High-precision Estimation of Random Walks in Small Space

AmirMahdi Ahmadinejad*, Jonathan Kelner†, Jack Murtagh‡, John Peebles§, Aaron Sidford*, Salil Vadhan‡
*Department of Management Science & Engineering, Stanford University, Stanford, CA USA
e-mail: ahmadi@stanford.edu, sidford@stanford.edu
†Department of Mathematics, MIT, Cambridge, MA USA
e-mail: kelner@mit.edu
‡School of Engineering & Applied Sciences, Harvard University, Cambridge, MA USA
e-mail: jmurtagh@alumni.harvard.edu, salil_vadhan@harvard.edu
§Computer Science Department, Yale University, New Haven, CT USA
e-mail: john.peebles@yale.edu

***Abstract*—In this paper, we provide a deterministic $\tilde{O}(\log N)$-space algorithm for estimating random walk probabilities on undirected graphs, and more generally Eulerian directed graphs, to within inverse polynomial additive error ($\epsilon = 1/\text{poly}(N)$) where N is the length of the input. Previously, this problem was known to be solvable by a randomized algorithm using space $O(\log N)$ (following Aleliunas et al., FOCS '79) and by a deterministic algorithm using space $O(\log^{3/2} N)$ (Saks and Zhou, FOCS '95 and JCSS '99), both of which held for arbitrary directed graphs but had not been improved even for undirected graphs. We also give improvements on the space complexity of both of these previous algorithms for non-Eulerian directed graphs when the error is negligible ($\epsilon = 1/N^{\omega(1)}$), generalizing what Hoza and Zuckerman (FOCS '18) recently showed for the special case of distinguishing whether a random walk probability is 0 or greater than ϵ.**

We achieve these results by giving new reductions between powering Eulerian random-walk matrices and inverting Eulerian Laplacian matrices, providing a new notion of spectral approximation for Eulerian graphs that is preserved under powering, and giving the first deterministic $\tilde{O}(\log N)$-space algorithm for inverting Eulerian Laplacian matrices. The latter algorithm builds on the work of Murtagh et al. (FOCS '17) that gave a deterministic $\tilde{O}(\log N)$-space algorithm for inverting undirected Laplacian matrices, and the work of Cohen et al. (FOCS '19) that gave a randomized $\tilde{O}(N)$-time algorithm for inverting Eulerian Laplacian matrices. A running theme throughout these contributions is an analysis of "cycle-lifted graphs," where we take a graph and "lift" it to a new graph whose adjacency matrix is the tensor product of the original adjacency matrix and a directed cycle (or variants of one).

***Keywords*—derandomization, space complexity, random walks, Markov chains, Laplacian systems, spectral sparsification, Eulerian graphs**

This is an extended abstract. Please see the full version of our paper [1] for precise statements of our results and missing proofs.

I. Introduction

We give the first deterministic, nearly logarithmic-space algorithm for accurately estimating random walk probabilities on undirected graphs. Our algorithm extends to Eulerian digraphs, which are directed graphs where the indegree of a vertex v is equal to its outdegree for every vertex v. (Note that a random walk on an undirected graph is equivalent to a random walk on the associated Eulerian digraph obtained by replacing each undirected edge $\{u, v\}$ with two directed edges (u, v) and (v, u).) In more detail, our main result is as follows:

Theorem I.1 (informally stated; see [1])**.** *There is a deterministic, $\tilde{O}(\log(k \cdot N))$-space algorithm that given an Eulerian digraph G (or an undirected graph G), two vertices s, t, and a positive integer k, outputs the probability that a k-step random walk in G started at s ends at t, to within additive error of ϵ, where N is the length of the input and $\epsilon = 1/\text{poly}(N)$ is any desired polynomial accuracy parameter.*

Estimating random walk probabilities to inverse polynomial accuracy, even in general digraphs, can easily be done by randomized algorithms running in space $O(\log N)$, since that much space is sufficient to simulate random walks [2]. In fact, estimating random walk probabilities in general digraphs is complete for randomized logspace.[1] The best known deterministic algorithm prior to our work was that of Saks and Zhou [3], which runs in space $O(\log^{3/2} N)$, and works for general digraphs. (See the excellent survey of Saks [4] for more discussion of the close connection between randomized space-bounded com-

[1]Formally, given G, s, t, k, a threshold τ, and a unary accuracy parameter 1^a, the problem of deciding whether the k-step random walk probability from s to t is larger than $\tau + 1/a$ or smaller than τ is complete for the class BPL of promise problems having randomized logspace algorithms with two-sided error. By binary search over the threshold τ, this problem is log-space equivalent to estimating the same probability to within error $1/a$.

2575-8454/20/$31.00 ©2020 IEEE
DOI 10.1109/FOCS46700.2020.00123

putation and random walks, as well as the state-of-art in derandomizing such computations up to the mid-1990's.)

For undirected graphs, Murtagh et al. [5] recently gave a deterministic $\tilde{O}(\log(k \cdot N))$-space algorithm that computes a much weaker form of approximation for random walks: given any subset S of vertices, the algorithm estimates, to within a multiplicative factor of $(1 \pm 1/\text{polylog}(N))$, the *conductance* of the set S, namely the probability that a k-step random walk started at a random vertex of S (with probability proportional to vertex degrees) ends outside of S. Our result is stronger because in undirected graphs, all nonzero conductances can be shown to be of magnitude at least $1/\text{poly}(N)$ and can be expressed as a sum of at most $N^2/4$ random walk probabilities. Consequently, with the same space bound our algorithm can estimate the conductance of any set S to within a multiplicative factor of $(1 \pm 1/\text{poly}(N))$.

Like [5], our work is part of a larger project, initiated in [6], that seeks to make progress on the derandomization of space-bounded computation by importing ideas from the literature on time-efficient randomized algorithms for solving graph Laplacian systems [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19] . While we consider Theorem I.1 to be a natural derandomization result in its own right, it and our analogous result for solving Eulerian Laplacian systems (Theorem I.2 below) can also be viewed as a step toward handling general directed graphs and thereby having an almost-complete derandomization of randomized logspace. Indeed, in recent years, nearly linear-time randomized algorithms for estimating properties of random walks on general directed graphs (with polynomial mixing time, which also yield complete problems for randomized logspace [20]) were obtained by reduction to solving Eulerian Laplacian systems [16], [17], [18]. A deterministic and sufficiently space-efficient analogue of such a reduction, combined with our results, would put randomized logspace in deterministic space $\tilde{O}(\log N)$, i.e. $\text{BPL} \subseteq \widetilde{\text{L}}$.

To achieve our main result and prove Theorem I.1, we provide several results that may be of interest even outside of the space-bounded derandomization context, such as a new notion of spectral approximation and new reductions between estimating random walk probabilities and inverting Laplacian systems. Below we provide more details on how our work relates to both the space-bounded derandomization and the Laplacian solving literature, and describe some of our other contributions.

A. Derandomization of Space-Bounded Computation

It is widely conjectured that every problem solvable in randomized logarithmic space can also be solved in deterministic logarithmic space (i.e. RL = L, BPL=L for the one-sided and two-sided error versions, respectively) [4]. Though this is known to follow from mild assumptions in complexity theory (e.g. that there is a Boolean function in $\mathbf{DSPACE}(n)$ that requires branching programs of size $2^{\Omega(n)}$ [21]), the best known unconditional derandomization is the aforementioned, quarter-century-old result of Saks and Zhou [3], which places randomized logspace in deterministic space $O(\log^{3/2} N)$.

Most of the effort on derandomizing logspace computations over the past three decades has gone towards the design of *pseudorandom generators* that fool *ordered branching programs*. An ordered branching program of *width* w and *length* k is given by a directed graph on vertex set $[k] \times [w]$. which we view as consisting of k layers of w vertices. All of the edges from the ith layer go to the $(i+1)$'st layer (so there are no edges entering the first layer or exiting the last layer). We call the first vertex of the first layer (i.e. vertex $(1,1)$) the *start vertex*, and the first vertex of the last layer (i.e. vertex $(k,1)$) the *accept vertex* t. Typically, every vertex in layers 1 to $k-1$ has outdegree 2, with the two edges labelled by 0 and 1. Intuitively, the vertices in the ith layer correspond to possible states of a space-bounded algorithm before it makes its ith coin toss, and the two edges lead to its two possible states after that coin toss. The acceptance probability of an ordered branching program is exactly the probability that a random walk from the start vertex s of length $k-1$ ends at accept vertex t. Generating such a truly random walk takes $k-1$ random bits, so the goal of a pseudorandom generator for ordered branching programs is to generate a walk of length $k-1$ using a much shorter random seed such that the acceptance probability is preserved up to an additive ϵ. Given such a pseudorandom generator, we can obtain a deterministic algorithm for estimating the acceptance probability by enumerating all seeds of the pseudorandom generator.

A general $O(\log N)$-space computation can have $w = 2^{O(\log N)} = \text{poly}(N)$ states and toss $k = \text{poly}(N)$ coins. The best known pseudorandom generator for such ordered branching programs is the classic generator of Nisan [22], which has a seed length of $O(\log^2 N)$ (for any error $\epsilon \geq 1/\text{poly}(N)$) and thus does not directly yield a derandomization of space complexity better than $O(\log^2 N)$ (due to enumerating the seeds), which can be achieved more easily by recursively multiplying the transition matrices between layers. (Multiplying k boolean $w \times w$ matrices to within a final accuracy of ϵ can be done recursively in space $O((\log k) \cdot (\log w + \log\log(k/\epsilon)))$.) Nisan's generator is also a crucial tool in the algorithm of Saks and Zhou [3].

Due to the long lack of progress in improving Nisan's generator, effort has turned to restricted classes of branching programs, such as those of constant width ($w = O(1)$), with there being significant progress in recent years for the case of width $w = 3$. [23], [24], [25]. Another restriction that has been studied is that of *regular* branching programs, where every vertex in layers $2, \ldots, k$ in the branching program has indegree 2. For this case, Braverman, Rao, Raz, and Yehudayoff [26] give a pseudorandom generator with seed length $\tilde{O}(\log N)$ when $w \leq \text{polylog}(N)$ and $\epsilon \geq 1/\text{polylog}(N)$, which again does not yield a deter-

ministic algorithm that improves upon recursive matrix multiplications.

In contrast, our algorithm for Eulerian graphs can be used to estimate the acceptance probability of a regular branching program in space $\tilde{O}(\log N)$ even when $w = \text{poly}(N)$ and $\epsilon = 1/\text{poly}(N)$. Indeed, by adding edges from the kth layer back to the first layer, a regular branching program can be made into an Eulerian graph, without changing the probability that a random walk of length $k-1$ from s ends at t. In addition, our techniques also yield an improved pseudorandom generator for *permutation branching programs* (regular branching programs where the labelling is constrained so that for each $b \in \{0,1\}$, the edges labelled b form perfect matchings between the vertices in consecutive layers). Specifically, [27] use our results to derive a pseudorandom generator with seed length $\tilde{O}(\log N)$ for permutation branching programs (with a single accept vertex in layer k) of *unbounded* width w, albeit with error only $\epsilon = 1/\text{polylog}(N)$. Even for the special case of permutation branching programs, seed length $\tilde{O}(\log N)$ was previously only achieved for width $w = \text{polylog}(N)$ [28], [29], [30].

It is also worth comparing our result with Reingold's Theorem, which gives a deterministic logspace algorithm for deciding s-t connectivity in undirected graphs. Reingold, Trevisan, and Vadhan [20] interpreted and generalized Reingold's methods to obtain "pseudoconverging walk generators" for regular digraphs where each edge label forms a permutation of the vertices, as in the permutation branching programs described above. These generators provide a way to use a seed of $O(\log N)$ random bits to generate walks of length $\text{poly}(N)$ that converge to the uniform distribution on the connected component of the start vertex (just like a truly random walk would). Such generators turn out to suffice for deciding s-t connectivity on arbitrary Eulerian digraphs. However, these generators are not guaranteed to closely approximate the behavior of a truly random walk at shorter walk lengths. Indeed, even the length of the walks needed for mixing is polynomially larger than with a truly random walk. Nevertheless, one of the techniques we use, the *derandomized square*, originated from an effort to simplify Reingold's algorithm and these pseudoconverging walk generators [31].

Our work builds on recent papers of Murtagh et al. [5], [6], which gave deterministic, nearly logarithmic-space algorithms for estimating certain quantities associated with random walks on undirected graphs. Specifically, the first of these papers [6] gave a "Laplacian solver" (defined below) that implied accurate (to within $1/\text{poly}(N)$ error) estimates of *escape probabilities* (the probability that a random walk from s visits t before visiting another vertex v); these again refer to the long-term behavior of random walks, rather than the behavior at a given time below mixing. The second paper [5] dealt with random walks of a fixed length k, but as discussed earlier, only gave a weak approximation to the conductance of subsets of vertices.

B. Inverting Laplacian Systems

We prove Theorem I.1 by a novel reduction from estimating k-step random walk probabilities to solving linear systems given by graph Laplacians, and giving a small-space algorithm for the latter in the case of Eulerian graphs.

Let G be a digraph on n vertices, and let $\mathbf{W}$ be the $n \times n$ transition matrix for the random walk on G. Then we will call $\mathbf{L} = \mathbf{I} - \mathbf{W}$ the *random-walk Laplacian of* G.[2] Solving Laplacian systems refers to the problem of given a vector $b \in \mathbb{R}^n$, finding a vector $x \in \mathbb{R}^n$ such that $\mathbf{L}x = b$ (if one exists). Since the matrix $\mathbf{L}$ is not invertible (a stationary distribution of the random walk on G is in the kernel), a Laplacian system can be solved by instead computing its *pseudoinverse* $\mathbf{L}^+$, which acts as an inverse on $\text{Image}(\mathbf{L})$, and is zero on the orthogonal complement of $\text{Image}(\mathbf{L})$.

We show that we can compute the pseudoinverse of an Eulerian Laplacian in nearly logarithmic space.

Theorem I.2 (informally stated; see [1])**.** *There is a deterministic, $\tilde{O}(\log N)$-space algorithm that given an Eulerian digraph G with random-walk transition matrix $\mathbf{W}$, outputs a matrix $\widetilde{\mathbf{L}^+}$ whose entries differ from $\mathbf{L}^+$ by at most an additive ϵ, where N is the length of the input and $\epsilon = 1/\text{poly}(N)$ is any desired polynomial accuracy parameter.*

Previously, Cohen et al. [17], [18] showed how to solve Eulerian Laplacian systems by randomized, nearly linear-time algorithms, and Murtagh et al. [6] showed how to solve undirected Laplacian systems by deterministic, nearly logarithmic-space algorithms. Our proof of Theorem I.2 draws on all of these works.

As explained below, the extension from undirected graphs (handled by [6]) to Eulerian graphs (as in Theorem I.2) is crucial for obtaining our high-precision estimation of random walks (Theorem I.1) even for the case of undirected graphs. In addition, as discussed earlier, this extension can also be viewed as a step toward handling general directed graphs and thereby having an almost-complete derandomization of randomized logspace.

We will describe the ideas in the proof of Theorem I.2 below in Section I-C. Here we describe our reduction from powering (Theorem I.1) to computing the pseudoinverse of a Laplacian (Theorem I.2).

Let G be an n-vertex digraph with random-walk transition matrix $\mathbf{W}$. Let $\mathbf{P}_k$ be the adjacency matrix of a k-vertex directed path. Note that $\mathbf{P}_k$ is not stochastic, but rather *substochastic* (nonnegative with column sums at most 1), since there are no edges leaving the last vertex

[2]The standard Laplacian of G, which we simply refer to as the *Laplacian of* G is $\mathbf{D}-\mathbf{A}$, where $\mathbf{D}$ is the diagonal matrix of outdegrees and $\mathbf{A}$ is the adjacency matrix (where we define $(\mathbf{A})_{ij}$ to be the number of edges from j to i in G). Notice that $\mathbf{I}-\mathbf{W} = (\mathbf{D}-\mathbf{A})\mathbf{D}^{-1}$. For undirected graphs, it is common to use a symmetric normalization of the Laplacian given by $\mathbf{D}^{-1/2}(\mathbf{D}-\mathbf{A})\mathbf{D}^{-1/2} = \mathbf{D}^{-1/2}(\mathbf{I}-\mathbf{W})\mathbf{D}^{1/2}$, but the $\mathbf{I}-\mathbf{W}$ formulation will be more convenient for us.

of the path (i.e. random walks "die off" when leaving that vertex). Then the $kn \times kn$ matrix $\mathbf{W'} = \mathbf{P}_k \otimes \mathbf{W}$, i.e. the Kronecker product of $\mathbf{P}_k$ and $\mathbf{W}$, is a $k \times k$ block matrix consisting of $n \times n$ blocks that equal $\mathbf{W}$ just below the diagonal and are zero elsewhere. For example, when $k = 4$, we have:

$$\mathbf{W'} = \mathbf{P}_k \otimes \mathbf{W} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \mathbf{W} & 0 & 0 & 0 \\ 0 & \mathbf{W} & 0 & 0 \\ 0 & 0 & \mathbf{W} & 0 \end{bmatrix}.$$

$\mathbf{W'}$ is also a substochastic matrix describing random walks on a graph with k layers of n vertices each, where there is a bipartite version of G going from the ith layer to the $(i+1)$'st layer for each $i = 1, \ldots, k-1$, and again there are no edges leaving the kth layer. We call this the *path-lift of G of length k*, or the *path-lifted graph* when G and k are clear from context. (This construction is inspired by the ordered branching programs that arise in space-bounded computation, as described above.)

The "Laplacian" of this layered graph, $\mathbf{L} = \mathbf{I}_{kn} - \mathbf{W'}$, is invertible, and noting that $(\mathbf{W'})^k = 0$, we can calculate its inverse as:

$$\mathbf{L}^{-1} = \mathbf{I}_{nk} + \mathbf{W'} + (\mathbf{W'})^2 + \cdots + (\mathbf{W'})^{k-1} \quad (1)$$
$$= \mathbf{I}_k \otimes \mathbf{I}_n + \mathbf{P}_k \otimes \mathbf{W} + \mathbf{P}_k^2 \otimes \mathbf{W}^2 + \cdots \mathbf{P}_k^{k-1} \otimes \mathbf{W}^{k-1}. \quad (2)$$

Thinking of $\mathbf{L}^{-1}$ as a block matrix, the term $\mathbf{P}_k^j \otimes \mathbf{W}^j$ places a copy of $\mathbf{W}^j$ in each of the blocks that are in the jth diagonal below the main diagonal. (So on the main diagonal are blocks of $\mathbf{I}_n$, just below the main diagonal are blocks of $\mathbf{W}$, below that $\mathbf{W}^2$, and so on.) For example, for $k = 4$, we can write $\mathbf{L}^{-1}$ in block form as

$$\mathbf{L}^{-1} = \begin{bmatrix} \mathbf{I} & 0 & 0 & 0 \\ \mathbf{W} & \mathbf{I} & 0 & 0 \\ \mathbf{W}^2 & \mathbf{W} & \mathbf{I} & 0 \\ \mathbf{W}^3 & \mathbf{W}^2 & \mathbf{W} & \mathbf{I} \end{bmatrix}.$$

Thus from an accurate estimate of $\mathbf{L}^{-1}$, we can read off accurate estimates of the powers of $\mathbf{W}$. For example, entry $((\ell, t), (1, s))$ of $\mathbf{L}^{-1}$ is exactly the probability that a length $\ell - 1$ random walk in G started at s ends at t.

However, we can only apply Theorem I.2 directly if $\mathbf{L}$ is the Laplacian of an Eulerian graph, and the above $\mathbf{W'}$ is not even stochastic. We can fix this by (a) starting with an Eulerian graph G and (b) considering a *cycle-lifted graph* instead of a path-lifted graph, i.e. considering transition matrix $\mathbf{C}_k \otimes \mathbf{W}$. Additionally, it is convenient to collapse all of the vertices in layer k to a single vertex v. Then it turns out from the Laplacian pseudoinverse $\mathbf{L}^+$, it is possible to read off *escape probabilities* — the probability that a random walk from one vertex, say $(1, s)$, visits another vertex, say (ℓ, t), before visiting a third vertex, say v. The condition "before visiting v" allows us to not worry about walks that traverse all the way around the cycle, and thus we get exactly the probability that a length $\ell - 1$ random walk in G started at s ends at t.[3]

Note that even if G is undirected, this reduction requires inverting a Laplacian of a directed layered graph. Thus, our extension of the small-space undirected Laplacian solver of [6] to Eulerian graphs (Theorem I.2) seems essential for obtaining high-precision estimates of powers even for undirected graphs (Theorem I.1).

The reduction above from computing powers to inverting also allows us to obtain new algorithms for general digraphs and Markov chains:

Theorem I.3 (informally stated; see [1])**.** *Given a Markov chain G specified by a stochastic matrix $\mathbf{W}$, two states s, t, and a positive integer k, we can compute the probability that a k-step random walk in G started at s ends at t, to within an additive error of ϵ:*

1) *In randomized space $O((\log Nk) \cdot \log(\log_{Nk}(1/\epsilon)))$, or*
2) *In deterministic space $O(\log^{3/2}(Nk) + (\log(Nk)) \cdot \log(\log_{Nk}(1/\epsilon)))$.*

where N is the length of the input.

This theorem generalizes one of the results from recent work of Hoza and Zuckerman [32] that gave the same bounds for the 1-sided version of the problem, namely distinguishing probability 0 from probability greater than ϵ. For the two-sided version of the problem that we consider, a randomized algorithm using space $O(\log(Nk/\epsilon))$ follows from performing $\mathrm{poly}(1/\epsilon)$ random walks and counting how many end at t. For deterministic algorithms, the best previous algorithm is from Saks and Zhou [3], which uses space $O(\log(Nk/\epsilon) \cdot \log^{1/2} k)$. Note that Theorem I.3 has a doubly-logarithmic dependence on ϵ rather than a singly-logarithmic one. In particular, Saks and Zhou [3] only achieves space $O(\log^{3/2} Nk)$ for $\epsilon = 1/\mathrm{poly}(Nk)$ whereas we achieve it for a much smaller $\epsilon = 1/\exp(\exp(\sqrt{\log Nk}))$.

The proof of Theorem I.3 begins with the observation that we can approximate $\mathbf{L}^{-1} = (\mathbf{I}_{nk} - \mathbf{P}_k \otimes \mathbf{W})^{-1}$ to within accuracy $1/\mathrm{poly}(N, k)$ in randomized space $O(\log Nk)$ or deterministic space $O(\log^{3/2} Nk)$. Indeed, by Equation (1), it suffices to estimate $\mathbf{I}, \mathbf{W}, \mathbf{W}^2, \ldots, \mathbf{W}^{k-1}$ up to accuracy $\pm 1/\mathrm{poly}(N, k)$.

We then use the fact that matrix inversion has a very efficient error reduction procedure, equivalent to what is commonly known as "preconditioned Richardson iterations". Let $\widetilde{\mathbf{L}^{-1}}$ denote our approximation to $\mathbf{L}^{-1}$ with error $1/\mathrm{poly}(N, k)$. For an appropriate choice of the polynomial error bound, it follows that the "error matrix" $\mathbf{E} = \mathbf{I}_{nk} - \widetilde{\mathbf{L}^{-1}}\mathbf{L}$ has norm at most $1/Nk$ (in, say, spectral

[3] Alternatively (and essentially equivalently), we could note that if $D - A$ is the Laplacian of G, then $I_k \otimes D - P_k \otimes A = I_k \otimes D \cdot (I_{nk} - P_k \otimes W)$ is a "row-column diagonally dominant matrix" and apply the reduction from inverting such matrices to pseudo-inverting Eulerian Laplacian systems [16].

norm). Then we can obtain a more accurate estimate of $\mathbf{L}^{-1}$ by using the identity:

$$\mathbf{L}^{-1} = (\mathbf{I}_{nk} - \mathbf{E})^{-1}\widetilde{\mathbf{L}^{-1}} = (\mathbf{I}_{nk} + \mathbf{E} + \mathbf{E}^2 + \mathbf{E}^3 + \cdots)\widetilde{\mathbf{L}^{-1}}.$$

Since $\mathbf{E}$ has norm at most $1/Nk$, the series converges very quickly, and can be truncated at $O(\log_{Nk}(1/\epsilon))$ terms to achieve an approximation to within $\pm\epsilon$. As noted earlier, from such an accurate estimation of $\mathbf{L}^{-1} = (\mathbf{I}_{nk} - \mathbf{P}_k \otimes \mathbf{W})^{-1}$, we can accurately estimate random walks of length $k-1$.

This same error reduction procedure is also used in our proof of Theorem I.2 (and also throughout the literature on time-efficient Laplacian solvers), and thus is also key to the high precision estimates we obtain in Theorem I.1. Although we fixed error $1/\mathrm{poly}(N)$ in the statement of the theorem, we can also achieve smaller error ϵ at a price of $O((\log N) \cdot \log(\log_N(1/\epsilon)))$ in the space complexity.

Interestingly, early work on randomized space-bounded computation [33], [34], [35], [36] reduced the problem of *exactly* computing *hitting probabilities* of Markov chains (the probability that an *infinite* random walk from s ever hits t) to computing $(\mathbf{I}-\mathbf{W})^{-1}$ for a substochastic matrix $\mathbf{W}$, and used this to show that *unbounded-error* and *non-halting* randomized logspace is contained in deterministic space $O(\log^2 N)$. As far as we know, ours is the first application of inverting Laplacian systems to estimating finite-time random-walk probabilities to within polynomially small error, and consequently it is also the first application of inverting Laplacian systems to the commonly accepted formulation of randomized logspace (i.e. bounded error and halting).

C. Complex spectral approximation, cycle-lifted graphs, and powering

We now describe the techniques underlying our space-efficient Eulerian Laplacian inverter (Theorem I.2). Let $\mathbf{W}$ be the transition matrix for the random walk on an n-vertex Eulerian graph G, for which we want to estimate $(\mathbf{I} - \mathbf{W})^{-1}$. Because of (a generalization of) the error-reduction procedure described above, it suffices to compute a rough approximation to $(\mathbf{I} - \mathbf{W})^{-1}$. For symmetric matrices (as arising from undirected graphs), a sufficient notion of approximation is *spectral approximation* as introduced by Spielman and Teng [37]. Applying the Spielman–Teng notion to symmetric Laplacians $\mathbf{I}-\mathbf{W}$ and $\mathbf{I} - \widetilde{\mathbf{W}}$, we say that $\widetilde{\mathbf{W}}$ is an *ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$) if $\forall x \in \mathbb{R}^n$,

$$\left|x^\top(\mathbf{W} - \widetilde{\mathbf{W}})x\right| \leq \epsilon \cdot x^\top(\mathbf{I} - \mathbf{W})x = \epsilon \cdot \left(\|x\|^2 - x^\top\mathbf{W}x\right). \tag{3}$$

Cohen et al. [17] introduced a generalization of spectral approximation for asymmetric matrices and directed graphs: $\forall x, y \in \mathbb{R}^n$,

$$\left|x^\top(\mathbf{W} - \widetilde{\mathbf{W}})y\right| \leq \frac{\epsilon}{2} \cdot \left(\|x\|^2 + \|y\|^2 - x^\top\mathbf{W}x - y^\top\mathbf{W}y\right). \tag{4}$$

In the case of symmetric matrices, their definition is equivalent to the Spielman–Teng notion, and thus we use the same terminology *ϵ-approximation* and notation $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ for their notion.

In this paper, we introduce a stronger notion of spectral approximation. Specifically, we say $\widetilde{\mathbf{W}}$ is a *unit-circle ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$) if $\forall x, y \in \mathbb{C}^n$

$$\left|x^*(\mathbf{W} - \widetilde{\mathbf{W}})y\right| \leq \frac{\epsilon}{2} \cdot \left(\|x\|^2 + \|y\|^2 - |x^*\mathbf{W}x + y^*\mathbf{W}y|\right), \tag{5}$$

where v^* refers to the conjugate transpose of v. Note that we now allow the vectors to range over $\mathbb{C}^n$ rather than $\mathbb{R}^n$. As we show (see Section III) this in itself does not make the definition stronger as the earlier notions of [7], [17] have equivalent formulations using complex vectors. The more important change is the introduction of the complex magnitude $|\cdot|$ in the term $|x^*\mathbf{W}x + y^*\mathbf{W}y|$.

The significance of this change can be seen by considering an eigenvector v of $\mathbf{W}$ whose eigenvalue $\lambda \in \mathbb{C}$ has magnitude 1. Consider what happens if we set $x = y = v$ in both the Spielman–Teng definition (3) and our definition (5). If $\lambda = 1$ (e.g. v is a stationary distribution of the random walk specified by $\mathbf{W}$), then the right-hand side of the inequality in both cases is zero, so we must have exact equality on the left-hand side, i.e. $v^*\widetilde{\mathbf{W}}v = v^*\mathbf{W}v$. On the other hand, if λ is some other root of unity (e.g. an eigenvalue of the k-cycle $\mathbf{C}_k$, or in any random walk with periodicity), then only our definition requires exact equality. This also explains our terminology *unit-circle* approximation. It also can be shown that $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$ if and only if $z\widetilde{\mathbf{W}} \approx_\epsilon z\mathbf{W}$ for all complex z of magnitude 1. That is, our definition amounts to demanding that all unit-circle multiples of the matrices approximate each other in the previous sense. In the case of symmetric matrices (undirected graphs), it suffices to consider $z = \pm 1$, corresponding to the fact that the eigenvalues are all real and the only periodicity that can occur is 2.

The benefit of unit-circle approximation is that, unlike the previous notions of spectral approximation, it is preserved under cycle-lifts and powers.

Lemma I.4. *Suppose $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$. Then for all $k \in \mathbb{N}$, we have:*

1) $\mathbf{C}_k \otimes \widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{C}_k \otimes \mathbf{W}$, *and*
2) $\widetilde{\mathbf{W}^k} \overset{\circ}{\approx}_{\epsilon/(1-\epsilon)} \mathbf{W}^k$.

We note that in previous work ([6], [38]), it was observed that for symmetric matrices, if $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ and $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$ then we do get $\widetilde{\mathbf{W}^2} \approx_\epsilon \mathbf{W}^2$. Lemma I.4 holds even for asymmetric matrices and handles all powers k.

Item 1 is proven by observing that the diagonalization of $\mathbf{C}_k$ (using the discrete Fourier basis, which are its eigenvectors) has all kth roots of unity along the diagonal, so approximation of the cycle-lifted graphs $\mathbf{C}_k \otimes \widetilde{\mathbf{W}}$ and $\mathbf{C}_k \otimes \mathbf{W}$ amounts to requiring that the approximation of $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ is preserved under multiplication by k'th

roots of unity. Item 2 is derived from Item 1 by observing that the kth powers can be obtained by "short-cutting" random walks through all but one layer of the cycle-lifted graphs. This shortcutting amounts to taking the Schur complements of the corresponding Laplacians, and we show that taking Schur complements of Eulerian Laplacians preserves spectral approximation (generalizing analogous results for undirected and symmetrized Schur complements [18], [39]).

Now we can sketch our algorithm that we use to prove Theorem I.2. Given an Eulerian digraph G, we want to approximate the pseudoinverse of the Laplacian $\mathbf{I}_n - \mathbf{W}$. By standard reductions, we may assume that G is regular, connected, and aperiodic, and therefore, it has polynomial mixing time. Rather than directly approximating the inverse of the Laplacian $\mathbf{I}_n - \mathbf{W}$ of the original graph, we instead approximate the inverse of the Laplacian of the cycle-lifted graph, i.e. $\mathbf{I}_{2^k \cdot n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$, for 2^k larger than the mixing time of $\mathbf{W}$. Then the pseudoinverse of $\mathbf{I}_n - \mathbf{W}$ can be well-approximated by an appropriate $n \times n$ projection of the pseudoinverse of $\mathbf{I}_{2^k \cdot n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$.

To approximate the pseudoinverse of $\mathbf{I}_{2^k \cdot n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$, we follow the recent approach of [18] and recursively compute an LU factorization (i.e. a product of a lower-triangular and upper-triangular matrix) that approximates $\mathbf{I}_{2^k \cdot n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$, as LU factorizations can be easily inverted. Each recursive step reduces the task to computing an LU factorization of the Laplacian of the random-walk on a chosen subset S of the vertices, where we short-cut steps of the walk through S^c. For our algorithm, we choose the set S to consist of every other layer of the cycle-lifted graph, as opposed to using a randomly chosen and pruned set of vertices as in [18]. Then shortcutting walks through S^c yields a graph on S whose transition matrix is equal to $\mathbf{C}_{2^{k-1}} \otimes \mathbf{W}^2$ — a cycle-lifted version of the two-step random walk, with a cycle of half the length! Unfortunately, we can't just directly recurse, because repeatedly squaring $\mathbf{W}$ k times takes space $O((k \cdot \log N))$. Thus, following [6], we utilize the "derandomized square" of [31], which produces an explicit sparse ϵ-approximation to $\mathbf{W}^2$ such that k iterated derandomized squares can be computed in space $O(\log N + k \cdot \log(1/\epsilon)) = \tilde{O}(\log N)$. (We take $\epsilon = 1/O(k)$ so that we can tolerate incurring an ϵ error in approximation for each of the k levels of recursion.) To make the analysis work, we prove that for regular digraphs, the derandomized square produces a graph $\widetilde{\mathbf{W}^2}$ that is a unit-circle approximation to $\mathbf{W}^2$, so that we can deduce that $\mathbf{C}_{2^{k-1}} \otimes \widetilde{\mathbf{W}^2}$ approximates $\mathbf{C}_{2^{k-1}} \otimes \mathbf{W}^2$ via Lemma I.4. Previous work [6] only showed approximation for undirected graphs, and with respect to the original Spielman-Teng notion of spectral approximation. (Rozenman and Vadhan [31] showed that for regular digraphs, the derandomized square improves spectral expansion nearly as much as the true square, but that is weaker than spectral approximation, as it only refers to the 2nd singular value rather than the entire spectrum.)

We remark that the $n \times n$ projection of the pseudoinverse of the approximate LU factorization we obtain is exactly the matrix we would get if we applied the repeated-squaring-based Laplacian inversion algorithm of Peng and Spielman [12] (or, more accurately, its space-efficient implementation via derandomized squaring [6]) to the original Laplacian $\mathbf{I} - \mathbf{W}$. Thus, another conceptual contribution of our paper is connecting the LU factorization approach of [18] to the squaring-based approach of [12] via the cycle-lifted graph. (However, for technical reasons in our analysis, we don't do the $n \times n$ projection until after applying the error-reduction procedure to obtain a highly accurate pseudoinverse of the cycle-lifted Laplacian.)

II. Preliminaries

In this section we introduce notation and facts we use through out the paper.

A. Notation

We denote by $\mathbb{C}$ the set of complex numbers. For $w = x + yi \in \mathbb{C}$, we use $w^* = x - yi$ to denote the conjugate of w. We use $|w| = \sqrt{x^2 + y^2}$ to denote the magnitude of w.

a) Matrices and vectors.: We use bold capital letters to denote matrices. We use $\mathbf{I}_n \in \mathbb{R}^{n \times n}$ to denote the identity matrix. For a matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ we use $\mathbf{A}^*$ to denote the conjugate transpose of $\mathbf{A}$ and we write $\mathbf{U}_\mathbf{A} = \frac{\mathbf{A} + \mathbf{A}^*}{2}$ to denote its *symmetrization*. We use $\vec{\mathbf{1}}_k \in \mathbb{R}^k$ to denote the all 1's vector or just $\vec{\mathbf{1}}$ when k is clear from context. We denote the conjugate transpose of a vector similarly. We use $\mathbf{A}^\top$ to denote the transpose of a real matrix.

b) Positive Semidefinite (PSD) matrices.: For Hermitian matrices $\mathbf{A}, \mathbf{B} \in \mathbb{C}^{n \times n}$ we say $\mathbf{A}$ is PSD or write $\mathbf{A} \succeq 0$ if $x^* \mathbf{A} x \geq 0$ for all $x \in \mathbb{C}^n$. If $\mathbf{A}$ is real the condition is equivalent to $x^\top \mathbf{A} x \geq 0$ for all $x \in \mathbb{R}^n$. Further we use $\mathbf{A} \succeq \mathbf{B}$ to denote $\mathbf{A} - \mathbf{B} \succeq 0$. We define $\preceq$, $\prec$, and $\succ$ analogously.

Proposition II.1. *Given a PSD matrix $\mathbf{A} \in \mathbb{C}^{n \times n}$ and matrix $\mathbf{B} \in \mathbb{C}^{n \times m}$*

$$\mathbf{B}^* \mathbf{A} \mathbf{B} \succeq 0.$$

c) Pseudo-inverse and square root of matrices.: For a matrix $\mathbf{A}$, we use $\mathbf{A}^+$ to denote the (Moore-Penrose) pseudo-inverse of $\mathbf{A}$. For a PSD matrix $\mathbf{B}$, we let $\mathbf{B}^{1/2}$ to denote the square root of $\mathbf{B}$, which is the unique PSD matrix such that $\mathbf{B}^{1/2} \mathbf{B}^{1/2} = \mathbf{B}$. Furthermore, we let $\mathbf{B}^{+/2}$ denote the pseudo-inverse of the square root of $\mathbf{B}$.

d) Operator norms.: For any vector norm $\|\cdot\|$ defined on $\mathbb{C}^n$ we define the operator semi-norm it induces on $\mathbb{C}^{n \times n}$ by $\|\mathbf{A}\| = \max_{x \neq 0} \frac{\|\mathbf{A}x\|}{\|x\|}$. For a PSD matrix $\mathbf{H}$ and vector x we let $\|x\|_\mathbf{H} = \sqrt{x^* \mathbf{H} x}$, and define the operator semi-norm $\|\mathbf{A}\|_\mathbf{H}$ accordingly. We can relate the $\|\cdot\|_\mathbf{H}$ and $\|\cdot\|_2$ operator norms as follows. For a matrix $\mathbf{A}$, we have $\|\mathbf{A}\|_\mathbf{H} = \|\mathbf{H}^{1/2} \mathbf{A} \mathbf{H}^{+/2}\|_2$. We use the term *spectral norm* to refer to the operator norm induced by $\|\cdot\|_2$. We

write $\|\mathbf{A}|_V\|_2$ to denote the spectral norm restricted to a subspace V. That is, $\|\mathbf{A}|_V\|_2 = \max_{x\in V, x\neq 0} \frac{\|\mathbf{A}x\|}{\|x\|}$.

Lemma II.2. *Let $\mathbf{M}\colon \mathbb{R}^n \to \mathbb{R}^n$ be a linear operator and extend it to $\mathbf{M}\colon \mathbb{C}^n \to \mathbb{C}^n$ by defining $\mathbf{M}(u+iv) = \mathbf{M}u + i\mathbf{M}v$ for all $z = u + iv \in \mathbb{C}^n$. Then $\|\mathbf{M}\|_{\mathbb{C}^n\to\mathbb{C}^n} = \|\mathbf{M}\|_{\mathbb{R}^n\to\mathbb{R}^n}$.*

e) Graphs: Throughout this paper we work with unweighted directed multigraphs (digraphs). These graphs can have parallel edges and self loops and can be viewed as digraphs with integer edge weighs. We specify graphs by $G = (V, E)$ where V is the set of vertices and E is the multiset of edges.

f) Adjacency and Random Walk Matrices.: The adjacency matrix of a digraph G on n vertices is the matrix $\mathbf{A} \in \mathbb{R}^{n\times n}$ where $\mathbf{A}_{ij}$ is the number of edges from vertex j to vertex i in G.[4] The degree matrix $\mathbf{D}$ of a digraph G is the diagonal matrix containing the out-degrees of the vertices in G. The random walk matrix or transition matrix of a digraph G is $\mathbf{W} = \mathbf{A}\mathbf{D}^{-1}$. $\mathbf{W}_{ij}$ is the probability that a random step from vertex j leads to i in G. Note that $\vec{\mathbf{1}}^\top\mathbf{W} = \vec{\mathbf{1}}^\top$. A matrix $\mathbf{W} \in \mathbb{R}^{n\times n}_{\geq 0}$ is called substochastic if $\vec{\mathbf{1}}^\top\mathbf{W} \leq \vec{\mathbf{1}}^\top$ (the inequality is entry-wise).

g) Directed Laplacians.: We follow the approach in [18] to define graph Laplacians. A matrix $\mathbf{L} \in \mathbb{R}^{n\times n}$ is a *directed Laplacian* if its off-diagonal entries are non-positive, i.e. $\mathbf{L}_{ij} \leq 0$ for $i \neq j$, and $\vec{\mathbf{1}}^\top\mathbf{L} = 0$. Every digraph is associated with a directed Laplacian. Occasionally we write $\mathbf{L} = \mathbf{D}-\mathbf{A}$ to express the decomposition of $\mathbf{L}$ into the degree matrix and adjacency matrix of the corresponding digraph. The random-walk Laplacian of a digraph with Laplacian $\mathbf{D} - \mathbf{A}$ is the matrix $(\mathbf{D} - \mathbf{A})\mathbf{D}^{-1} = \mathbf{I} - \mathbf{W}$, where $\mathbf{W}$ is the transition matrix of G. We will often write $\mathbf{C}_k$ to denote the adjacency matrix of the k-vertex unidirectional directed cycle (which is equal to its transition matrix, as the graph is 1-regular).

h) Eulerian graphs and Eulerian Laplacians.: A directed graph is Eulerian if the in-degree of every node is equal to its out-degree. A directed Laplacian $\mathbf{L}$ is Eulerian if $\mathbf{L}\vec{\mathbf{1}} = 0$. A graph is Eulerian if and only if its Laplacian is Eulerian.

B. Kronecker Product

Given matrices $\mathbf{A} \in \mathbb{C}^{n\times m}, \mathbf{B} \in \mathbb{C}^{p\times q}$, the Kronecker product or tensor product of $\mathbf{A}$ and $\mathbf{B}$ denoted by $\mathbf{A}\otimes\mathbf{B} \in \mathbb{C}^{pn\times qm}$ is

$$\mathbf{A}\otimes\mathbf{B} = \begin{bmatrix} \mathbf{A}_{11}\mathbf{B} & \mathbf{A}_{12}\mathbf{B} & \cdots & \mathbf{A}_{1m}\mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{A}_{n1}\mathbf{B} & \mathbf{A}_{n2}\mathbf{B} & \cdots & \mathbf{A}_{nm}\mathbf{B} \end{bmatrix}.$$

Proposition II.3. *Given four matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$, if the matrix dimensions make $\mathbf{AC}$ and $\mathbf{BD}$ well-defined, then*

$$(\mathbf{A}\otimes\mathbf{B})(\mathbf{C}\otimes\mathbf{D}) = (\mathbf{AC})\otimes(\mathbf{BD}).$$

[4]Often the adjacency matrix is defined to be $\mathbf{A}^\top$ but we find the current formulation more convenient for our purposes.

C. Schur Complement

For a matrix $\mathbf{A} \in \mathbb{C}^{n\times n}$ and sets $F, C \subseteq [n]$, let $\mathbf{A}_{FC}$ denote the submatrix corresponding to the rows in F and columns in C. Similarly, for a vector $v \in \mathbb{C}^n$ let $v_F \in \mathbb{C}^{|F|}$ be the restriction of v onto coordinates in F. If F, C partition $[n]$ and $\mathbf{A}_{FF}$ is invertible, then we denote the Schur complement of A onto the set C by

$$\mathrm{Sc}(\mathbf{A}, C) \stackrel{\text{def}}{=} \mathbf{A}_{CC} - \mathbf{A}_{CF}\mathbf{A}_{FF}^{-1}\mathbf{A}_{FC}.$$

When it is clear from context we may reload this notation as follows to make the Schur complement dimension consistent with $\mathbf{A}$.

$$\mathrm{Sc}(\mathbf{A}, C) \stackrel{\text{def}}{=} \begin{bmatrix} 0_{FF} & 0_{FC} \\ 0_{CF} & \mathbf{A}_{CC} - \mathbf{A}_{CF}\mathbf{A}_{FF}^{-1}\mathbf{A}_{FC} \end{bmatrix}.$$

III. Spectral Approximation

Since its introduction by Spielman and Teng [37], spectral approximation of graphs and their associated matrices [37] has served as a powerful tool for graph-theoretic algorithm development. Below we review the original definition and later generalizations to directed graphs and asymmetric matrices [17], and then present our new, stronger definition of unit-circle approximation in several equivalent formulations.

A. Definitions

Definition III.1 (Undirected Spectral Approximation [37])**.** Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{R}^{n\times n}$ be symmetric matrices. We say that $\widetilde{\mathbf{W}}$ is an *undirected ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$) if $\forall x \in \mathbb{R}^n$,

$$(1-\epsilon)\cdot x^\top(\mathbf{I}-\mathbf{W})x \leq x^\top(\mathbf{I}-\widetilde{\mathbf{W}})x \leq (1+\epsilon)\cdot x^\top(\mathbf{I}-\mathbf{W})x$$

or equivalently, $\forall x \in \mathbb{R}^n$,

$$\left|x^\top(\mathbf{W}-\widetilde{\mathbf{W}})x\right| \leq \epsilon\cdot x^\top(\mathbf{I}-\mathbf{W})x = \epsilon\cdot\left(\|x\|^2 - x^\top\mathbf{W}x\right).$$

Typically Definition III.1 is phrased in terms of Laplacian matrices of the form $\mathbf{I} - \mathbf{W}$ and approximation is denoted by $\mathbf{I}-\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{I}-\mathbf{W}$ to indicate the multiplicative approximation between the quadratic forms defined by $\mathbf{I}-\widetilde{\mathbf{W}}$ and $\mathbf{I}-\mathbf{W}$. However, in the more general definitions of this paper it will be more convenient to think of spectral approximation as a measure of approximation between $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ rather than between $\mathbf{I}-\widetilde{\mathbf{W}}$ and $\mathbf{I}-\mathbf{W}$. Note that the definition is asymmetric in $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ but $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ for $\epsilon < 1$ implies $\mathbf{W} \approx_{\epsilon/(1-\epsilon)} \widetilde{\mathbf{W}}$.

Spectral approximation is a strong definition that guarantees the two matrices have similar eigenvalues, and their corresponding graphs have similar cuts and random walk behavior [37], [40]. Below we show the generalization to directed graphs from [17].

Definition III.2 (Directed Spectral Approximation [17])**.** Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{R}^{n\times n}$ be (possibly asymmetric) matrices. We

say that $\widetilde{\mathbf{W}}$ is a *directed ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$) if $\forall x, y \in \mathbb{R}^n$,

$$\begin{aligned}&\left|x^\top(\mathbf{W}-\widetilde{\mathbf{W}})y\right| \\ &\leq \frac{\epsilon}{2}\cdot\left(x^\top(\mathbf{I}-\mathbf{W})x + y^\top(\mathbf{I}-\mathbf{W})y\right) \\ &= \frac{\epsilon}{2}\cdot\left(\|x\|^2+\|y\|^2 - x^\top\mathbf{W}x - y^\top\mathbf{W}y\right) \\ &= \frac{\epsilon}{2}\cdot\left(\|x\|^2+\|y\|^2 - x^\top\mathbf{U}_\mathbf{W}x - y^\top\mathbf{U}_\mathbf{W}y\right).\end{aligned}$$

The main difference between the above and Definition III.1 is the introduction of the y vector instead of having $y = x$. Indeed, using the same vector on both sides would lose the asymmetric information in the matrices $\mathbf{W}$ and $\widetilde{\mathbf{W}}$). However, note that the last inequality shows that the right-hand side depends only on the symmetrization $\mathbf{U}_\mathbf{W}$.

We are justified using the same notation for undirected and directed spectral approximation because of the following lemma

Lemma III.3 ([5] Lemma 2.9)**.** *Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{R}^{n\times n}$ be symmetric matrices. Then $\widetilde{\mathbf{W}}$ is a directed ϵ-approximation of $\mathbf{W}$ if and only if it is an undirected ϵ-approximation of $\mathbf{W}$.*

It will be convenient for us to generalize Definition III.2 to complex matrices. In that case, we will quantify over $x, y \in \mathbb{C}^n$ and replace the transposes with Hermitian transposes.

Definition III.4 (Complex Spectral Approximation)**.** Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^{n\times n}$ be (possibly asymmetric) matrices. We say that $\widetilde{\mathbf{W}}$ is a *complex ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$) if $\forall x, y \in \mathbb{C}^n$,

$$\begin{aligned}&\left|x^*(\mathbf{W}-\widetilde{\mathbf{W}})y\right| \\ &\leq \frac{\epsilon}{2}\cdot\left(\|x\|^2+\|y\|^2 - x^*\mathbf{U}_\mathbf{W}x - y^*\mathbf{U}_\mathbf{W}y\right) \\ &= \frac{\epsilon}{2}\cdot\left(\|x\|^2+\|y\|^2 - \mathrm{Re}\left(x^*\mathbf{W}x + y^*\mathbf{W}y\right)\right)\end{aligned}$$

The equality in Definition III.4 comes from the observation that for all $v \in \mathbb{C}^n$ and all matrices $\mathbf{A} \in \mathbb{C}^{n\times n}$ we have

$$\begin{aligned}v^*\mathbf{U}_\mathbf{A}v &= \frac{1}{2}\left(v^*\mathbf{A}v + v^*\mathbf{A}^*v\right) \\ &= \frac{1}{2}\left(v^*\mathbf{A}v + (v^*\mathbf{A}v)^*\right) = \mathrm{Re}(v^*\mathbf{A}v)\end{aligned}$$

because the average of a complex number and its conjugate is simply its real part. Notice that the definitions of undirected, directed, and complex spectral approximation are only achievable when the matrix $\mathbf{W}$ has the property that $\mathrm{Re}(x^*\mathbf{W}x) \leq \|x\|^2$ for all vectors $x \in \mathbb{C}^n$ (when $\mathbf{W}$ is real and symmetric as in the case of undirected spectral approximation, this requirement is equivalent to $x^\top\mathbf{W}x \leq \|x\|^2$ for all $x \in \mathbb{R}^n$). When working with these types of approximation, we will often implicitly restrain the matrices to have this property.

Again, we are justified in using the same notation for complex approximation that we use for directed and undirected approximations because of the following lemma.

Lemma III.5. *Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{R}^{n\times n}$ be (possibly asymmetric) matrices. Then $\widetilde{\mathbf{W}}$ is a directed ϵ-approximation of $\mathbf{W}$ if and only if $\widetilde{\mathbf{W}}$ is a complex ϵ-approximation of $\mathbf{W}$.*

A proof of Lemma III.5 can be found in the full version of this paper [1]. Now we introduce our new stronger definition, which we call *unit-circle spectral approximation.*

Definition III.6 (Unit-circle Spectral Approximation)**.** Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^{n\times n}$ be (possibly asymmetric) matrices. We say that $\widetilde{\mathbf{W}}$ is a *unit-circle ϵ-approximation* of $\mathbf{W}$ (written $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$) if $\forall x, y \in \mathbb{C}^n$,

$$\left|x^*(\mathbf{W}-\widetilde{\mathbf{W}})y\right| \leq \frac{\epsilon}{2}\cdot\left(\|x\|^2+\|y\|^2 - |x^*\mathbf{W}x + y^*\mathbf{W}y|\right).$$

The change from Definition III.4 is that we have replaced the real part with the complex magnitude $|\cdot|$ on the quadratic forms $x^*\mathbf{W}x + y^*\mathbf{W}y$ on the right-hand side. To understand what we gain from this, suppose $x = y$ is an eigenvector of $\mathbf{W}$ with eigenvalue λ such that $|\lambda| = 1$. Then the right-hand side of the inequality equals zero and so we must have $x^*\widetilde{\mathbf{W}}y = x^*\mathbf{W}y$. In other words, $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ must behave identically on the entire unit circle of eigenvalues with magnitude 1. This is in contrast to the previous definitions, which only required exact preservation in the case where $\lambda = 1$. For example, can an undirected bipartite graph (which has a periodicity of 2 and an eigenvalue of -1) have a non-bipartite spectral approximation? Under previous definitions, the answer is yes but under unit-circle approximation, the answer is no because we require exact preservation on all eigenvalues of magnitude 1, not just $\lambda = 1$.

Unit circle approximation applies to a smaller class of matrices than the previous definitions of spectral approximation. While the previous definitions only required that $\mathrm{Re}(x^*\mathbf{W}x) \leq \|x\|^2$ for all $x \in \mathbb{C}^n$, unit circle approximation requires that $|x^*\mathbf{W}x| \leq \|x\|^2$ for all $x \in \mathbb{C}^n$. Again, we will often implicitly restrict our matrices to have this property. Note that all complex matrices $\mathbf{W}$ such that $\|\mathbf{W}\|_1 \leq 1$ and $\|\mathbf{W}\|_\infty \leq 1$ satisfy this property. In particular, if $\mathbf{W}$ is the transition matrix of an Eulerian graph, then $\mathbf{W}$ satisfies the property as does $z \cdot \mathbf{W}$ for all $z \in \mathbb{C}$ such that $|z| \leq 1$.

We will see in the coming sections that unit-circle approximation is preserved under powering of $\widetilde{\mathbf{W}}$ and $\mathbf{W}$ and is useful for achieving spectral approximation of a class of graphs we call *cycle-lifted graphs*, which are essential for the analysis of our Eulerian Laplacian solver.

B. Equivalent Formulations

There are many useful equivalent formulations of Definition III.6. First we look at what our definition gives in

the case of real and symmetric matrices.

Lemma III.7 (Real, Symmetric Equivalence)**.** *Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{R}^{n\times n}$ be symmetric matrices. Then the following are equivalent:*

1) $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$.
2) $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ *and* $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$.
3) *For all* $x \in \mathbb{R}^n$ *we have*

$$\left|x^\top(\mathbf{W} - \widetilde{\mathbf{W}})x\right| \leq \epsilon \cdot \left(\|x\|^2 - |x^\top \mathbf{W} x|\right).$$

A proof of Lemma III.7 can be found in the full version of this paper [1]. In the original [37] formulation of spectral approximation as multiplicative approximation between quadratic forms, Lemma III.7 says that in the real, symmetric setting, unit-circle spectral approximation is equivalent to $\mathbf{I} - \widetilde{\mathbf{W}}$ approximating $\mathbf{I} - \mathbf{W}$ *and* $\mathbf{I} + \widetilde{\mathbf{W}}$ approximating $\mathbf{I} + \mathbf{W}$. This makes intuitive sense because symmetric matrices have real eigenvalues so the only eigenvalues that can lie on the unit circle are $+1$ and -1.

This "plus and minus" approximation has been studied before in [5], [38], where it was found to be useful because spectral approximation is preserved under squaring when both the "plus" and "minus" approximations hold. We will see in Section IV that even in the general directed, complex case, unit-circle approximation is preserved under all powering.

We now show some convenient equivalent formulations of unit-circle spectral approximation.

Lemma III.8. *Let $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^{n\times n}$ be (possibly asymmetric) matrices. Then the following are equivalent*

1) $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$
2) *For all* $z \in \mathbb{C}$ *such that* $|z| = 1$, $z \cdot \widetilde{\mathbf{W}} \approx_\epsilon z \cdot \mathbf{W}$
3) *For all* $z \in \mathbb{C}$ *such that* $|z| = 1$,
   - $\ker(\mathbf{U}_{\mathbf{I}-z\cdot\mathbf{W}}) \subseteq \ker(\widetilde{\mathbf{W}} - \mathbf{W}) \cap \ker((\widetilde{\mathbf{W}} - \mathbf{W})^\top)$ *and*
   - $\left\|\mathbf{U}^{+/2}_{\mathbf{I}-z\cdot\mathbf{W}}(\widetilde{\mathbf{W}} - \mathbf{W})\mathbf{U}^{+/2}_{\mathbf{I}-z\cdot\mathbf{W}}\right\| \leq \epsilon$

A proof of Lemma III.8 can be found in the full version of this paper [1].

IV. Approximating Cycle-Lifted Graphs and Powers

In this section we discuss how unit-circle spectral approximation allows us to approximate powers of random walk matrices of digraphs and a class of graphs we call *cycle-lifted graphs*, which play an essential role in our Eulerian Laplacian solver. Preservation under powering is a useful property for a definition of matrix approximation but even in the case of symmetric transition matrices, the original definition of spectral approximation does not guarantee this, as is seen in the following proposition.

Proposition IV.1. *For all rational $\epsilon \in (0,1)$, there exist undirected graphs with transition matrices $\widetilde{\mathbf{W}}, \mathbf{W}$ such that $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ but $\widetilde{\mathbf{W}}^2 \not\approx_c \mathbf{W}^2$ for any finite $c > 0$.*

Lemma IV.2. *Fix $\widetilde{\mathbf{W}}, \mathbf{W} \in \mathbb{C}^{n\times n}$. For any matrix $M \in \mathbb{C}^{n\times n}$, let $V_\lambda(M)$ denote the eigenspace of M of eigenvalue λ.*

1) *If $\widetilde{\mathbf{W}} \approx_c \mathbf{W}$ for a finite $c > 0$, then $V_1(\mathbf{W}) \subseteq V_1(\widetilde{\mathbf{W}})$. If $c < 1$, then $V_1(\mathbf{W}) = V_1(\widetilde{\mathbf{W}})$.*
2) *If $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_c \mathbf{W}$ for a finite $c > 0$, then for all $\lambda \in \mathbb{C}$ such that $|\lambda| = 1$, $V_\lambda(\mathbf{W}) \subseteq V_\lambda(\widetilde{\mathbf{W}})$ and if $c < 1$ then $V_\lambda(\mathbf{W}) = V_\lambda(\widetilde{\mathbf{W}})$.*

A proof of Lemma IV.2 can be found in the full version of this paper [1]. In previous work ([6], [38]), it was observed that for symmetric matrices, if $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ and $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$ then we do get $\widetilde{\mathbf{W}}^2 \approx_\epsilon \mathbf{W}^2$. Furthermore, when $\mathbf{W}$ is PSD we have that $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ implies $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$. Since $\mathbf{W}^2$ is trivially PSD, the above can be applied recursively to conclude that $\widetilde{\mathbf{W}}^{2^k} \approx_\epsilon \mathbf{W}^{2^k}$ for all positive integers k. However, we observe that analogous approximation guarantees do not hold for Eulerian graphs (or even regular digraphs).

Proposition IV.3. *For all rational $\epsilon \in (0,1)$, there exist regular digraphs with transition matrices $\widetilde{\mathbf{W}}, \mathbf{W}$ such that $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ and $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$ but $\widetilde{\mathbf{W}}^4 \not\approx_c \mathbf{W}^4$ for any finite c.*

Now we show that if a matrix is a unit-circle approximation of another, then all of their powers are as well (with small loss in approximation quality). In fact, we show something stronger, namely that their *cycle-lifted graphs* approximate each other.

Definition IV.4 (Cycle-Lifted Graph)**.** Let $\mathbf{C}_k$ denote the transition matrix of the k-vertex directed cycle. Given a graph $G = (V, E)$ on n vertices with transition matrix $\mathbf{W}$ the *cycle-lifted graph of length k*, $C_k(G)$, is a layered graph with k layers (numbered 1 to k) of n vertices each, where for every $i \in [k]$, there is an edge from vertex u in layer i to vertex v in layer $(i+1) \mod k$ with multiplicity ℓ if and only if (u, v) exists with multiplicity ℓ in G. That is, $C_k(G) = (V', E')$ with $V' = [k] \times V$ and $E' = \{((i, u), (i+1 \mod k, v) \colon (u, v) \in E\}$. The transition matrix of $C_k(G)$ is $\mathbf{C}_k \otimes \mathbf{W}$.

Theorem IV.5. *Fix $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^{n\times n}$ and let $\mathbf{C}_k$ be the transition matrix for the directed cycle on k vertices. Then $\mathbf{C}_k \otimes \widetilde{\mathbf{W}} \approx_\epsilon \mathbf{C}_k \otimes \mathbf{W}$ if and only if for all z* such that *$z^k = 1$, we have $z \cdot \widetilde{\mathbf{W}} \approx_\epsilon z \cdot \mathbf{W}$.*

Recall that unit-circle spectral approximation requires that for all $z \in \mathbb{C}$ with $|z| = 1$ we have $z \cdot \widetilde{\mathbf{W}} \approx_\epsilon z \cdot \mathbf{W}$. Theorem IV.5 then tells us that unit-circle spectral approximation implies approximations of the corresponding cycle-lifted graphs of *every length*.

Corollary IV.6. *Fix $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^{n\times n}$. If $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$ then for all positive integers k, $\mathbf{C}_k \otimes \widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{C}_k \otimes \mathbf{W}$.*

Theorem IV.5 allows us to reason about approximation under powering by observing that the kth power of a

matrix can be expressed in terms of the Schur complement of its cycle-lifted graph of length k. In [39], they showed that undirected spectral approximation is preserved under Schur complements. Here we show that the same is true of directed spectral approximation (with a small loss in approximation quality).

Theorem IV.7. *Fix $\mathbf{W}, \widetilde{\mathbf{W}} \in \mathbb{C}^n$ and suppose that $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ for $\epsilon \in (0, 2/3)$. Let $F \subseteq [n]$ such that $(\mathbf{I}_{|F|} - \mathbf{W}_{FF})$ is invertible and let $C = [n] \setminus F$. Then*

$$\mathbf{I}_{|C|} - \mathrm{Sc}(\mathbf{I}_n - \widetilde{\mathbf{W}}, C) \approx_{\epsilon/(1-3\epsilon/2)} \mathbf{I}_{|C|} - \mathrm{Sc}(\mathbf{I}_n - \mathbf{W}, C)$$

A proof of Theorem IV.7 is in the full version of this paper [1]. The expression in Theorem IV.7 has a natural interpretation in terms of random walks. Notice that

$$\mathbf{I}_{|C|} - \mathrm{Sc}(\mathbf{I}_n - \mathbf{W}, C) = \mathbf{W}_{CC} + \mathbf{W}_{CF}(\mathbf{I}_{|F|} - \mathbf{W}_{FF})^{-1}\mathbf{W}_{FC}.$$

When $\mathbf{W}$ is the transition matrix for a random walk, the right-hand side above can be interpreted as the transition matrix for the random walk induced by "short-cutting" walks that traverse through the set of vertices in F. In other words, walk behavior on C remains the same (the $\mathbf{W}_{CC}$ term) and walks that go from C to F (via $\mathbf{W}_{FC}$) can instantly take arbitrary length walks in F (the $(\mathbf{I}_{|F|} - \mathbf{W}_{FF})^{-1}$ term) before returning to C (via $\mathbf{W}_{CF}$). The theorem above says that spectral approximation is preserved under such "short-cutting".

Now we get the following corollary, which says that unit-circle approximation is preserved under powering.

Corollary IV.8. *Let $\mathbf{W}$, $\widetilde{\mathbf{W}}$ be the transition matrices of digraphs $G, \widetilde{G}$. If $\widetilde{\mathbf{W}} \overset{\circ}{\approx}_\epsilon \mathbf{W}$ then for all $k \in \mathbb{N}$ we have $\widetilde{\mathbf{W}}^k \overset{\circ}{\approx}_{\epsilon/(1-3\epsilon/2)} \mathbf{W}^k$.*

Interestingly, in the case of undirected graphs, Corollary IV.8 says that if $\widetilde{\mathbf{W}} \approx_\epsilon \mathbf{W}$ and $-\widetilde{\mathbf{W}} \approx_\epsilon -\mathbf{W}$ then all kth powers approximate one another (with small loss in approximation quality). This was not known (to the best of our knowledge) for any k other than powers of 2.

V. Derandomized Square of Regular Digraphs

In order to achieve a space-efficient and deterministic implementation of our algorithm, we need a way to efficiently approximate high powers of regular digraphs. Before defining such an operation, called derandomized square, we introduce two-way labelings and rotation maps.

Definition V.1 ([31], [41])**.** A *two-way labeling* of a d-regular directed multigraph G is a labeling of the edges in G such that

1) Every edge (u, v) has two labels in $[d]$, one as an outgoing edge from u and one as an incoming edge to v,
2) For every vertex v the labels of the d outgoing edges from v are distinct, as are the incoming edges to v.

In a two-way labeling, each vertex v has its own labeling from 1 to d for the d edges leaving it and its own labeling from 1 to d for the d edges entering it. Since every edge is incident to two vertices, each edge receives two labels, which may or may not be the same. It is convenient to specify a multigraph with a two-way labeling by a rotation map:

Definition V.2 ([20], [41])**.** Let G be a d-regular directed multigraph on n vertices with a two-way labeling. The *rotation map* $\mathrm{Rot}_G \colon [n]\times[d] \to [n]\times[d]$ is defined as follows: $\mathrm{Rot}_G(v, i) = (w, j)$ if the ith edge leaving vertex v leads to vertex w and this edge is the jth edge entering w.

Now we can define the derandomized square. Recall that the square of a graph G^2 is a graph on the same vertex set whose edges correspond to all walks of length 2 in G. The derandomized square picks out a pseudorandom subset of the walks of length 2 by correlating the 2 steps via edges on an expander graph.

Definition V.3 ([31])**.** Let G be a d-regular multigraph on n vertices with a two-way labeling. Let H be a c-regular undirected graph on d vertices. The *derandomized square* $G \circledS H$ is a $c \cdot d$-regular graph on n vertices with rotation map $\mathrm{Rot}_{G \circledS H}$ defined as follows: For $v_0 \in [n], i_0 \in [d]$, and $j_0 \in [c]$, we compute $\mathrm{Rot}_{G \circledS H}(v_0, (i_0, j_0))$ as

1) Let $(v_1, i_1) = \mathrm{Rot}_G(v_0, i_0)$
2) Let $(i_2, j_1) = \mathrm{Rot}_H(i_1, j_0)$
3) Let $(v_2, i_3) = \mathrm{Rot}_G(v_1, i_2)$
4) Output $(v_2, (i_3, j_1))$

In the square of a directed graph, for each vertex v, there exists a complete, uni-directional bipartite graph from the in-neighbors of v, to it's out-neighbors. This corresponds to a directed edge for every two-step walk that has v in the middle of it. A useful way to view the derandomized square is that it replaces each of these complete bipartite graphs with a uni-directional bipartite *expander*.

Definition V.4. Let $H = (V, E)$ be an undirected d vertex graph. Then $\mathrm{Bip}(H)$ is a bipartite graph with d vertices on each side and an edge (u, v) from vertex u on the left to vertex v on the right iff $(u, v) \in E$.

Note that since we're working with multigraphs, the incoming or outgoing neighbors to/from a vertex may form a multi-set rather than a set due to parallel edges. So when we say that a "copy" of $\mathrm{Bip}(H)$ exists from the in-neighbors of v to its out-neighbors, we mean that if we were to split all of the in-neighbors of v and out-neighbors of v into two sets of d distinct vertices, place the edges from $\mathrm{Bip}(H)$ across the sets, and then re-merge vertices that correspond to a repeat neighbor of v, then a copy of that resulting graph can be found across vertex v in $G \circledS H$.

We show the derandomized square of a regular digraph yields a unit-circle approximation to the true square.

Theorem V.5. *Let $G = (V, E)$ be a d-regular directed multigraph with random walk matrix $\mathbf{W}$. Let H be a c-*

regular expander with $\lambda(H) \leq \epsilon$ and let $\widetilde{\mathbf{W}}$ be the random walk matrix of $G \circledS H$. Then

$$\widetilde{\mathbf{W}} \overset{\circ}{\approx}_{2\cdot\epsilon} \mathbf{W}^2.$$

VI. Approximate Pseudoinverse for Cycle-Lifted Graphs

Let $\mathbf{I} - \mathbf{W}$ be the random-walk Laplacian of a strongly connected, aperiodic, regular digraph G. Our goal is to compute an accurate approximation of $(\mathbf{I} - \mathbf{W})^+$. To do this we consider the Laplacian of a cycle-lifted graph $\mathbf{L} = \mathbf{I}_{2^k n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$ for some positive integer k, and show how to compute an accurate approximation of $\mathbf{L}^+$, namely $\widetilde{\mathbf{L}^+}$. Then we show that under some conditions, an $n \times n$ projection of $\widetilde{\mathbf{L}^+}$ (specifically, $(\vec{\mathbf{1}}_{2^k} \otimes \mathbf{I}_n)^\top \widetilde{\mathbf{L}^+} (\vec{\mathbf{1}}_{2^k} \otimes \mathbf{I}_n)$) gives an accurate approximation for $(\mathbf{I} - \mathbf{W})^+$.

To estimate $\mathbf{L}^+$ we first show how to obtain a weak approximation to it. Then we show how to get an accurate approximation using Richardson iteration (see Lemma VI.2). The following is the main theorem we prove in this section. In this theorem, we only give sufficient conditions for having an approximate pseudo-inverse of the cycle-lifted graphs, and discuss an actual space-efficient algorithm for computing such a matrix in the full version of this paper [1].

Theorem VI.1. *Let $\mathbf{W}$ be the transition matrix of a strongly connected regular digraph with n vertices, $\epsilon \in (0, 1/(2k))$ and suppose we have a sequence of matrices $\mathbf{W} = \mathbf{W}_0, \ldots, \mathbf{W}_k$, such that*

$$\mathbf{W}_{i+1} \overset{\circ}{\approx}_{\epsilon} \mathbf{W}_i^2 \quad \forall 0 < i < k$$

and each $\mathbf{W}_i$ is a transition matrix of a strongly connected regular digraph. We use $\mathbf{W}_i$'s to define a sequence of $2^k n$ by $2^k n$ matrices $\mathbf{L}^{(i)}$. Then for $\mathbf{L} = \mathbf{I}_{2^k n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$ and $\widetilde{\mathbf{L}} \overset{\text{def}}{=} \mathbf{L}^{(k)}$, there exists a PSD matrix $\mathbf{F}$ such that $\|\mathbf{I}_{2^k n} - \widetilde{\mathbf{L}}^+ \mathbf{L}\|_{\mathbf{F}} \leq O(k^2 \epsilon)$ and $\mathbf{U}_{\mathbf{L}}/O(k) \preceq \mathbf{F} \preceq O(2^{2k} n^2 k^5)\mathbf{U}_{\mathbf{L}}$.

In the above theorem $\widetilde{\mathbf{L}}$ is defined in a way so that it has a nice LU factorization. This lets us efficiently compute $\widetilde{\mathbf{L}}^+$. Below we describe how we use Theorem VI.1. In the full version of this paper [1], we show for $k = O(\log n)$ how to space efficiently generate the $\mathbf{W}_i$'s and compute $\widetilde{\mathbf{L}}^+$.

The following lemma shows how we can obtain an accurate solver by boosting the precision of an approximate pseudo-inverse through the well-known method of preconditioned Richardson iteration [12], [17].

Lemma VI.2. *Given matrices $\mathbf{A}, \mathbf{B}, \mathbf{F} \in \mathbb{R}^{n \times n}$, such that $\mathbf{F}$ is PSD, and $\|\mathbf{I} - \mathbf{B}\mathbf{A}\|_{\mathbf{F}} \leq \alpha$ for some constant $\alpha > 0$. Let $\mathbf{P}_m = \sum_{i=0}^{m} (\mathbf{I} - \mathbf{B}\mathbf{A})^i \mathbf{B}$. Then*

$$\|\mathbf{I} - \mathbf{P}_m \mathbf{A}\|_{\mathbf{F}} \leq \alpha^{m+1}$$

Since we can obtain a reasonably good approximate pseudo-inverse for $\mathbf{L}$ via Theorem VI.1 and boost the quality of that approximation with Lemma VI.2, we can ultimately get a very accurate approximate pseudo-inverse. This is stated rigorously in the following corollary.

Corollary VI.3. *Given a transition matrix $\mathbf{W}$ of a regular digraph with n vertices, and $\delta \in (0, 1/2)$. Let $\mathbf{L} = \mathbf{I}_{2^k n} - \mathbf{C}_{2^k} \otimes \mathbf{W}$, and let $\widetilde{\mathbf{L}}^+$ be the approximate pseudo-inverse obtained from Theorem VI.1 by setting $\epsilon = \frac{1}{ck}$ for a large enough constant c. For $m = O\left(k + \log n + \log(\frac{1}{\delta})\right)$, and $\mathbf{P}_m = \sum_{i=0}^{m} (\mathbf{I} - \widetilde{\mathbf{L}}^+ \mathbf{L})^i \widetilde{\mathbf{L}}^+$, we have*

$$\|\mathbf{I} - \mathbf{P}_m \mathbf{L}\|_{\mathbf{U}_{\mathbf{L}}} \leq \delta$$

In the full version of the paper [1], we show how to obtain an approximate pseudo-inverse of the original Laplacian system, given an approximate pseudo-inverse for the cycle-lifted graph. To get an approximate pseudo-inverse of $\mathbf{L}$ we first compute an approximate LU factorization of it. This is done by repeatedly eliminating blocks of the cycle lifted graph containing every other vertex in the cycle.

Acknowledgements

We thank William Hoza for pointing out an error in an earlier version of Theorem IV.7. A.A. was supported by a Dantzig-Lieberman Operations Research Fellowship. J.K. was supported by NSF award 1565235. J.M. was supported by NSF grant CCF-1763299. J.P. was supported by Swiss National Science Foundation Grant #200021_182527. A.S. was supported by NSF CAREER Award CCF-1844855. S.V. was supported by NSF grant CCF-1763299 and a Simons Investigator Award.

References

[1] A. Ahmadinejad, J. Kelner, J. Murtagh, J. Peebles, A. Sidford, and S. Vadhan, "High-precision estimation of random walks in small space," CoRR, Tech. Rep. arXiv:1912.04524 [cs.CC], December 2019. [Online]. Available: https://arxiv.org/abs/1912.04524

[2] R. Aleliunas, R. M. Karp, R. J. Lipton, L. Lovász, and C. Rackoff, "Random walks, universal traversal sequences, and the complexity of maze problems," in *20th Annual Symposium on Foundations of Computer Science (San Juan, Puerto Rico, 1979)*. New York: IEEE, 1979, pp. 218–223.

[3] M. Saks and S. Zhou, "$\text{BP}_{\text{H}}\text{SPACE}(S) \subseteq \text{DSPACE}(S^{3/2})$," *Journal of Computer and System Sciences*, vol. 58, no. 2, pp. 376–403, 1999.

[4] M. E. Saks, "Randomization and derandomization in space bounded computation," in *IEEE Conference on Computational Complexity*, 1996, pp. 128–149.

[5] J. Murtagh, O. Reingold, A. Sidford, and S. Vadhan, "Deterministic approximation of random walks in small space," in *Proceedings of the 23rd International Conference on Randomization and Computation (RANDOM '19)*, ser. LIPIcs, D. Achlioptas and L. A. Végh, Eds., vol. 145. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 42:1–42:22. [Online]. Available: https://doi.org/10.4230/LIPIcs.APPROX-RANDOM.2019.42

[6] J. Murtagh, O. Reingold, A. Sidford, and S. P. Vadhan, "Derandomization beyond connectivity: Undirected laplacian systems in nearly logarithmic space," in *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, 2017, pp. 801–812. [Online]. Available: https://doi.org/10.1109/FOCS.2017.79

[7] D. A. Spielman and S.-H. Teng, "Nearly-linear time algorithms for graph partitioning, graph sparsification, and solving linear systems," in *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing*. ACM, 2004, pp. 81–90.

[8] I. Koutis, G. L. Miller, and R. Peng, "Approaching optimality for solving sdd systems. march 2010."

[9] ——, "A nearly-m log n time solver for sdd linear systems," in *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*. IEEE, 2011, pp. 590–598.

[10] 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 forty-fifth annual ACM symposium on Theory of computing*. ACM, 2013, pp. 911–920.

[11] Y. T. Lee and A. Sidford, "Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems," in *Foundations of Computer Science (FOCS), 2013 IEEE 54th Annual Symposium on*. IEEE, 2013, pp. 147–156.

[12] R. Peng and D. A. Spielman, "An efficient parallel solver for SDD linear systems," *STOC*, 2014.

[13] M. B. Cohen, R. Kyng, G. L. Miller, J. W. Pachocki, R. Peng, A. B. Rao, and S. C. Xu, "Solving SDD linear systems in nearly $m \log^{1/2} n$ time," in *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, 2014, pp. 343–352.

[14] R. Kyng, Y. T. Lee, R. Peng, S. Sachdeva, and D. A. Spielman, "Sparsified cholesky and multigrid solvers for connection laplacians," in *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*. ACM, 2016, pp. 842–850.

[15] R. Kyng and S. Sachdeva, "Approximate gaussian elimination for laplacians - fast, sparse, and simple," in *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, 2016, pp. 573–582.

[16] M. B. Cohen, J. Kelner, J. Peebles, R. Peng, A. Sidford, and A. Vladu, "Faster algorithms for computing the stationary distribution, simulating random walks, and more," in *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*. IEEE, 2016, pp. 583–592.

[17] M. B. Cohen, J. Kelner, J. Peebles, R. Peng, A. B. Rao, A. Sidford, and A. Vladu, "Almost-linear-time algorithms for markov chains and new spectral primitives for directed graphs," in *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 2017, pp. 410–419.

[18] M. B. Cohen, J. Kelner, R. Kyng, J. Peebles, R. Peng, A. B. Rao, and A. Sidford, "Solving directed laplacian systems in nearly-linear time through sparse lu factorizations," in *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2018, pp. 898–909.

[19] A. Ahmadinejad, A. Jambulapati, A. Saberi, and A. Sidford, "Perron-frobenius theory in nearly linear time: Positive eigenvectors, m-matrices, graph kernels, and other applications," in *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2019, San Diego, California, USA, January 6-9, 2019*, 2019, pp. 1387–1404.

[20] O. Reingold, L. Trevisan, and S. Vadhan, "Pseudorandom walks in regular digraphs and the RL vs. L problem," in *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC '06)*, 21–23 May 2006, pp. 457–466, preliminary version as *ECCC* TR05-22, February 2005.

[21] A. R. Klivans and D. van Melkebeek, "Graph nonisomorphism has subexponential size proofs unless the polynomial-time hierarchy collapses," *SIAM Journal on Computing*, vol. 31, no. 5, pp. 1501–1526 (electronic), 2002. [Online]. Available: http://dx.doi.org/10.1137/S0097539700389652

[22] N. Nisan, "Pseudorandom bits for constant depth circuits," *Combinatorica*, vol. 11, no. 1, pp. 63–70, 1991.

[23] J. Síma and S. Zák, "Almost k-wise independent sets establish hitting sets for width-3 1-branching programs," in *CSR*, ser. Lecture Notes in Computer Science, A. S. Kulikov and N. K. Vereshchagin, Eds., vol. 6651. Springer, 2011, pp. 120–133.

[24] P. Gopalan, R. Meka, O. Reingold, L. Trevisan, and S. Vadhan, "Better pseudorandom generators via milder pseudorandom restrictions," in *Proceedings of the 53rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '12)*. IEEE, 20–23 October 2012.

[25] R. Meka, O. Reingold, and A. Tal, "Pseudorandom generators for width-3 branching programs," in *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*. ACM, 2019, pp. 626–637.

[26] M. Braverman, A. Rao, R. Raz, and A. Yehudayoff, "Pseudorandom generators for regular branching programs," in *FOCS*. IEEE Computer Society, 2010, pp. 40–47.

[27] W. Hoza, E. Pyne, and S. Vadhan, "Pseudorandom generators for unbounded-width permutation branching programs," Unpublished manuscript, April 2020.

[28] M. Koucký, P. Nimbhorkar, and P. Pudlák, "Pseudorandom generators for group products: extended abstract," in *STOC*, L. Fortnow and S. P. Vadhan, Eds. ACM, 2011, pp. 263–272.

[29] A. De, "Pseudorandomness for permutation and regular branching programs," in *IEEE Conference on Computational Complexity*. IEEE Computer Society, 2011, pp. 221–231.

[30] T. Steinke, "Pseudorandomness for permutation branching programs without the group theory," Electronic Colloquium on Computational Complexity (ECCC), Tech. Rep. TR12-083, July 2012. [Online]. Available: http://eccc.hpi-web.de/report/2012/083/

[31] E. Rozenman and S. Vadhan, "Derandomized squaring of graphs," in *Proceedings of the 8th International Workshop on Randomization and Computation (RANDOM '05)*, ser. Lecture Notes in Computer Science, no. 3624. Berkeley, CA: Springer, August 2005, pp. 436–447.

[32] W. Hoza and D. Zuckerman, "Simple optimal hitting sets for small-success RL," in *59th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2018, Paris, France, October 7-9, 2018*, 2018, pp. 59–64. [Online]. Available: https://doi.org/10.1109/FOCS.2018.00015

[33] J. Gill, "Computational complexity of probabilistic Turing machines," *SIAM Journal on Computing*, vol. 6, no. 4, pp. 675–695, 1977.

[34] J. Simon, "On tape-bounded probabilistic Turing machine acceptors," *Theoretical Computer Science*, vol. 16, no. 1, pp. 75–91, 1981. [Online]. Available: https://doi-org.ezp-prod1.hul.harvard.edu/10.1016/0304-3975(81)90032-3

[35] A. Borodin, S. Cook, and N. Pippenger, "Parallel computation for well-endowed rings and space-bounded probabilistic machines," *Information and Control*, vol. 58, no. 1-3, pp. 113–136, 1983. [Online]. Available: https://doi-org.ezp-prod1.hul.harvard.edu/10.1016/S0019-9958(83)80060-6

[36] H. Jung, "Relationships between probabilistic and deterministic tape complexity," in *Mathematical foundations of computer science, 1981 (Štrbské Pleso, 1981)*, ser. Lecture Notes in Comput. Sci. Springer, Berlin-New York, 1981, vol. 118, pp. 339–346.

[37] D. A. Spielman and S.-H. Teng, "Spectral sparsification of graphs," *SIAM Journal on Computing*, vol. 40, no. 4, pp. 981–1025, 2011.

[38] D. Cheng, Y. Cheng, Y. Liu, R. Peng, and S.-H. Teng, "Spectral sparsification of random-walk matrix polynomials," *arXiv preprint arXiv:1502.03496*, 2015.

[39] G. L. Miller and R. Peng, "Approximate maximum flow on separable undirected graphs," in *Proceedings of the twenty-fourth annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, 2013, pp. 1151–1170.

[40] J. Batson, D. A. Spielman, N. Srivastava, and S.-H. Teng, "Spectral sparsification of graphs: theory and algorithms," *Communications of the ACM*, vol. 56, no. 8, pp. 87–94, 2013.

[41] O. Reingold, S. Vadhan, and A. Wigderson, "Entropy waves, the zig-zag graph product, and new constant-degree expanders," *Annals of Mathematics*, vol. 155, no. 1, January 2001.