point method update step—which relies on controlling the ℓ_2 norm of the relative updates to the slack variables—we want to perform steps for which it is only guaranteed that these relative updates are small in ℓ_∞ norm. To this end, we employ a custom residual correction procedure that works by re-weighting the capacity constraints. (It is worth noting that a similar procedure has been used in [14].[3])

This paves the way for the final algorithm that has the further improved running time of $m^{4/3+o(1)} \log W$. As a matter of fact, the key bottleneck to obtaining a faster algorithm using the above approach is the need to ensure that the residual error in the solution obtained after performing a step bounded in ℓ_∞ norm can be reduced to zero. This requires increasing the weights on the constraint barriers, and these weight increases are exactly what limits the exponent in the running time to $11/8$. The step problem we need to solve, however, is well conditioned within a local ℓ_∞ ball around the current iterate. Therefore, being able to certify that the point returned by solving the step problem optimally lies within this local ℓ_∞ ball, implies that we can efficiently find it using a direct optimization subroutine. This latter observation is the key insight in the very recent preprint by Liu-Sidford [15] that enables them to improve the running time for maximum flow in graphs with unit capacity. We employ this insight in our setting in order to obtain an improved running time for unit-capacity minimum cost flow as well.

Finally, in order to guarantee that the ℓ_∞ norm of each progress step is indeed as small as needed, we employ a convex optimization subroutine with mixed ℓ_2 and ℓ_p terms [19], instead of solving a linear system of equations in each update step as is typically done. (Such subroutine was similarly used by Liu and Sidford [14], [15], in a slightly different form.)

[3]While our analysis aims to enforce a small ℓ_2 norm of the residual error, [14] seek to control the ℓ_4 norm of the congestion vector. These techniques turn out to be largely equivalent.

B. Previous Work

Due to the size of the existing literature on the studied problems, we focus our discussion only on the works that are the most relevant to our results and refer the reader to [20] and Section 1.2 in [11] for a more detailed discussion.

In 2013, Mądry [5] developed an algorithm that produces an optimal solution to the unit capacity maximum flow problem in $\widetilde{O}(m^{10/7})$ time and thus improves over a long standing running time barrier of $\widetilde{O}(n^{3/2})$ in the case of sparse graphs. An important characteristic of this approach was the careful tracking of an electrical energy quantity which allowed to control the step size. The underlying approach was then simplified by providing a more direct correspondence between the update steps of the interior point method and computing augmenting paths via electrical flow computations [9]. This framework has been also extended to a more general setting of unit capacity minimum cost flow [11], achieving a running time of $\widetilde{O}(m^{10/7} \log W)$, where W upper bounds the largest cost of a graph edge in absolute value.

In a different context, motivated by new developments involving regression problems [21], [22], [23], Kyng et al. [19] studied the ℓ_p regression problems on graphs, obtaining an algorithm which runs in $m^{1+o(1)}$ time for a range of large values of p. This algorithm's running time was subsequently further improved by Adil and Sachdeva [24].

Liu and Sidford [14] have recently obtained an improved algorithm for the unit capacity maximum flow problem with a running time of $m^{11/8+o(1)}$. One of their key insights was that the work on ℓ_p-regression problems enables treating energy control as a self-contained problem in each iteration of the interior point method, rather than maintaining energy as a global potential over the whole course of the algorithm, which was the case in previous work. Then, in their recent follow-up work, Liu and Sidford [15] strengthen the step problem primitive by directly optimizing a regularized log barrier function as opposed to performing a sequence of regularized Newton step. This led to a running time of $m^{4/3+o(1)}$ for the unit capacity maximum flow problem.

C. Organization of the Paper

We begin with technical preliminaries in Section II. In Section III, we present our interior point framework specialized to minimum cost flow, and provide a basic analysis which yields an algorithm running in $\widetilde{O}(m^{3/2} \log W)$ time. We further refine our framework in Section IV, where we develop the key tools needed for our results, giving a faster, $m^{11/8+o(1)} \log W$-time algorithm for obtaining the solution to a slightly perturbed instance of the original minimum cost flow problem. Finally, in Section V, we demonstrate how to combine the framework developed in the previous sections with an insight from the recent work of Liu and Sidford [15] to achieve the final running time of $m^{4/3+o(1)} \log W$.

II. Preliminaries

In this section, we introduce some basic notation and definitions that we will need later.

A. Basic Notation

Vectors: We use $\mathbf{0}$ and $\mathbf{1}$ to represent the all-zeros and all-ones vectors, respectively. Given two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ of the same dimension, we use $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ to represent their inner product. We apply scalar operations to vectors with the interpretation that they are applied point-wise, for example $\boldsymbol{x}/\boldsymbol{y}$ represents the vector whose i^{th} entry is x_i/y_i. We use the inline notation $(\boldsymbol{x}; \boldsymbol{y})$ to represent the concatenation of vectors $\boldsymbol{x}$ and $\boldsymbol{y}$.

Norms: Given a vector $\boldsymbol{x} \in \mathbb{R}^n$ and a scalar $p \geq 1$, we write the ℓ_p norm of $\boldsymbol{x}$ as $\|\boldsymbol{x}\|_p = \left(\sum_{i=1}^n |x_i|^p\right)^{1/p}$. Using this definition we also obtain $\|\boldsymbol{x}\|_\infty = \max_{i=1}^n |x_i|$. Throughout this paper we will be working especially with the ℓ_1, ℓ_2 and ℓ_∞ norms.

Graphs: Given a graph $G = (V, E)$ and a vertex $v \in V$, we will write $e \sim v$ to denote the set of edges $e \in E$ that are incident to v in G, i.e. the set of edges that have v as an endpoint.

Asymptotic notation: Given a parameter m denoting the number of edges of a graph, we use $\widetilde{O}(c)$ to denote a quantity that is $O(c \log^k m)$ for some constant k.

B. Minimum Cost Flow

We denote by $G = (V, E, \boldsymbol{c})$ a directed graph with vertex set V, arc set E and cost vector $\boldsymbol{c} \in \mathbb{R}^{|E|}$. We denote by $m = |E|$ the number of arcs, and by $n = |V|$ the number of vertices in G. An arc e of G connects an ordered pair (u, v), where u is its *tail* and v is its *head*. The basic notion of this paper is the notion of a flow. Given a graph G we view a flow in G as a vector $\boldsymbol{f} \in \mathbb{R}^m$ that assigns a value to each arc of G. If this value is negative we interpret it as having a flow of $|f_e|$ flowing in the direction opposite to the arc orientation. This convention is especially useful when discussing flows in undirected graphs.

We will be working with flows in G that satisfy a certain *demand* $\boldsymbol{d} \in \mathbb{R}^n$ such that $\sum_u d_u = 0$. We say that a flow $\boldsymbol{f}$ satisfies or routes demand $\boldsymbol{d}$ if it satisfies the flow conservation constraints with respect to the demands. That is:

$$\sum_{e \in E^+(u)} f_e - \sum_{e \in E^-(u)} f_e = d_u, \quad \text{for all } u \in V\,. \tag{1}$$

Here, $E^+(u)$ and $E^-(u)$ are the sets of arcs of G that are entering u and leaving u, respectively. Intuitively, these constraints enforce that the net balance of the total in-flow into vertex u and the total out-flow leaving that vertex is equal to d_u. A flow for which the demand vector $\boldsymbol{d}$ is zero everywhere is called a *circulation*.

We say that a flow $\boldsymbol{f}$ is *feasible* (or that it respects capacities) in G if it obeys the capacity constraints:

$$0 \leq f_e \leq u_e, \quad \text{for all } e \in E\,, \tag{2}$$

where $\boldsymbol{u} \in \mathbb{R}^m$ is a vector of arc capacities.

The *unit capacity minimum cost flow problem* is to find a flow $\boldsymbol{f} \in \mathbb{R}^m$ that meets the *unit capacity constraints* $0 \leq f_e \leq 1$ for all $e \in E$ and routes the demand $\boldsymbol{d}$, while minimizing the *cost* $\sum\limits_{e \in E} c_e f_e$.

Cycle Basis: A set of circulations $\mathcal{C}$ in G is called a *cycle basis* if any circulation in G can be expressed as a linear combination of circulations in $\mathcal{C}$. If G is connected, the dimension of a cycle basis of G is $m - n + 1$.

III. Minimum Cost Flow by Circulation Improvement

In this section we present our (customized) interior point method–based framework for solving the minimum cost flow problem, setting the foundations for the faster algorithm of Section IV.

A. LP Formulation and Interior Point Method

We first cast the minimum cost flow problem as a linear program that we then proceed to solve using an interior point method.

LP formulation: It will be useful to consider the parametrization of a flow in terms of the circulation space of the graph. The goal of this re-parametrization is to initialize the interior point method with an initial flow $\boldsymbol{f}_0$ which routes the prescribed demand $\boldsymbol{d}$, then iteratively improve it by adding circulations to get a flow which routes the same demand $\boldsymbol{d}$ but has lower duality gap. It is noteworthy that the specific parametrization of the circulation space is irrelevant to the interior point method, due to its affine invariance. We will elaborate on this point later. For us it will be a useful tool for understanding the centrality condition arising from the interior point method and applying more aggressive progress steps.

Given the (connected) underlying graph $G = (V, E)$, let $\boldsymbol{C} \in \mathbb{R}^{m \times (m-n+1)}$ be a matrix whose columns encode the characteristic vectors of a basis for G's circulation space.

In order to construct such a matrix, we let $C_1, C_2, ..., C_{m-n+1}$ be an arbitrary cycle basis for G, where we ignore the arc orientations. An easy way to produce one is to consider a spanning tree $T \subseteq G$. For each arc $(u, v) \in E$ which is not in T, consider the unique path in T connecting v and u. The arcs on this path along with the arc (u, v) determine a cycle in the basis. More specifically, consider the set of arcs of G present in C_i, sorted according to the order in which they are visited along the cycle, starting with the off-tree arc (u, v), then continuing with those witnessed along the tree path from v

to u. If an arc $e \in E$ has the opposite orientation to the one corresponding to the traversal of the cycle, we represent it as $\bar{e}$, otherwise we write it just as e.

Now, letting C_i consist of a subset of arcs in E, each of which appears either with its original orientation e, or the opposite orientation $\bar{e}$, we write the i^{th} column of matrix $\boldsymbol{C}$ as follows.

$$\boldsymbol{C}_{e,i} = \begin{cases} 1\,, & \text{if } e \in C_i\,, \\ -1\,, & \text{if } \bar{e} \in C_i\,, \\ 0\,, & \text{otherwise.} \end{cases}$$

We can now use $\boldsymbol{C}$ to represent any circulation in G. Given any $\boldsymbol{x} \in \mathbb{R}^{m-n+1}$ we have that $\boldsymbol{f} = \boldsymbol{C}\boldsymbol{x}$ is a circulation. Furthermore the sign of each coordinate f_e, $e = (u,v) \in E$, shows whether $\boldsymbol{f}_e$ is a flow that runs in the same direction as e or vice-versa, i.e. $f_e > 0$ if $\boldsymbol{f}$ carries flow from u to v, and similarly $f_e < 0$ if $\boldsymbol{f}$ carries flow from v to u. On the other hand, for any circulation $\boldsymbol{f} \in \mathbb{R}^m$ there exists an $\boldsymbol{x} \in \mathbb{R}^{m-n+1}$ such that $\boldsymbol{f} = \boldsymbol{C}\boldsymbol{x}$ (or in other words the image of $\boldsymbol{C}$ is the space of circulations).

Now let $\boldsymbol{f}_0$ be a flow in G such that for each arc $e \in E$, $0 < (\boldsymbol{f}_0)_e < 1$, and $\boldsymbol{f}_0$ routes the demand $\boldsymbol{d}$. The minimum cost flow problem can be cast as the following linear program:

$$\begin{aligned} &\min \langle \boldsymbol{c}, \boldsymbol{C}\boldsymbol{x}\rangle \\ &\boldsymbol{0} \le \boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x} \le \boldsymbol{1}\,. \end{aligned} \tag{3}$$

We see that the objective value of this linear program differs by a term of $\langle \boldsymbol{c}, \boldsymbol{f}_0\rangle$ from the original objective. We did not include it here, since it is a constant. It is useful to also consider its dual:

$$\begin{aligned} &\max -\langle \boldsymbol{1} - \boldsymbol{f}_0, \boldsymbol{y}^+\rangle - \langle \boldsymbol{f}_0, \boldsymbol{y}^-\rangle \\ &\boldsymbol{C}^\top\left(\boldsymbol{y}^+ - \boldsymbol{y}^-\right) = -\boldsymbol{C}^\top c \\ &\boldsymbol{y}^+, \boldsymbol{y}^- \ge \boldsymbol{0}\,. \end{aligned} \tag{4}$$

The objective we are left to solve simply suggests that in order to find the minimum cost flow in the graph with unit capacities, we equivalently have to find the minimum cost circulation in the residual graph under shifted capacity constraints. This carries a significant similarity with the *network simplex* algorithm [2], which has been used in the past as a specialization of the simplex method to the minimum cost flow problem. It essentially consisted of maintaining a solution routing the prescribed demand $\boldsymbol{d}$, and iteratively improving it by pushing flow around a cycle, while satisfying capacity constraints. Rather than performing such updates, which always maintain a flow on the boundary of polytope corresponding to the set of feasible solutions, the interior point method maintains a more sophisticated condition on these intermediate solutions. Another similar approach can be found in [25], where updates are iteratively pushed around cycles in order to solve Laplacian linear systems.

Like these methods, our approach will be to repeatedly improve the cost of the solution by pushing augmenting circulations. Crucially, maintaining a solution centrality condition, stemming from interior point methods, will allow us to make significant progress during each augmentation step.

Barrier Formulation: In order to apply an interior point method on (3), we need to replace the feasibility constraints by a convex barrier function. We seek a nearly optimal solution, i.e. one that has small duality gap. The vanilla interior point method consists of iteratively finding the optimizer $\boldsymbol{x}_\mu$ for a family of functions parametrized by $\mu > 0$

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathbb{R}^{m-n+1}} F_\mu^{\boldsymbol{w}}(\boldsymbol{x}) = &\frac{1}{\mu}\cdot\langle \boldsymbol{c}, \boldsymbol{C}\boldsymbol{x}\rangle \\ &- \sum_{e\in E}\Big(w_e^+ \cdot \log(\boldsymbol{1} - \boldsymbol{f}_0 - \boldsymbol{C}\boldsymbol{x})_e \\ &+ w_e^- \cdot \log(\boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x})_e\Big)\,. \end{aligned} \tag{5}$$

where $w_e^+, w_e^- > 0$ are weights on the flow capacity constraints. In order to find the optimizer $\boldsymbol{x}_\mu$, one performs Newton method on $F_\mu^{\boldsymbol{w}}$, after warm starting with $\boldsymbol{x}_{\mu(1+\delta)}$ for some $\delta > 0$.

While classical methods maintain $\boldsymbol{w} = \boldsymbol{1}$ at all times, this extra parameter has been introduced in previous work in order to allow the method to make progress more aggressively. To simplify notation we define the slack vector $\boldsymbol{s} = (\boldsymbol{s}^+; \boldsymbol{s}^-)$ as

$$\boldsymbol{s}^+ = \boldsymbol{1} - \boldsymbol{f}_0 - \boldsymbol{C}\boldsymbol{x}\,, \tag{6}$$

$$\boldsymbol{s}^- = \boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x}\,, \tag{7}$$

representing the upper slack (i.e. the distance of the current flow $\boldsymbol{f} = \boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x}$ to the upper capacity constraint of $\boldsymbol{f} \le \boldsymbol{1}$) and the lower slack (i.e. the distance from the current flow to the lower capacity constraint $\boldsymbol{0} \le \boldsymbol{f}$). We will use the vector $\boldsymbol{w} = (\boldsymbol{w}^+; \boldsymbol{w}^-)$ to represent the weights for the two sets of barriers that we are using.

B. Optimality and Duality Gap

In order to describe the method and analyze it, it is important to understand the optimality conditions for $F_\mu^{\boldsymbol{w}}$. We say that a vector $\boldsymbol{x}$ which minimizes $F_\mu^{\boldsymbol{w}}$ is *central* (or satisfies *centrality*). This condition is described in the following lemma.

Lemma 3.1: The vector $\boldsymbol{x}$ is a minimizer for $F_\mu^{\boldsymbol{w}}$ if and only if

$$\boldsymbol{C}^\top\left(\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+} - \frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}\right) = -\frac{\boldsymbol{C}^\top\boldsymbol{c}}{\mu}\,. \tag{8}$$

Furthermore the vector $\boldsymbol{y} = (\boldsymbol{y}^+; \boldsymbol{y}^-)$ with $\boldsymbol{y}^+ = \mu\cdot\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+}$, $\boldsymbol{y}^- = \mu\cdot\frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}$ is a feasible dual vector, and the duality gap of the primal-dual solution $(\boldsymbol{x}, \boldsymbol{y})$ is exactly $\mu\|\boldsymbol{w}\|_1$.

Maintaining the centrality condition (8) will be the key challenge in obtaining a faster interior point method for this linear program. This emphasizes the fact that the aim of this method is to construct a feasible set of slacks $\boldsymbol{s}^+ = \mathbf{1} - \boldsymbol{f}_0 - \boldsymbol{C}\boldsymbol{x} > \mathbf{0}$ and $\boldsymbol{s}^- = \boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x} > \mathbf{0}$ such that $\boldsymbol{C}^\top \left(\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+} - \frac{\boldsymbol{w}^-}{\boldsymbol{s}^-} \right) = -\frac{\boldsymbol{C}^\top \boldsymbol{c}}{\mu}$ for a very small $\mu > 0$. It is important to note that even though the existence of such an $\boldsymbol{x}$ needs to be guaranteed, it is not necessary to explicitly maintain it. This will be apparent in the definition below.

Definition 3.2 (μ-central flow): Given weights $\boldsymbol{w} = (\boldsymbol{w}^+; \boldsymbol{w}^-) \in \mathbb{R}^{2m}_{>0}$, a flow $\mathbf{0} < \boldsymbol{f} < \mathbf{1}$ is called μ-*central* with respect to $\boldsymbol{w}$ if for some cycle basis matrix $\boldsymbol{C} \in \mathbb{R}^{m \times (m-n+1)}$,

$$\boldsymbol{C}^\top \left(\frac{\boldsymbol{w}^+}{\mathbf{1} - \boldsymbol{f}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}} \right) = -\frac{\boldsymbol{C}^\top \boldsymbol{c}}{\mu} \tag{9}$$

for some $\mu > 0$. We will call the parameter μ the *centrality* of $\boldsymbol{f}$ with respect to $\boldsymbol{w}$.
It should be noted that the precise choice of cycle basis $\boldsymbol{C}$ in the above definition is irrelevant, as the property is invariant under the choice of cycle basis.

C. Initialization

The initialization procedure description and analysis is standard and thus deferred to the full version of the paper. From now on we can assume that we have a graph G together with a μ-central flow with $\mu \le 2\|\boldsymbol{c}\|_2$.

D. Vanilla Interior Point Method

The steps mentioned enable us to recover the classical $\widetilde{O}\left(m^{1/2}\right)$ iteration bound. This is shown in the following lemma.

Lemma 3.3: Given a μ^0-central flow with respect to weights $\mathbf{1}$ and $\mu^0 = m^{O(1)}$, we can obtain a minimum cost flow solution with duality gap at most $\varepsilon = 1/m^{O(1)}$ using $\widetilde{O}\left(m^{1/2}\right)$ calls to AUGMENT.

As previously discussed, each iteration of the interior point point method can be implemented in $\widetilde{O}(m)$ time using fast Laplacian solvers. This carries over to an algorithm with a total running time of $\widetilde{O}(m^{3/2} \log W)$, matching that of previous classical algorithms.

IV. A Faster Algorithm for Minimum Cost Flow

Our improved algorithm will be based on the interior point method framework that was developed in Section III. The main bottleneck for the running time of that algorithm stems from the fact that the augmenting circulation we compute might not allow us to decrease the duality gap by more than a factor of $1 + 1/\Omega(\sqrt{m})$, as otherwise it is generally impossible to guarantee that the circulation will never congest some edges by more than the available capacity. Hence the iteration bound of $\widetilde{O}(m^{1/2})$, common to standard interior point methods.

We alleviate this difficulty by adding an ℓ_p regularization term in the augmenting flow objective, similarly to [14]. In [14], the authors follow the idea of [9] by computing augmenting s-t flows. A crucial ingredient is the fact that the congestion of these resulting augmenting flows is then immediately bounded by using a result from [9] which states that as long as there is enough s-t residual capacity, these flows come together with an electrical potential embedding, where no edge is too stretched.

However, this property is specific to the s-t maximum flow problem. To apply a similar argument for the minimum cost flow problem, one would need to guarantee that *all* cuts of the graph have sufficient residual capacity, which is not automatically enforced as in the case of s-t max flow. In order to enforce this cut property, we further regularize our objective in a different way. We do this by temporarily superimposing a star on top of our graph, thus obtaining an augmented graph. This transformation improves the conductance properties of the graph, ensuring that there is enough residual capacity in all cuts of the graph.

In Section IV-A, we describe the regularized step problem and outline the guarantees of the solution. In particular, the bias introduced by the regularizers implies that the augmenting flow is not a circulation anymore, and that we have introduced an additional residual for our solution in the barrier objective. We bound the magnitude of both of these perturbations and "undo" them at a later stage. Finally, we present our electrical stretch guarantee, which serves as the crucial ingredient in both preserving feasibility and maintaining centrality.

Even though the electrical stretch guarantee suffices for all purposes if the interior point method barrier terms are unweighted, as soon as weights come in the guarantee is affected. In particular, for any edge whose forward and backward weights are too imbalanced, the electrical stretch and congestion bounds that we obtain loosen. We deal with this issue by ensuring that the forward and backward weights for each edge are always relatively balanced, while introducing an additional demand perturbation.

In Section IV-B we provide the full view of the algorithm, which consists of combining all the ingredients of the previous sections, together with a residual routing scheme that includes both vanilla centering steps and constraint re-weighting to obtain an ℓ_∞-based interior point method rather than an ℓ_4-based one, as achieved by the vanilla algorithm.

As we mentioned, the solution obtained by the interior point method is for a minimum cost flow problem with a slightly perturbed demand. By an approach given in [11], one can turn such a solution into an optimal solution for the original demand, as long as the total demand perturbation is small.

A. Regularized Newton Step

The initialization procedure from Section III-C produces a solution with large duality gap, i.e. $O(\mu m)$ where $\mu \leq 2\|\boldsymbol{c}\|_2$. Our goal will be to reduce this by gradually lowering the parameter μ, while maintaining centrality. While in general to achieve this we require solving a sequence of linear systems of equations (as we saw in Section III-D), here we choose to solve a slightly perturbed linear system.

In order to do so, we modify the optimization problem from the previous section by adding two regularization terms, which will force the produced solution to be well-behaved. In addition, we allow the newly produced flow $\widetilde{\boldsymbol{f}}$, which we will use to update the current solution, to not be a circulation, as long as the demand it routes is small in ℓ_1 norm. While this breaks the structure of the problem we are solving, it only does so mildly – therefore once the interior point method has finished running we can repair the broken demand using combinatorial techniques.

Mixed Objective: To specify the regularized objective, we first augment the graph G with $O(m)$ extra edges, which are responsible for routing a subset of the flows that would otherwise force the output of the objective to be too degenerate.

Definition 4.1 (Weighted degree): Given a graph $G(V,E)$ and a weight vector $\boldsymbol{w} = (\boldsymbol{w}^+; \boldsymbol{w}^-) \in \mathbb{R}^{2m}$, the *weighted degree* of $v \in V$ in G with respect to $\boldsymbol{w}$ is defined as $d_v^{\boldsymbol{w}} = \sum_{e \sim v}(w_e^+ + w_e^-)$.

Definition 4.2: Given a graph $G = (V,E)$ we define the augmented graph $G_\star = (V \cup \{v_\star\}, E_\star)$, where $E_\star = E \cup E'$ and E' is obtained by constructing $\lceil d_v^{\boldsymbol{w}} \rceil$ parallel edges $(v, v_\star)$ for each $v \in V$.

Furthermore, if $\boldsymbol{C}$ is a cycle basis for G, we let $\boldsymbol{C}_\star$ be a cycle basis for $G_\star$ obtained by appending columns to $\boldsymbol{C}$, i.e.

$$\boldsymbol{C}_\star = \left[\begin{array}{cc} \boldsymbol{C} & \boldsymbol{P}_1 \\ \boldsymbol{0} & \boldsymbol{P}_2 \end{array}\right]. \tag{10}$$

We observe that $|E'| = \sum_{v \in V} \lceil d_v^{\boldsymbol{w}} \rceil \leq \sum_{v \in V} (d_v^{\boldsymbol{w}} + 1) \leq 3\|\boldsymbol{w}\|_1$. We can now write the regularized objective.

Definition 4.3: Given a vector $\boldsymbol{h}$, we define the regularized objective as

$$\max_{\widetilde{\boldsymbol{f}} = \boldsymbol{C}_\star \widetilde{\boldsymbol{x}}} \sum_{e \in E} h_e \cdot \widetilde{f}_e - \frac{1}{2}\sum_{e \in E} (\widetilde{f}_e)^2 \cdot \left(\frac{w_e^+}{(s_e^+)^2} + \frac{w_e^-}{(s_e^-)^2}\right) - \frac{R_\star}{2}\sum_{e \in E'} (\widetilde{f}_e)^2 - \frac{R_p}{p} \sum_{e \in E \cup E'} (\widetilde{f}_e)^p , \tag{11}$$

where $p > 2$ is an even positive integer, and $R_\star$, R_p are some appropriately chosen non-negative scalars.

While this objective might seem difficult to handle, the fact that we are solving a problem on graphs makes it feasible for our purposes. In particular, the works of [19], [24] show that this objective can be solved to high precision in time $O(m^{1+o(1)})$, whenever p is sufficiently large. The resulting error can be easily handled, but for simplicity purposes let us from now assume that we can solve (11) exactly.

Let us now understand the effect of the augmenting edges E'. Since they allow routing some of the flow through $v_\star$, if we look at the restriction of $\widetilde{\boldsymbol{f}}$ to the edges of G we see that it stops being a circulation. Let $\widetilde{\boldsymbol{d}}$ be the demand routed by the restriction of $\widetilde{\boldsymbol{f}}$ to G. We will see that $\widetilde{\boldsymbol{f}}$ satisfies optimality conditions for an objective similar to (11) among all flows that route the demand $\widetilde{\boldsymbol{d}}$ in G.

Before that, we give a useful lemma that, given a residual $-\boldsymbol{C}^\top \boldsymbol{h}$, can be used to certify an upper bound on the energy required to route it. We capture this via the following definition.

Definition 4.4: Given a vector $\boldsymbol{h}$, weights $\boldsymbol{w}$, and slacks $\boldsymbol{s}$, we define

$$\mathcal{E}^{\max}(\boldsymbol{h}, \boldsymbol{w}, \boldsymbol{s}) = \frac{1}{2}\sum_{e \in E} h_e^2 \cdot \left(\frac{w_e^+}{(s_e^+)^2} + \frac{w_e^-}{(s_e^-)^2}\right)^{-1}. \tag{12}$$

Lemma 4.5: Given weights $\boldsymbol{w}$, slacks $\boldsymbol{s}$, and a residual $-\boldsymbol{C}^\top \boldsymbol{h}$, we have that

$$\mathcal{E}_{\boldsymbol{w},\boldsymbol{s}}(\boldsymbol{h}) \leq \mathcal{E}^{\max}(\boldsymbol{h}, \boldsymbol{w}, \boldsymbol{s}).$$

Furthermore, if $\boldsymbol{h} = \delta\left(\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+} - \frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}\right)$, we have that

$$\mathcal{E}^{\max}(\boldsymbol{h}, \boldsymbol{w}, \boldsymbol{s}) \leq \frac{1}{2}\delta^2 \|\boldsymbol{w}\|_1 .$$

We are now ready to state the lemma that gives guarantees for the restriction of $\widetilde{f}$ to G.

Lemma 4.6 (Optimality in the non-augmented graph): Let $\widetilde{\boldsymbol{f}}_\star = \boldsymbol{C}_\star \widetilde{\boldsymbol{x}}_\star$ be the optimizer of the regularized objective from (11), and let $\widetilde{\boldsymbol{f}}$ be its restriction to the edges of G. Let $\widetilde{\boldsymbol{d}}$ be the demand routed by $\widetilde{\boldsymbol{f}}$ in G. Then $\widetilde{\boldsymbol{f}}$ optimizes the objective

$$\max_{\substack{\widetilde{\boldsymbol{f}}: \\ \widetilde{\boldsymbol{f}} \text{ routes } \widetilde{\boldsymbol{d}} \text{ in G}}} \left\langle \boldsymbol{h}, \widetilde{\boldsymbol{f}} \right\rangle - \frac{1}{2}\sum_{e \in E} (\widetilde{f}_e)^2 \cdot \left(\frac{w_e^+}{(s_e^+)^2} + \frac{w_e^-}{(s_e^-)^2}\right) - \frac{R_p}{p}\sum_{e \in E} (\widetilde{f}_e)^p . \tag{13}$$

Furthermore

$$\boldsymbol{C}^\top \left(\frac{\boldsymbol{w}^+}{(\boldsymbol{s}^+)^2} + \frac{\boldsymbol{w}^-}{(\boldsymbol{s}^-)^2}\right) \cdot \widetilde{\boldsymbol{f}} = \boldsymbol{C}^\top (\boldsymbol{h} + \Delta \boldsymbol{h}) , \tag{14}$$

where $\Delta \boldsymbol{h} = -R_p (\widetilde{\boldsymbol{f}})^{p-1}$, and

$$\|\widetilde{\boldsymbol{d}}\|_1 \leq \left(\frac{6\|\boldsymbol{w}\|_1 \cdot \mathcal{E}^{\max}(\boldsymbol{h}, \boldsymbol{w}, \boldsymbol{s})}{R_\star}\right)^{1/2}, \tag{15}$$

$$\|\widetilde{\boldsymbol{f}}_\star\|_p \leq \left(\frac{p \cdot \mathcal{E}^{\max}(\boldsymbol{h}, \boldsymbol{w}, \boldsymbol{s})}{R_p}\right)^{1/p}. \tag{16}$$

Finally, the energy required to route the perturbed residual can be bounded by the energy required to route the original residual:

$$\frac{1}{2}\sum_{e\in E}(\widetilde{f}_e)^2\left(\frac{w_e^+}{(s_e^+)^2}+\frac{w_e^-}{(s_e^-)^2}\right)\leq 4\cdot\mathcal{E}^{\max}(\boldsymbol{h},\boldsymbol{w},\boldsymbol{s})\,. \quad (17)$$

Finally, we present an important property of the solution of the regularized Newton step, which will be crucial for obtaining the final result.

Lemma 4.7: Let $\widetilde{\boldsymbol{f}}_\star$ be the solution of the regularized objective (11) and $\widetilde{\boldsymbol{f}}$ its restriction on G, and suppose that $\|\boldsymbol{w}\|_1\geq 3$. Then one has that over the edges $e\in E$:

$$\left|\left(\frac{w_e^+}{(s_e^+)^2}+\frac{w_e^-}{(s_e^-)^2}+R_p\cdot\widetilde{f}_e^{p-2}\right)\widetilde{f}_e-h_e\right|\leq\hat{\gamma}\,, \quad (18)$$

where

$$\hat{\gamma}=\left(R_\star+R_p\cdot\|\widetilde{\boldsymbol{f}}_\star\|_\infty^{p-2}\right)^{1/2}\cdot\left\|\frac{\boldsymbol{h}}{\sqrt{(\boldsymbol{w}^++\boldsymbol{w}^-)\left(\frac{\boldsymbol{w}^+}{(\boldsymbol{s}^+)^2}+\frac{\boldsymbol{w}^-}{(\boldsymbol{s}^-)^2}\right)}}\right\|_\infty\cdot 32\log\|\boldsymbol{w}\|_1\,.$$

Furthermore, this implies that

$$\left(\frac{w_e^+}{(s_e^+)^2}+\frac{w_e^-}{(s_e^-)^2}\right)\cdot\left|\widetilde{f}_e\right|\leq|h_e|+\hat{\gamma}\,. \quad (19)$$

B. Executing the Interior Point Method

Having defined the regularized objective, we now show how to execute the interior point method using the solution returned by a high precision solver. Since the solution to this objective will not exactly optimize the unregularized objective, we will have to do some slight manual adjustments.

In the vanilla interior point method analysis that we saw earlier, we witnessed a very stringent requirement on the condition that we are able to correct a residual. Namely, we required that the energy required to route it decreases in every iteration of the correction step, which was guaranteed by the fact that after performing the first correction step the upper bound on energy $\sum w_i\rho_i^4$ is at most a small constant (i.e. $1/4$).

This requirement is too strong since, as a matter of fact, the most important obstacle handled by interior point methods is preserving slack feasibility. In our specific context this means that we want to perform updates to the current flow without violating capacity constraints, which is guaranteed by a weaker ℓ_∞ bound, i.e. $\|\boldsymbol{\rho}\|_\infty\leq 1/2$. While this condition is sufficient to preserve feasibility, it is not clear that after performing the corresponding update to the flow, the energy required to route the residual will be small, so the resulting residual can be reduced to $\mathbf{0}$. Instead we can enforce this property by canceling the components of the gradient which cause this energy to be large.

Definition 4.8 (Perturbed residual correction): Consider a flow $\boldsymbol{f}$ with the corresponding slack vector $\boldsymbol{s}>\mathbf{0}$, weights $\boldsymbol{w}$ and parameter $\mu>0$, with a corresponding residual $\nabla F_\mu^{\boldsymbol{w}}(\boldsymbol{x})=-\boldsymbol{C}^\top\boldsymbol{h}$ where $\boldsymbol{h}=\delta\left(\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+}-\frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}\right)$ and $\delta\leq\|\boldsymbol{w}\|_1^{-1/4}/2$. The perturbed residual correction step is defined as an update to $\boldsymbol{f}$ via:

$$\boldsymbol{f}'=\boldsymbol{f}+\widetilde{\boldsymbol{f}}\,, \quad (20)$$

$$(\boldsymbol{s}^-)'=\boldsymbol{s}^-+\widetilde{\boldsymbol{f}}\,, \quad (21)$$

$$(\boldsymbol{s}^+)'=\boldsymbol{s}^+-\widetilde{\boldsymbol{f}}\,, \quad (22)$$

where $\widetilde{\boldsymbol{f}}$ is the solution to the linear system

$$\boldsymbol{\rho}^+=\frac{\widetilde{\boldsymbol{f}}}{\boldsymbol{s}^+}\,, \quad (23)$$

$$\boldsymbol{\rho}^-=\frac{-\widetilde{\boldsymbol{f}}}{\boldsymbol{s}^-}\,, \quad (24)$$

$$\boldsymbol{C}^\top\left(\frac{\boldsymbol{w}^+\boldsymbol{\rho}^+}{\boldsymbol{s}^+}-\frac{\boldsymbol{w}^-\boldsymbol{\rho}^-}{\boldsymbol{s}^-}\right)=\boldsymbol{C}^\top(\boldsymbol{h}+\Delta\boldsymbol{h})\,, \quad (25)$$

such that

$$\|\boldsymbol{\rho}\|_\infty\leq\frac{1}{2}\,, \quad (26)$$

for some perturbation $\Delta\boldsymbol{h}$, followed by the updates to the $\boldsymbol{w}$ vector via:

$$(w_e^+)'=\begin{cases}w_e^++\frac{(s_e^+)'}{(s_e^-)'}\cdot w_e^-(\rho_e^-)^2 & \text{if } |\rho_e^-|\geq C_\infty\,,\\ w_e^+ & \text{otherwise,}\end{cases} \quad (27)$$

and

$$(w_e^-)'=\begin{cases}w_e^-+\frac{(s_e^-)'}{(s_e^+)'}\cdot w_e^+(\rho_e^+)^2 & \text{if } |\rho_e^+|\geq C_\infty\,,\\ w_e^- & \text{otherwise,}\end{cases} \quad (28)$$

where

$$C_\infty=\frac{1}{2\delta\sqrt{2\,\|\boldsymbol{w}\|_1}}\,.$$

We will also use the following weight balancing procedure, in order to avoid extreme weight imbalances that might otherwise impair our argument.

Definition 4.9 (Weight balancing procedure): An edge $e\in E$ is called *balanced* if $\max\{w_e^+,w_e^-\}\leq\delta\,\|\boldsymbol{w}\|_1$ or $\min\{w_e^+,w_e^-\}\geq 96\cdot\delta^4\,\|\boldsymbol{w}\|_1^2$. Otherwise it is called *imbalanced*. Now,given a flow $\boldsymbol{f}$ that is μ-central with respect to weights $\boldsymbol{w}$ and with slacks $\boldsymbol{s}$, let $S\subseteq E$ be the set of edges that are not balanced. The weight balancing procedure consists of computing new weights $\boldsymbol{w}'$ such that

- For each $e\in S$: If $w_e^+\leq w_e^-$ then $w_e^{'+}=96\cdot\delta^4\,\|\boldsymbol{w}\|_1^2$, $w_e^{'-}=w_e^-$, while if $w_e^+>w_e^-$ then $w_e^{'+}=w_e^+$, $w_e^{'-}=96\cdot\delta^4\,\|\boldsymbol{w}\|_1^2$.
- For each $e\notin S$ we set $w_e^{'+}=w_e^+$, $w_e^{'-}=w_e^-$.

Additionally, we compute a flow $\boldsymbol{f}'$ with slacks $\boldsymbol{s}'>\mathbf{0}$ such that

$$\frac{\boldsymbol{w}'^+}{\boldsymbol{s}'^+}-\frac{\boldsymbol{w}'^-}{\boldsymbol{s}'^-}=\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+}-\frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}\,.$$

We are now ready to state the lemma quantifying the progress in each iteration.

Lemma 4.10 (Progress lemma): Given a central instance with parameter μ and weights $\boldsymbol{w}$, we can obtain a new central instance with parameter $\mu/(1+\delta)$ and weights $\boldsymbol{w}''' + \Delta\boldsymbol{w} \geq \boldsymbol{w}$ with

$$\delta \geq m^{-1/4}/10\,,$$

such that

$$\begin{aligned}
&\|\boldsymbol{w}''' - \boldsymbol{w}\|_1 \\
&\leq \|\Delta\boldsymbol{w}\|_1 \\
&+ \left(\delta^2\|\boldsymbol{w}+\Delta\boldsymbol{w}\|_1\right)^{5/2} \cdot 6\cdot 10^4 \cdot \log\|\boldsymbol{w}+\Delta\boldsymbol{w}\|_1 \\
&+ m^{-10} \\
&+ \left(10^6\cdot\delta^2\|\boldsymbol{w}+\Delta\boldsymbol{w}\|_1\cdot\log\|\boldsymbol{w}+\Delta\boldsymbol{w}\|_1\right)^2 \\
&\cdot p\cdot\|\boldsymbol{w}+\Delta\boldsymbol{w}\|_1^{1/p}\,.
\end{aligned}$$

where $\Delta\boldsymbol{w} \geq \boldsymbol{0}$ is the weight increase caused by applying the procedure described in Definition 4.9 on weights $\boldsymbol{w}$. Furthermore, the demand perturbation is $\widetilde{\boldsymbol{d}} + \Delta\widetilde{\boldsymbol{d}}$, where

$$\|\widetilde{\boldsymbol{d}}\|_1 \leq 1\,.$$

and $\Delta\widetilde{\boldsymbol{d}}$ is the demand perturbation caused by applying the procedure described in Definition 4.9 on weights $\boldsymbol{w}$.

Lemma 4.10 is the main workhorse of the improved algorithm. It shows that we can make large progress within the interior point method, while paying for some demand perturbation and for some slight increase in $\|\boldsymbol{w}\|_1$. In order to guarantee sufficient progress, all we are left to do is to ensure that we can set an appropriate δ such that the sum of weights never increases beyond $O(m)$. This is a mere consequence of the result given above.

Lemma 4.11: Suppose we have a μ-central instance with weights $\boldsymbol{w} \geq \boldsymbol{1}$, where $\|\boldsymbol{w}\|_1 \leq 2m+1$ and $\mu = m^{O(1)}$. Let $\varepsilon = m^{-O(1)}$, and let $\delta = m^{-(3/8+o(1))}$. In $\widetilde{O}(\delta^{-1})$ iterations of the procedure described in Lemma 4.10 we obtain an instance with duality gap at most ε with a total demand perturbation of $\widetilde{O}(\delta^{-1})$.

Combining with the repairing procedure elaborated in [11], which can repair an excess demand $\widetilde{\boldsymbol{d}}$ in $\widetilde{O}\left(m\|\widetilde{\boldsymbol{d}}\|_1\right)$ time, we obtain the main theorem.

Theorem 4.12: Given a directed graph $G(V,E,\boldsymbol{c})$ with m arcs and n vertices, such that $\|\boldsymbol{c}\|_\infty \leq W$, and a demand vector $\boldsymbol{d} \in \mathbb{Z}^n$, in $m^{11/8+o(1)}\log W$ time we can obtain a flow $\boldsymbol{f}$ which routes $\boldsymbol{d}$ in G while satisfying the capacity constraints $\boldsymbol{0} \leq \boldsymbol{f} \leq \boldsymbol{1}$ and minimizing the cost $\sum_{e\in E} c_e f_e$, or certifies that no such flow exists.

V. Improving the Running Time to $m^{4/3+o(1)}\log W$

The running time of the algorithm we presented above hits a barrier at $m^{11/8+o(1)}\log W$. The key bottleneck there is the post-processing we do on the residual error of the (intermediate) solutions we obtain after performing each progress step. Indeed, while the length of our steps is dictated by the ℓ_∞-norm of our step size $\|\boldsymbol{\rho}\|_\infty$ (which is, in a sense optimal), it is unclear how to ensure that the energy required to route a flow that fixes the corresponding residual error is sufficiently small, without overly increasing the weights of the constraint barriers. In fact, the extent of weight perturbations necessary for this post-processing step are exactly what determines the $m^{11/8+o(1)}\log W$ running time. All the other weight perturbations, which are caused by the regularization terms, are much milder and would lead to the desired $m^{4/3+o(1)}\log W$ running time.

After the first version of this paper was posted [16], Liu and Sidford [15] published a preprint that obtains an improved running time for the unit-capacity maximum flow. The main technique introduced in that paper boils down exactly to avoiding the aforementioned bottleneck. Roughly, instead of advancing from a central point to the next one via a progress step followed by a sequence of residual correction steps, they instead directly solve the optimization problem which lands them at the next central point.

To do so, one must guarantee that this optimization problem is well-conditioned at all times, in the sense that the objective has a Hessian which always stays within a constant factor from the one at the origin. While such a Hessian stability condition is not true in general, Liu and Sidford [15] modify the logarithmic barriers they use by extending them with quadratics outside the region where they would be naturally well-behaved. The resulting new objective function can be efficiently optimized by slightly extending the mixed ℓ_2-ℓ_p solver of Kyng et al [19].

Incorporating this idea in our framework yields a the desired running time improvement of our unit-capacity minimum cost flow algorithm as well. Most of the details, particularly those involving preconditioning and weight perturbations carry over from the previous sections. In fact, the only change to the algorithm needed is to replace the regularized Newton step with solving a regularized problem which directly involves the logarithmic barrier.

Method Overview: Let us specify the ideal optimization problem which we would solve in order to advance along the central path. Suppose we have a μ-central instance i.e. we have a flow $\boldsymbol{f} = \boldsymbol{f}_0 + \boldsymbol{C}\boldsymbol{x}$ which satisfies

$$\boldsymbol{C}^\top\left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}_0-\boldsymbol{C}\boldsymbol{x}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}_0+\boldsymbol{C}\boldsymbol{x}}\right) = -\frac{\boldsymbol{C}^\top\boldsymbol{c}}{\mu}\,. \qquad (29)$$

as it optimizes the convex objective

$$\min_{\boldsymbol{x}} F_\mu^{\boldsymbol{w}}(\boldsymbol{x}) = \frac{1}{\mu}\langle\boldsymbol{c},\boldsymbol{C}\boldsymbol{x}\rangle - \sum_{e\in E}\Big(w_e^+\log(\boldsymbol{1}-\boldsymbol{f}_0-\boldsymbol{C}\boldsymbol{x})_e + w_e^-\log(\boldsymbol{f}_0+\boldsymbol{C}\boldsymbol{x})_e\Big)\,.$$

Our goal is to design an optimization procedure which enables us to augment $\boldsymbol{f}$ with a circulation $\boldsymbol{C}\widetilde{\boldsymbol{x}}$ in order to obtain an optimizer for

$$\begin{aligned}&\min_{\boldsymbol{x}'} F^{\boldsymbol{w}}_{\mu/(1+\delta)}(\boldsymbol{x}')\\&= \frac{1+\delta}{\mu}\langle \boldsymbol{c}, \boldsymbol{C}\boldsymbol{x}'\rangle\\&-\sum_{e\in E}\Big(w_e^+\log(\boldsymbol{1}-\boldsymbol{f}_0-\boldsymbol{C}\boldsymbol{x}')_e + w_e^-\log(\boldsymbol{f}_0+\boldsymbol{C}\boldsymbol{x}')_e\Big)\\&= \frac{1+\delta}{\mu}\langle \boldsymbol{c}, \boldsymbol{f}+\boldsymbol{C}\widetilde{\boldsymbol{x}}\rangle\\&-\sum_{e\in E}\Big(w_e^+\log(\boldsymbol{1}-\boldsymbol{f}-\boldsymbol{C}\widetilde{\boldsymbol{x}})_e + w_e^-\log(\boldsymbol{f}+\boldsymbol{C}\widetilde{\boldsymbol{x}})_e\Big)\,.\end{aligned}$$

We can equivalently rewrite this optimization procedure, after adding a constant term to it, as

$$\min_{\widetilde{\boldsymbol{x}}} \Psi^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}})\,,$$

where

$$\begin{aligned}\Psi^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}}) &:= F^{\boldsymbol{w}}_{\mu/(1+\delta)}(\boldsymbol{x}+\widetilde{\boldsymbol{x}}) - \frac{1+\delta}{\mu}\langle \boldsymbol{c},\boldsymbol{f}\rangle\\&+\sum_{e\in E}\left(w_e^+\log(1-f_e)+w_e^-\log f_e\right)\\&= \frac{1+\delta}{\mu}\langle \boldsymbol{c}, \boldsymbol{C}\widetilde{\boldsymbol{x}}\rangle\\&-\sum_{e\in E}\left(w_e^+\log\left(\boldsymbol{1}-\frac{\boldsymbol{C}\widetilde{\boldsymbol{x}}}{\boldsymbol{1}-\boldsymbol{f}}\right)_e\right.\\&\left.+w_e^-\log\left(\boldsymbol{1}+\frac{\boldsymbol{C}\widetilde{\boldsymbol{x}}}{\boldsymbol{f}}\right)_e\right)\,.\end{aligned}$$

Optimizing this function is hard to do in general. However, assuming its optimal augmenting circulation $\boldsymbol{C}\widetilde{\boldsymbol{x}}$ satisfies

$$\left\|\frac{\boldsymbol{C}\widetilde{\boldsymbol{x}}}{\min\{\boldsymbol{f},\boldsymbol{1}-\boldsymbol{f}\}}\right\|_\infty \leq \frac{1}{10}\,, \tag{30}$$

which in other words, says that augmenting $\boldsymbol{f}$ with $\boldsymbol{C}\widetilde{\boldsymbol{x}}$ will not come close to breaking the feasibility constraints, we can instead solve a well-conditioned objective obtained by replacing the log's with a better behaved function $\widetilde{\log}$ satisfying $\log(1+t) = \widetilde{\log}(1+t)$ whenever $|t| \leq 1/10$.

More precisely, we use the definition from [15] which we reproduce below:

- If $t\in[-\theta,\theta]$, then $\widetilde{\log}(1+t)=\log(1+t)$
- if $t>\theta$, then
$$\begin{aligned}\widetilde{\log}(1+t) = \log(1+\theta) &+ (t-\theta)\cdot\log'(1+\theta)\\&+(t-\theta)^2\cdot\frac{1}{2}\log''(1+\theta)\,,\end{aligned}$$
- if $t<-\theta$, then
$$\begin{aligned}\widetilde{\log}(1+t) = \log(1-\theta) &+ (t+\theta)\cdot\log'(1-\theta)\\&+(t+\theta)^2\cdot\frac{1}{2}\log''(1-\theta)\,.\end{aligned}$$

where we set $\theta = 1/10$. We can also easily verify that on the boundary of the interval $[-\theta,\theta]$ the first and second order derivatives exactly match those of $\log(1+t)$. The essential feature is that outside this range, the second derivative stays constant, whereas in the case of $\log(1+t)$ it changes very fast.

Using this, we can define the minimization problem

$$\min_{\widetilde{\boldsymbol{x}}} \widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}})\,,$$

where

$$\begin{aligned}\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}}) :&= \frac{1+\delta}{\mu}\langle \boldsymbol{c}, \boldsymbol{C}\widetilde{\boldsymbol{x}}\rangle\\&-\sum_{e\in E}\left(w_e^+\widetilde{\log}\left(\boldsymbol{1}-\frac{\boldsymbol{C}\widetilde{\boldsymbol{x}}}{\boldsymbol{1}-\boldsymbol{f}}\right)_e\right.\\&\left.+w_e^-\widetilde{\log}\left(\boldsymbol{1}+\frac{\boldsymbol{C}\widetilde{\boldsymbol{x}}}{\boldsymbol{f}}\right)_e\right)\,.\end{aligned} \tag{31}$$

Observation 5.1: If the minimizer $\widetilde{\boldsymbol{x}}$ of $\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}})$ satisfies the low-congestion condition from (30), then it also minimizes $\Psi^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}})$.

Due to the fact that the second order derivatives of $\widetilde{\log}$ are bounded, $\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}})$ is well-conditioned and, as a matter of fact can easily be minimized by using a small number of calls to a routine which minimizes quadratics over the set of circulations. As specified in [15] this can be easily done by using fast Laplacian system solvers. However, the main difficulty that arises is ensuring that the minimizer $\widetilde{\boldsymbol{x}}$ satisfies the confition from (30).

Enforcing this property is non-trivial, and requires regularizing the function $\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}$ in an identical manner to the way we did it in the analysis from Section IV. Most of the results we used there carry over, after performing some minor modifications in the analysis.

Regularizing the Objective: In order to enforce the required property, we add two regularization terms to our optimization step. Just like in Section IV-A we augment the graph G to $G_\star$, which has a cycle basis $\boldsymbol{C}_\star$, and in this graph we write down the equivalent concave maximization problem for (31), which we regularize with two extra terms. We slightly abuse notation by making $\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}$ act on an element in the circulation space of $G_\star$, with the meaning that the linear and logarithmic terms only act on edges in E:

$$\max_{\widetilde{\boldsymbol{f}}_\star=\boldsymbol{C}_\star\widetilde{\boldsymbol{x}}} -\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}}) - \frac{R_\star}{2}\sum_{e\in E'}(\widetilde{f}_\star)_e^2 - \frac{R_p}{p}\sum_{e\in E\cup E'}(\widetilde{f}_\star)_e^p\,. \tag{32}$$

Writing $\widetilde{\boldsymbol{f}}_\star = \widetilde{\boldsymbol{f}} + \widetilde{\boldsymbol{f}}'$ where $\widetilde{\boldsymbol{f}}$ is the restriction of $\widetilde{\boldsymbol{f}}_\star$ to the edges E of G, and $\widetilde{\boldsymbol{f}}'$ is the restriction to the augmenting edges E', and using the centrality condition (29), we can further write this as

$$\max_{\widetilde{\boldsymbol{f}}_\star = \boldsymbol{C}_\star \widetilde{\boldsymbol{x}}} (1+\delta)\left\langle \frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}}, \widetilde{\boldsymbol{f}} \right\rangle + \sum_{e\in E}\left(w_e^+ \widetilde{\log}\left(1 - \frac{\widetilde{f}_e}{1-f_e}\right) + w_e^- \widetilde{\log}\left(1+\frac{\widetilde{f}_e}{f_e}\right)\right) - \frac{R_\star}{2}\sum_{e\in E'} (\widetilde{f}'_e)^2 - \frac{R_p}{p}\sum_{e\in E\cup E'} (\widetilde{f}_\star)_e^p . \tag{33}$$

This optimization problem can be solved efficiently by slightly extending the results of [19]. The key reason is that all the terms except for the one involving p powers of the flow $\widetilde{f}$ have a second order derivative which is either constant, or which stays bounded within a small multiplicative factor from the one at 0. Similarly to before, the solver yields a high-accuracy, yet inexact solution. We can, however, assume we obtain an exact solution by performing minor perturbations to our problem.

We can now write the first order optimality condition for the objective in (32), thus providing an analogue of Lemma 4.6. Before doing so, we give the following helper lemma, which will be the main driver of the results in this section. It intuitively states that the optimal solution to (33) can be thought of as the solution to a regularized Newton step as the one in (11), but where the coefficients of the quadratic $\frac{1}{2}\sum_{e\in E} \widetilde{f}_e^2 \cdot \left(\frac{w_e^+}{(s_e^+)^2} + \frac{w_e^-}{(s_e^-)^2}\right)$ have been slightly perturbed by a small multiplicative constant.

Lemma 5.2 (Optimality with average Hessian): Let $\widetilde{\boldsymbol{f}}_\star = \boldsymbol{C}_\star \widetilde{\boldsymbol{x}}_\star$ be the optimizer of (33), and let $\widetilde{\boldsymbol{f}}_\star = \widetilde{\boldsymbol{f}} + \widetilde{\boldsymbol{f}}'$, where the two components are supported on E and E', respectively. Then there exists a vector $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^+;\boldsymbol{\alpha}^-)$, $(1+\theta)^{-2}\cdot\boldsymbol{1} \leq \boldsymbol{\alpha} \leq (1-\theta)^{-2}\cdot\boldsymbol{1}$, which can be explicitly computed, such that for any circulation $\boldsymbol{g}$ in $\boldsymbol{C}_\star$,

$$\left\langle \boldsymbol{g}, \begin{bmatrix} \boldsymbol{v} \\ -R_\star\cdot\widetilde{\boldsymbol{f}}' - R_p\cdot(\widetilde{\boldsymbol{f}}')^{p-1} \end{bmatrix} \right\rangle = 0\,,$$

where $\boldsymbol{v} = \delta\left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}}\right) - \widetilde{\boldsymbol{f}}\left(\frac{\boldsymbol{\alpha}^+\boldsymbol{w}^+}{(\boldsymbol{1}-\boldsymbol{f})^2} + \frac{\boldsymbol{\alpha}^-\boldsymbol{w}^-}{\boldsymbol{f}^2}\right) - R_p\cdot(\widetilde{\boldsymbol{f}})^{p-1}$.

Lemma 5.2 enables us to use an optimality condition very similar to the one we had before in Section IV. As a matter of fact, all the remaining statements are nothing but "robust" versions of those we previously used. Essential here are new versions of Lemma 4.6 and Lemma 4.7 which accommodate the extra multiplicative factors on resistances. Roughly, our goal is to provide upper bounds on $\|\widetilde{\boldsymbol{f}}\|_\infty$ and $\|\widetilde{\boldsymbol{f}}/\min\{\boldsymbol{1}-\boldsymbol{f},\boldsymbol{f}\}^2\|_\infty$, which together will imply that the condition from (30) is satisfied.

After proving that this is the case, we will show how to advance to the next point on the central path – the regularization terms on (33) will require us to increase the weights $\boldsymbol{w}$ in order to obtain optimality for the non-regularized objective i.e. $\nabla\widetilde{\Psi}^{\boldsymbol{w},\boldsymbol{f}}_{\mu/(1+\delta)}(\widetilde{\boldsymbol{x}}) = 0$. Nevertheless, this procedure is essentially identical to the one we used in the previous section.

A. Bounding Congestion

Here we show that the congestion condition from (30) is satisfied, and hence the minimizer of (33) also minimizes the expression after replacing $\widetilde{\log}$ with $\log$. For consistency we will use the slack notation

$$\boldsymbol{s}^- = \boldsymbol{f}\,,\quad \boldsymbol{s}^+ = \boldsymbol{1}-\boldsymbol{f}\,,$$

and use the shorthand notation for the residual

$$\boldsymbol{h} = \delta\left(\frac{\boldsymbol{w}^+}{\boldsymbol{s}^+} - \frac{\boldsymbol{w}^-}{\boldsymbol{s}^-}\right).$$

We first give a short lemma providing an optimality condition for the restriction of the flow $\widetilde{\boldsymbol{f}}_\star$ computed by (33) to the edges E of the original graph G.

Lemma 5.3 (Optimality in the non-augmented graph): Let $\widetilde{\boldsymbol{f}}_\star = \boldsymbol{C}_\star\widetilde{\boldsymbol{x}}_\star$ be the optimizer of (32) and let $\widetilde{\boldsymbol{f}}$ be its restriction to the edges of G. Let $\widetilde{\boldsymbol{d}}$ be the demand routed by $\widetilde{\boldsymbol{f}}$ in G. Then there exists a vector $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^+;\boldsymbol{\alpha}^-) \in \mathbb{R}^{2m}$, $\frac{1}{(1+\theta)^2}\cdot\boldsymbol{1} \leq \boldsymbol{\alpha} \leq \frac{1}{(1-\theta)^2}\cdot\boldsymbol{1}$, which can be explicitly computed, such that

$$\boldsymbol{C}^\top\cdot\left(\frac{\boldsymbol{\alpha}^+\boldsymbol{w}^+}{(\boldsymbol{s}^+)^2} + \frac{\boldsymbol{\alpha}^-\boldsymbol{w}^-}{(\boldsymbol{s}^-)^2}\right)\cdot\widetilde{\boldsymbol{f}} = \boldsymbol{C}^\top(\boldsymbol{h}+\Delta\boldsymbol{h}) \tag{34}$$

where $\Delta\boldsymbol{h} = -R_p(\widetilde{\boldsymbol{f}})^{p-1}$, and

$$\|\widetilde{\boldsymbol{d}}\|_1 \leq 3\left(\frac{\|\boldsymbol{w}\|_1\cdot\mathcal{E}^{\max}(\boldsymbol{h},\boldsymbol{w},\boldsymbol{s})}{R_\star}\right)^{1/2},$$

$$\|\widetilde{\boldsymbol{f}}_\star\|_p \leq \left(\frac{p\cdot\frac{3}{2}\mathcal{E}^{\max}(\boldsymbol{h},\boldsymbol{w},\boldsymbol{s})}{R_p}\right)^{1/p}.$$

Next we provide a guarantee enforced by the component of the regularizer involving augmenting edges.

Lemma 5.4: Let $\widetilde{\boldsymbol{f}}_\star$ be the solution of the regularized objective and $\widetilde{\boldsymbol{f}}$ its restriction on G, and suppose that $\|\boldsymbol{w}\|_1 \geq 3$. Then there exists a vector $\boldsymbol{\alpha} = (\boldsymbol{\alpha}^+;\boldsymbol{\alpha}^-) \in \mathbb{R}^{2m}$ such that for all edges $e\in E$:

$$\left|\left(\frac{\alpha_e^+ w_e^+}{(s_e^+)^2} + \frac{\alpha_e^- w_e^-}{(s_e^-)^2} + R_p\cdot\widetilde{f}_e^{p-2}\right)\widetilde{f}_e - h_e\right| \leq \hat{\gamma}\,, \tag{35}$$

where

$$\hat{\gamma} = \left(R_\star + R_p\cdot\|\widetilde{\boldsymbol{f}}_\star\|_\infty^{p-2}\right)^{1/2} \cdot\left\|\frac{\boldsymbol{h}}{\sqrt{(\boldsymbol{w}^+ + \boldsymbol{w}^-)\left(\frac{\boldsymbol{\alpha}^+\boldsymbol{w}^+}{(\boldsymbol{s}^+)^2} + \frac{\boldsymbol{\alpha}^-\boldsymbol{w}^-}{(\boldsymbol{s}^-)^2}\right)}}\right\|_\infty \cdot 32\log\|\boldsymbol{w}\|_1\,.$$

Furthermore, this implies that

$$\left(\frac{\alpha_e^+ w_e^+}{(s_e^+)^2} + \frac{\alpha_e^- w_e^-}{(s_e^-)^2}\right)\cdot\left|\widetilde{f}_e\right| \leq |h_e| + \hat{\gamma}\,. \tag{36}$$

Combining Lemmas 5.3 and 5.4 we can finally prove the main statement of this section.

In order to obtain the desired congestion bound, we set our regularization parameters to

$$p = \min\left\{k \in 2\mathbb{Z} : k \geq (\log m)^{1/3}\right\},$$
$$R_p = p \cdot \left(10^6 \cdot \delta^2 \|\boldsymbol{w}\|_1 \cdot \log \|\boldsymbol{w}\|_1\right)^{p+1},$$
$$R_\star = 3 \cdot \delta^2 \|\boldsymbol{w}\|_1^2 .$$

Lemma 5.5 (Congestion bound): Suppose that all edges $e \in E$ are *balanced*, per Definition 4.9, and $\delta > 10 \cdot \|\boldsymbol{w}\|_1^{-1/2}$. Then the restriction $\widetilde{\boldsymbol{f}}$ of the flow $\widetilde{\boldsymbol{f}}_\star$ computed via (33) satisfies the low-congestion condition (30).

B. Making Progress

Here we show how to use the flow obtained from optimizing (33) in order to achieve centrality for a new parameter $\mu/(1+\delta)$, at the expense of slightly increasing weights from $\boldsymbol{w}$ to some $\boldsymbol{w}'$.

Lemma 5.6 (Almost-centrality after executing step): Let $\widetilde{\boldsymbol{f}}_\star$ be the optimizer of (33) and let $\widetilde{\boldsymbol{f}}$ be its restriction to the edges of G. Then

$$\boldsymbol{C}^\top \left((1+\delta)\left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}}\right) - \left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}-\widetilde{\boldsymbol{f}}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}+\widetilde{\boldsymbol{f}}}\right)\right)$$
$$= \boldsymbol{C}^\top \cdot R_p \cdot (\widetilde{\boldsymbol{f}})^{p-1} .$$

As we can see, the regularization terms have two effects. One is that the update $\widetilde{\boldsymbol{f}}$ is not exactly a circulation, so this will account for some change in the routed demand. The other effect is that after augmenting the current flow $\boldsymbol{f}$ with $\widetilde{\boldsymbol{f}}$ we do not obtain a central solution, as shown in Lemma 5.6. We proceed to fix this manually by slightly increasing the weights $\boldsymbol{w}$, thus correcting the residual $\Delta \boldsymbol{h} = -R_p(\widetilde{\boldsymbol{f}})^{p-1}$ as we did in the previous section, which produces a central solution with a new set of weights $\boldsymbol{w}' \geq \boldsymbol{w}$ such that $\|\boldsymbol{w}' - \boldsymbol{w}\|_1 \leq R_p \cdot \sum_{e \in E} \left|\widetilde{f}_e\right|^{p-1}$.

We are now ready to characterize the amount of progress we make in a single iteration of the method previously described.

Lemma 5.7 (Progress lemma): Given a μ-central instance, i.e. a flow $\boldsymbol{f}$ and balanced weights $\boldsymbol{w}$ such that

$$\boldsymbol{C}^\top \left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}}\right) = -\frac{\boldsymbol{C}^\top \boldsymbol{c}}{\mu},$$

in the time require to solve (33) we can obtain a $\mu/(1+\delta)$-central instance, i.e. a flow $\boldsymbol{f}+\widetilde{\boldsymbol{f}}$ and weights $\boldsymbol{w}' \geq \boldsymbol{w}$, such that

$$\boldsymbol{C}^\top \left(\frac{\boldsymbol{w}^+}{\boldsymbol{1}-\boldsymbol{f}-\widetilde{\boldsymbol{f}}} - \frac{\boldsymbol{w}^-}{\boldsymbol{f}+\widetilde{\boldsymbol{f}}}\right) = -(1+\delta)\frac{\boldsymbol{C}^\top \boldsymbol{c}}{\mu},$$

where

$$\|\boldsymbol{w}' - \boldsymbol{w}\|_1 \leq p \cdot 10^{12} \cdot \delta^4 \|\boldsymbol{w}\|_1^{2+1/p} \cdot \log^2 \|\boldsymbol{w}\|_1 ,$$

and $\widetilde{\boldsymbol{f}}$ routes a demand $\widetilde{\boldsymbol{d}}$ such that

$$\|\widetilde{\boldsymbol{d}}\|_1 \leq 3/2 .$$

C. Wrapping Up

We can now give the main statement of this section, which follows from running the interior point method, based on the guarantee provided by Lemma 5.7.

Lemma 5.8: Suppose we have a μ-central instance with weights $\boldsymbol{w} \geq \boldsymbol{1}$, where $\|\boldsymbol{w}\|_1 \leq 2m+1$ and $\mu = m^{O(1)}$. Let $\varepsilon = m^{-O(1)}$, and let $\delta = m^{-(1/3+o(1))}$. In time dominated by $\widetilde{O}(\delta^{-1})$ iterations of the procedure described in Lemma 5.7 we obtain an instance with duality gap at most ε with a total demand perturbation of $\widetilde{O}(\delta^{-1})$.

This enables us to obtain a running time of $m^{4/3+o(1)}$ for minimum cost flow in unit-capacity graphs. The proof is identical to that of Theorem 4.12, we use scaling to obtain a logarithmic dependence in W, and resort to the fixing procedure from [11] to repair the demand perturbation. The time required to implement each iteration of the interior point method is dominated by the time required by one call to the solver of (33), which is $m^{1+o(1)}$ by our choice of parameters.

Theorem 5.9: Given a directed graph $G(V, E, \boldsymbol{c})$ with m arcs and n vertices, such that $\|\boldsymbol{c}\|_\infty \leq W$, and a demand vector $\boldsymbol{d} \in \mathbb{Z}^n$, in $m^{4/3+o(1)} \log W$ time we can obtain a flow $\boldsymbol{f}$ which routes $\boldsymbol{d}$ in G while satisfying the capacity constraints $\boldsymbol{0} \leq \boldsymbol{f} \leq \boldsymbol{1}$ and minimizing the cost $\sum_{e \in E} c_e f_e$, or certifies that no such flow exists.

Acknowledgment

KA and AM were supported in part by the NSF grants CCF-1553428 and CNS-1815221. AV was supported by NSF grant CCF-1718342. AV acknowledges Alina Ene for providing funding and useful discussions throughout the development of this project.

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, "Network flows," Alfred P. Sloan School of Management, Cambridge, MA, Tech. Rep., 1988.

[2] J. B. Orlin, "A polynomial time primal network simplex algorithm for minimum cost flows," *Mathematical Programming*, vol. 78, no. 2, pp. 109–129, 1997.

[3] 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.

[4] P. Christiano, J. A. Kelner, A. Mądry, D. A. Spielman, and S.-H. Teng, "Electrical flows, Laplacian systems, and faster approximation of maximum flow in undirected graphs," in *Proceedings of the 43rd Annual ACM Symposium on Theory of computing*, 2011, pp. 273–282.

[5] A. Mądry, "Navigating central path with electrical flows: From flows to matchings, and back," in *54th Annual Symposium on Foundations of Computer Science*. IEEE, 2013, pp. 253–262.

[6] Y. T. Lee, S. Rao, and N. Srivastava, "A new approach to computing maximum flows using electrical flows," in *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, 2013, pp. 755–764.

[7] J. Sherman, "Nearly maximum flows in nearly linear time," in *54th Annual Symposium on Foundations of Computer Science*. IEEE, 2013, pp. 263–269.

[8] J. A. Kelner, Y. T. Lee, L. Orecchia, and A. Sidford, "An almost-linear-time algorithm for approximate max flow in undirected graphs, and its multicommodity generalizations," in *Proceedings of the 25th Annual ACM-SIAM Symposium on Discrete algorithms*. SIAM, 2014, pp. 217–226.

[9] A. Mądry, "Computing maximum flow with augmenting electrical flows," in *57th Annual Symposium on Foundations of Computer Science*. IEEE, 2016, pp. 593–602.

[10] R. Peng, "Approximate undirected maximum flows in $O(m \text{ polylog } n)$ time," in *Proceedings of the 27th Annual ACM-SIAM Symposium on Discrete algorithms*. SIAM, 2016, pp. 1862–1867.

[11] M. B. Cohen, A. Mądry, P. Sankowski, and A. Vladu, "Negative-weight shortest paths and unit capacity minimum cost flow in $\tilde{O}(m^{10/7} \log W)$ time," in *Proceedings of the 28th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2017, pp. 752–771.

[12] J. Sherman, "Area-convexity, ℓ_∞ regularization, and undirected multicommodity flow," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, 2017, pp. 452–460.

[13] A. Sidford and K. Tian, "Coordinate methods for accelerating ℓ_∞ regression and faster approximate maximum flow," in *59th Annual Symposium on Foundations of Computer Science*. IEEE, 2018, pp. 922–933.

[14] Y. P. Liu and A. Sidford, "Faster energy maximization for faster maximum flow," in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing*, 2020.

[15] ——, "Faster divergence maximization for faster maximum flow," *arXiv:2003.08929*, 2020.

[16] K. Axiotis, A. Mądry, and A. Vladu, "Circulation control for faster minimum cost flow in unit-capacity graphs," *arXiv:2003.04863*, 2020.

[17] J. H. Wilkinson, *Rounding errors in algebraic processes*. Courier Corporation, 1994.

[18] N. J. Higham, *Accuracy and stability of numerical algorithms*. SIAM, 2002, vol. 80.

[19] R. Kyng, R. Peng, S. Sachdeva, and D. Wang, "Flows in almost linear time via adaptive preconditioning," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, 2019, pp. 902–913.

[20] A. V. Goldberg, S. Hed, H. Kaplan, and R. E. Tarjan, "Minimum-cost flows in unit-capacity networks," *Theory of Computing Systems*, vol. 61, no. 4, pp. 987–1010, 2017.

[21] D. Durfee, K. A. Lai, and S. Sawlani, "ℓ_1 regression using lewis weights preconditioning and stochastic gradient descent," in *Conference on Learning Theory*, 2018, pp. 1626–1656.

[22] S. Bubeck, M. B. Cohen, Y. T. Lee, and Y. Li, "An homotopy method for ℓ_p regression provably beyond self-concordance and in input-sparsity time," in *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, 2018, pp. 1130–1137.

[23] D. Adil, R. Kyng, R. Peng, and S. Sachdeva, "Iterative refinement for ℓ_p-norm regression," in *Proceedings of the 30th Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2019, pp. 1405–1424.

[24] D. Adil and S. Sachdeva, "Faster p-norm minimizing flows, via smoothed q-norm problems," in *Proceedings of the 31st Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2020, pp. 892–910.

[25] J. A. Kelner, L. Orecchia, A. Sidford, and Z. A. Zhu, "A simple, combinatorial algorithm for solving SDD systems in nearly-linear time," in *Proceedings of the 45th Annual ACM Symposium on Theory of Computing*, 2013, pp. 911–920.