Toward Derandomizing RL via Graph Connectivity

Jyun-Jie Liao, Wei-Kai Lin

December 8, 2018

1 Introduction

A long-standing problem in computational complexity is whether randomized algorithms are more powerful than deterministic algorithms. Compared to derandomizing **RP** or **BPP**, derandomizing log-space algorithm seems to be a simpler task. Notably, based on Nisan's pseudorandom generator (PRG) w.r.t. read-once branching program [Nis92], Saks and Zhou [SZ95] proved that $\mathbf{RL} \subseteq \mathrm{SPACE}(\log^{3/2} n)$. However, this bound has not been improved for decades. In 2005, Reingold [Rei05] showed how to fully derandomize undirected s-t connectivity, which was the most famous problem in **RL** not known to be in **L**. Although this does not give any new deterministic space bound for **RL**, this is considered as a hopeful step toward proving $\mathbf{RL} = \mathbf{L}$ because of the close relation between log-space complexity class and s-t connectivity.

In this survey, we will introduce the technique of Reingold [Rei05] to solve undirected s-t connectivity, and show some following attempts toward proving $\mathbf{RL} = \mathbf{L}$. We will first show how Reingold, Trevisan and Vadhan [RTV06] generalize Reingold's algorithm to regular directed graph. Then we will introduce a new **RL**-complete connectivity problem, poly-mixing s-t connectivity, suggested in the same paper, and discuss some possible approaches suggested in [RTV06, CRV07, MRSV17] to generalize Reingold's result to poly-mixing directed graph.

2 Solving Undirected S-T Connectivity in L

The problem of undirected s-t connectivity is defined as follows. The input is an undirected graph $G = (V, E)$ and two vertices $s, t \in V$, and the output is "accept" iff there exists a path from s to t on G. It has been shown that undirected s-t connectivity is in **RL** in the work of Adelliunas et al. [AKL+79] early in 1979, and then Lewis and Papadimitriou [LP82] show that it is complete in the class **SL**. This section presents the construction of Reingold [Rei05] that proves that undirected s-t connectivity is in **L**, which implies that $\mathbf{SL} = \mathbf{L}$.

The intuition of Reingold is to transform the input G into a corresponding graph $G_{\exp}$ that $G_{\exp}$ is "more connected" and it is easier to solve s-t connectivity on $G_{\exp}$. Let $N = |V|$ and $|E|$ be $O(N^2)$. Concretely, we want a reduction such that (1) each connected component C in G is transformed into a connected component in $G_{\exp}$, (2) each connected component in $G_{\exp}$ is a $(\mathrm{poly}(N), O(1), 1/2)$-expander for some polynomial $\mathrm{poly}(\cdot)$, (3) the adjacency of any vertex in $G_{\exp}$ is computable in **L**. With constant expansion, the diameter of any connected component of $G_{\exp}$ is $O(\log N)$, and any connected vertex is reachable in a path of $O(\log N)$ steps. Because the degree of $G_{\exp}$ is constant, it takes only constant bits to store a step in any path. Putting together, we can deterministically traverse all connected vertices on G_{exp} by enumerating all paths of length $O(\log N)$ and by the adjacency is computable in **L**. To solve s-t connectivity on G, it suffices to transform vertices s, t into vertices s', t' on G_{exp} and then to solve the connectivity of s', t' on G_{exp} by traversing all connected vertices from s'. Hence, the main challenge is to construct the required transformation.

The intuition of the reduction is that any connected graph has a small vertex expansion as any strict subset $S \subset V$ always has a neighboring set consists of at least one more vertex than S. Then, informally, the spectral gap of the connected graph is at least $\frac{1}{\text{poly}(N)}$. Then, performing *powering* on the graph itself increases the spectral gap, and it takes only a poly(N) powering, or equivalently $O(\log N)$ squaring, to achieve a constant spectral gap. However, the powering increases also the degree, which is undesirable. To solve this, Reingold uses *zig-zag product* to bring the degree down to a constant while the spectral gap is reduced only by a small fraction, and showing that zig-zag product still works for such polynomially-small spectral gap is a new technical result of Reingold. The zig-zag product increases also the number of vertices, but it is applied with a constant-sized expander and thus the increment is only a constant factor, and the total increment is polynomial in N as zig-zag product is applied $O(\log N)$ times.

In the following, we define the graph operations and notations with some text copied verbatim from [RTV06].

Definition 2.1 (Rotation map). *For every vertex v in a D-regular graph G, label every incident edge of v with a distinct label in $[D]$. Then define the rotation map $Rot_G(u, i) = (v, j)$ if the edge from u with label i is an incident edge to v with label j.*

Definition 2.2 (Powering). *Let G be a D-regular graph given by rotation map $Rot_G : [N] \times [D] \mapsto [N] \times [D]$. The t-th power of G is the graph G^t with rotation map is given by $Rot_{G^t} : [N] \times [D]^t \mapsto [N] \times [D]^t$ defined by $Rot_{G^t}(v_0, (k_1, \ldots, k_t)) = (v_t, (l_t, \ldots, l_1))$, where these values are computed via the rule $(v_i, l_i) = Rot_G(v_{i-1}, k_i)$ (and if any of these evaluations yield $\perp$, then the final output is also $\perp$).*

Definition 2.3 (Zig-zag product [RVW00]). *If G is a D_1-regular graph on N vertices with rotation map $Rot_G : [N] \times [D_1] \mapsto [N] \times [D_1]$ and H is a D_2-regular graph on D_1 vertices with rotation map $Rot_H : [D_1] \times [D_2] \mapsto [D_1] \times [D_2]$, then their zig-zag product $G Ⓩ H$ is defined to be the graph on $[N] \times [D_1]$ vertices whose rotation map $Rot_{GⓏH} : ([N] \times [D_1]) \times [D_2]^2 \mapsto ([N] \times [D_1]) \times [D_2]^2$ is as follows:*
$Rot_{GⓏH}((v, k), (i, j))$:
let $(k', i') = Rot_H(k, i)$, let $(w, l') = Rot_G(v, k')$, let $(\ell, j') = Rot_H(l', j)$, and then output $((w, \ell), (j', i'))$.

For undirected graph G, we denote by $\lambda(G)$, the second largest eigenvalue (in absolute value) of G's normalized adjacency matrix, and we say G is an (N, D, λ)-expander iff G consists of N vertices, is D-regular, and has $\lambda(G) \leq \lambda$.

Transforming G to G_{exp}. Let H be a $(D_e^{80}, D_e, 1/2)$-expander for some constant D_e. Given an undirected graph $G = (V, E)$, the first step is to transform it into a regular graph G_{reg}. To do it, for every vertex $v \in V$,we expand it into a cycle C_v of $d(v)$ vertices in G_{reg} (where $d(v)$ is the vertex degree of v), and for every edge $(u, v) \in V$ that is the i-th edge of u and the j-th edge of v, we add an edge that connects the i-th vertex in C_v and the j-th vertex in C_u. The expanded graph is then augmented with $D_e^{80} - 3$ self loops. Let the result be $G_{\rm reg}$, a D_e^{80}-regular graph. Let $G_0 := G_{\rm reg}$, and define G_i iteratively as

$$G_{i+1} := (G_i ⓩ H)^{40}.$$

Let $\ell = 4 \log N + O(1)$, and $G_{\exp} := G_\ell$.

Analysis. The transformation from G into $G_{\exp}$ consists of expanding vertices into cycles, performing zig-zag product with H, and raising to the power of 40, where all three operations transform a connected component into another component. Hence, every connected component in G is a connect component in $G_{\exp}$.

To show that $G_{\exp}$ is a $(\mathrm{poly}(N), O(1), 1/2)$-expander, observe that the number of vertices is $N \cdot (D_e^{80})^\ell = \mathrm{poly}(N)$. The vertex degree of $G_i ⓩ H$ is D_e^2 for all i, and thus G_{i+1} is $(D_e^2)^{40} = O(1)$-regular, which implies $G_{\exp}$ is $O(1)$-regular. Hence, the remaining is to show that $\lambda(G_{\exp}) \le 1/2$, and the following lemmas will be used.

Lemma 2.4 ([AS00]). *For every N vertices, D-regular, connected, non-bipartite graph G, it holds that $\lambda(G) \le 1 - \frac{1}{DN^2}$.*

Lemma 2.5. *If G is an (N, D, λ)-expander, then G^t is an (N, D^t, λ^t)-expander.*

The above two is well-known and we omit the proof. The proof of the following lemma of zig-zag product is deferred to Theorem 3.7, a more general result on regular but directed graph.

Lemma 2.6 ([RVW00]). *Let G, H be undirected graphs. If $\lambda(G) \le 1-\gamma_1$ and $\lambda(H) \le 1-\gamma_2$, then $\lambda(G ⓩ H) \le 1 - \gamma_1\gamma_2^2$.*

By Lemma 2.4, it holds that $\lambda(G_0) \le 1 - \frac{1}{D_e^{80} N^2}$ as G_0 consists of N vertices and is D_e^{80}-regular. We claim that for all i, if $\lambda(G_i) > 1/2$, then $\lambda(G_{i+1}) \le (\lambda(G_i))^2$; otherwise if $\lambda(G_i) \le 1/2$, then $\lambda(G_{i+1}) \le 1/2$. Let $\lambda(G_i) = 1 - \gamma$ for some $\gamma < 1/2$. Then, $\lambda(G_i ⓩ H) \le 1 - \gamma/4$ by Lemma 2.6, and $\lambda(G_{i+1}) \le (1-\gamma/4)^{40}$ by Lemma 2.5. By taking the derivative, it holds that $(1-\gamma/4)^{40} \le (1-\gamma)^2$ for $\gamma \in [1/2, 1]$, and thus $\lambda(G_{i+1}) \le (\lambda(G_i))^2$. If $\lambda(G_i) \le 1/2$, then $\lambda(G_i ⓩ H) \le 7/8$ and then $\lambda(G_{i+1}) \le (7/8)^{40} \le 1/2$ as well. With such decrements in $\lambda(G_i)$, it follows that $\lambda(G_\ell) \le (\lambda(G_0))^{2^\ell} \le 1/2$ as desired.

Finally, we check that the adjacency of vertex in $G_{\exp}$ can be computed in **L**. Observe that for any i, to compute the adjacency of a vertex in G_{i+1} can be done by taking a constant number of steps, where each step follows an edge of either H or G_i. Following an edge of H takes constant bits; following an edge of G_i needs to compute the adjacency in G_i, but the space of the computation on G_i can be released and only constant bits is stored. Hence, the space is logarithmic in N and the computation is deterministic by the construction.

3 Generalization to Regular Directed Graph

Reingold's technique shows that undirected s-t connectivity algorithm can be derandomized space-efficiently, and hence **SL** = **L**. If Reingold's technique can be generalized to the configuration graph of every **RL** algorithm, it will imply **RL** = **L**. Reingold, Trevisan and Vadhan [RTV06] make the first step by generalizing the technique to directed graphs in which every vertex has in-degree and out-degree D (a.k.a. D-regular digraph). Their algorithm for regular digraph is basically the same as Reingold's algorithm, but one needs to define the analog of spectral expansion and zig-zag product in regular digraph, show that the input graph has non-negligible spectral gap, and prove that zig-zag product for regular digraph also preserves spectral gap.

For spectral expansion, the eigenvalue of a non-symmetric is not necessarily real, and the eigenvalue of stationary distribution does not necessarily have the largest absolute value. Therefore the second largest (in absolute value) eigenvalue (λ_2) is not a good way to define the spectral expansion of directed graph. In [RTV06] they used the following definition introduced by Mihail [Mih89] and Fill [Fil91], which is equivalent to λ_2 for directed graph:

Definition 3.1. *Let M be a Markov Chain and π be a stationary distribution of M (i.e. $M\pi = \pi$). Then*

$$\lambda_\pi(M) = \max_{x:\sum_i x_i=0} \frac{\|Mx\|_\pi}{\|x\|_\pi}$$

, where $\|x\|_\pi := \sum_{i:\pi_i>0}(x_i^2/\pi_i)$.

In this definition x can be considered as the deficiency of a distribution from the stationary distribution π. It is not hard to prove the following lemmas:

Lemma 3.2 ([RTV06])**.** *Let π be a stationary distribution of a Markov chain M. For every distribution v s.t. $supp(v) \subseteq supp(\pi)$, and any positive integer k,*

$$\left\|M^k v - \pi\right\|_\pi \leq \lambda_\pi(M)^k \|v - \pi\|_\pi.$$

In particular, for every vertices $s, t \in supp(\pi)$, a k-step random walk from s reaches t with probability at least $\pi(t) - \lambda_\pi(M)^k \cdot \sqrt{\pi(t)/\pi(s)}$.

Corollary 3.3. *Let s, t be two vertices in an n-vertex graph G. If there's a stationary distribution π s.t. $\pi(s), \pi(t) > 1/poly(n)$ and $\lambda_\pi(M) = O(1)$, there exists a path of length $O(\log n)$ from s to t.*

Same as Reingold's algorithm, if for every given graph G we can construct a graph G_{exp} satisfying the property above, we can solve s-t connectivity by enumerating every path. Next we show that all the properties we need for undirected graph in Reingold's algorithms also hold for regular digraph. For connected regular digraph, it is not hard to see that the stationary distribution is the uniform distribution, so we omit π in this case. Using a reduction to undirected graph Reingold, Trevisan and Vadhan [RTV06] proved the following lemma:

Lemma 3.4. *Let G be a connected, D-regular digraph on N vertices, and every vertex has at least αD self-loops. Then $\lambda(G) < 1 - \Omega(\alpha/DN^2)$.*

It is also not hard to see that $\lambda(G^t) \leq \lambda(G)^t$. The remaining part is to define zig-zag product and show that it also preserves spectral gap. We only show the definition of rotation map, and the remaining definition of zig-zag product is the same.

Definition 3.5. *For every vertex v in a D-regular digraph G, label every incoming edge of v with a distinct label in $[D]$. Similarly label every outgoing edge with a distinct label in $[D]$. Then define the rotation map $Rot_G(u, i) = (v, j)$ if the outgoing edge from u with label i is an incoming edge to v with label j.*

To show that the zig-zag product for regular digraph also preserves spectral gap, they give a new proof inspired by the analysis of "derandomized squaring" by Rozenman and Vadhan [RV05], which is also much simpler than the proof in [RVW00]. The proof is based on the following lemma (we omit the proof here but it is also very simple):

Lemma 3.6 ([RV05])**.** *Let M be the Markov chain corresponding to the random walk on a regular digraph G with N-vertex. Let J be an $N \times N$ matrix such that every entry of J is $1/N$. Then for every λ such that $\lambda(M) \leq \lambda$, $M = (1-\lambda)J + \lambda E$ for some matrix E with norm at most* 1.

Now we are ready to prove the key theorem in [RTV06].

Theorem 3.7. *Suppose $\lambda(G) \leq 1 - \gamma_1$ and $\lambda(H) \leq 1 - \gamma_2$. Then $\lambda(G \textcircled{z} H) \leq 1 - \gamma_1 \gamma_2^2$.*

Proof. Let $\tilde{H} = \hat{H} \otimes I_N$ where $\hat{H}$ is the normalized adjacency matrix of H. By Lemma 3.6 we can write $\hat{H} = \gamma_2 J + (1-\gamma_2)E$. Let $\tilde{J} = J \otimes I_N$ and $\tilde{E} = E \otimes I_N$. Let $\hat{G}$ be the normalized adjacency matix of G, and $\tilde{G}$ be the permutation matrix such that $\tilde{G}_{(u,i),(v,j)} = 1$ if $\mathrm{Rot}_G(u,i) = (v,j)$. By definition the Markov chain M of random walk on $G \textcircled{z} H$ is $\tilde{H}\tilde{G}\tilde{H}$. We can rewrite M as $\gamma_2^2 \tilde{J}\tilde{G}\tilde{J} + (1-\gamma_2^2)\tilde{F}$ for some matrix $\tilde{F}$ with norm at most 1. Observe that $\tilde{J}\tilde{G}\tilde{J} = J \otimes \hat{G}$. It is not hard to prove $\lambda(J \otimes \hat{G}) \leq \lambda(G)$. Hence we can conclude that

$$\lambda(G \textcircled{z} H) \leq \gamma_2^2(1-\gamma_1) + (1-\gamma_2^2) = 1 - \gamma_1\gamma_2^2.$$

□

Note that the proof above does not require that G and H are undirected. With all these properties we can show that Reingold's construction also converts a regular digraph to an expander, and hence one can find a s-t path in log space.

4 Toward Derandomizing RL

To deranomize RL, people hope to further generalize Reingold's algorithm to the configuration graph of RL problems. By definition, in the configuration graph of an YES instance, a random walk from starting state will reach the accept state in polynomial steps with probability at least 1/2. However, a *local* guarantee like this doesn't seem to be a good fit to Reingold's approach since spectral expansion is a *general* property. Therefore, Reingold, Trevisan and Vadhan [RTV06] suggested another RL-complete s-t connectivity problem:

Definition 4.1 (Poly-Mixing S-T Connectivity [RTV06])**.** *Given $(G, s, t, 1^k)$ with the following promise, check whether there exists a path in G from s to t.*

- ***YES**: There exists a stationary distribution π s.t. $\pi(s), \pi(t) > 1/k$ and $\lambda_\pi(G) > 1 - 1/k$.*
- ***NO**: There does not exist a path from s to t.*

In other word, this promise problem guarantees that the strongly connected component containing s has non-negligible spectral gap (i.e. polynomial mixing time), and the stationary distribution on G has non-negligible weight on s and t. Note that these guarantees are true for undirected graph and regular digraph with self-loop.

Theorem 4.2 ([RTV06]). *Poly-Mixing S-T Connectivity is RL-complete.*[1]

Proof. (Sketch) For a problem in **RL**, consider its corresponding probabilistic Turing machine. W.l.o.g. assume that it halts after exactly ℓ steps. Construct its configuration graph similar to the standard way, but also include the timestamp in the configurations. The configuration graph is an ℓ-layer graph, where the outgoing edges from vertices in the i-th layer always go to the $(i+1)$-th layer. Next, for every vertex in the ℓ-th layer, add two edges from it to the starting vertex s. Then add two self-loops to every vertex. Now we get a 4-outregular graph. It is not hard to verify that a stationary distribution π of this graph can be sampled as follows. Randomly sample a layer p. Run p steps in the Turing machine, and output the configuration vertex. By construction, $\pi(s) = 1/\ell$. By definition of **RL**, $\pi(t) > 1/2\ell$ given a YES instance. Moreover the spectral expansion $\lambda_\pi \leq 1 - 1/72\ell^2$. The bound is proved by computing the conductance in [RTV06] and we omit the proof here. The intuition is that a random walk starting from every vertex will pass through s in $O(\ell)$ steps with high probability, and the random walk starting from s also converges fast. In conclusion we can choose $k = 72\ell^2$, which is polynomial in input length. □

By Theorem 4.2, if there exists a log-space reduction from poly-mixing graph to regular digraph, then **RL** = **L**. Whether such reduction exists remains open, but in [RTV] a candidate reduction is given as follows. Suppose the given graph G has N vertices, D outgoing edges from each vertex, and has stationary distribution π.

1. Choose a large enough number $N' = \text{poly}(N)$, and replace each vertex u with a "cloud" C_u which consists of $N'\pi(u)$ vertices.
2. Choose a large enough number $D' = \text{poly}(N)$. For every edge $(u \to v)$ in the original graph, construct $D'/|C_v|$ edges from every vertex in C_u to every vertex in C_v.
3. Find a c_s to C_t path in the new graph. The transition between clouds in this path is an s-t path in the original graph.

If the construction is "perfect", i.e. $N'\pi(u)$ is integer for every vertex u and D' is a multiple of $|C_v|$ for every vertex v, it is not hard to verify that the new graph is regular, and the mixing time is preserved. Usually the perfect construction does not exist, but $\pi(s), \pi(t) > 1/\text{poly}(N)$ guarantees that s and t will not vanish even if there is some error in the construction. However, we don't know how to compute the stationary distribution in log space. There are two possible approaches introduced in [RTV06] and [CRV07], but neither of them is fully solved.

Oblivious Approach. In the construction, the size of each cloud is determined based on the stationary distribution. However, given a vertex in C_u and an outgoing edge labeled with $(i, j) \in [D] \times [D']$, we know it goes to the cloud C_v where v is the i-th neighbor of u in the original graph. Hence even if we don't know the size of each cloud, we know how to translate a sequence of labels in the new graph back to a path in the original graph. Reingold, Trevisan and Vadhan [RTV06] suggested that we can reduce the problem to the construction of a "pseudorandom walk generator":

Definition 4.3 (Pseudorandom Walk Generator [RTV06]). *We say an algorithm is a pseudorandom walk generator if given N, D, γ, δ, it can generate a pseudorandom sequence*

[1] Technically, promise-**RL** complete.

$I = (i_1, \ldots, i_\ell) \in [D]^\ell$ *of length* $\ell = \mathrm{poly}(ND/\delta\gamma)$ *with seed length* $O(\log(ND/\delta\gamma))$ *in space* $O(\log(ND/\delta\gamma))$ *s.t. for* every $(N, D, 1-\gamma)$ *graph with arbitrary edge label,* every *vertex* v*, and every vertex set* T *with density* δ*, a walk from* v *following edge label* I *visit* T *with probability at least* $1/\mathrm{poly}(ND/\delta\gamma)$*.*

This is called an oblivious approach since it produces a good walk without knowing the graph structure. They also showed that Reingold's algorithm generates a pseudorandom walk for a graph with *consistent labeling.* Recall that when translating a path (specified by a sequence of label) in $G Ⓩ H$ back to a path in G, we need to know H and the rotation map of G. A consistent labeling is a labeling of outgoing edges which allows us to assume $\mathrm{Rot}(u, i) = (v, i)$ for every edge $(u \to v)$ with label i on u. Since H is fixed, such assumption fix the translation of path in $G Ⓩ H$ to translation to G. Moreover, for every graph with constant spectral expansion, a sequence of uniformly random labels of logarithmic length always satisfies our requirement. Therefore, given parameters N, D, γ, δ we can compute the size N_{exp} of the expander constructed with Reingold's algorithms, generate a random label sequence of length $O(\log N_{\mathrm{exp}})$, and translate it back to a walk in a (N, D) graph.

Explicit Approach. Chung, Reingold and Vadhan [CRV07] suggested another approach by approximating the stationary distribution. They showed that there exists a polynomial p such that if we can approximate the stationary distribution of the given graph within error $1/p(n)$ in log space, we can solve s-t connectivity. The reduction from a poly-mixing graph given its stationary distribution to a regular digraph is the same as above. More precisely, they use the above construction to get an "approximately regular" graph, convert the graph to a consistently labeled graph (which is possible when the graph is *explicitly* given), and apply the pseudorandom walk generator mentioned above. Recently, Murtagh, Reingold, Sidford and Vadhan [MRSV17] made a step toward this approach. They use the "derandomized square" of Rozenman and Vadhan [RV05] to derandomize a approximate Laplacian solver by Peng and Spielman [PS14] in $O(\log n \log\log(n/\varepsilon))$ space. Since the time-efficient Laplacian solver has recently been extended to Eulerian digraph [CKP+16] and used to obtain stationary distribution for arbitrary poly-mixing digraph [CKP+17], it might be possible to extend this algorithm to prove $RL \subseteq SPACE(O(\log n \log\log n))$.

References

[AKL+79] Romas Aleliunas, Richard M. Karp, Richard J. Lipton, Laszlo Lovasz, and Charles Rackoff. Random walks, universal traversal sequences, and the complexity of maze problems. In *Proceedings of the 20th Annual Symposium on Foundations of Computer Science*, SFCS '79, pages 218–223, Washington, DC, USA, 1979. IEEE Computer Society.

[AS00] Noga Alon and Benny Sudakov. Bipartite subgraphs and the smallest eigenvalue. *Comb. Probab. Comput.*, 9(1):1–12, January 2000.

[CKP+16] Michael B. Cohen, Jonathan A. Kelner, John Peebles, Richard Peng, Aaron Sidford, and Adrian Vladu. Faster algorithms for computing the stationary distribution, simulating random walks, and more. In Irit Dinur, editor, *IEEE 57th Annual Symposium on Foundations of Computer Science, FOCS 2016, 9-11 October 2016, Hyatt Regency, New Brunswick, New Jersey, USA*, pages 583–592. IEEE Computer Society, 2016.

[CKP+17] Michael B. Cohen, Jonathan A. Kelner, John Peebles, Richard Peng, Anup B. Rao, Aaron Sidford, and Adrian Vladu. Almost-linear-time algorithms for markov chains and new spectral primitives for directed graphs. In Hamed Hatami, Pierre McKenzie, and Valerie King, editors, *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017*, pages 410–419. ACM, 2017.

[CRV07] Kai-Min Chung, Omer Reingold, and Salil P. Vadhan. S-T connectivity on digraphs with a known stationary distribution. In *22nd Annual IEEE Conference on Computational Complexity (CCC 2007), 13-16 June 2007, San Diego, California, USA*, pages 236–249. IEEE Computer Society, 2007.

[Fil91] James Allen Fill. Eigenvalue bounds on convergence to stationarity for nonreversible markov chains, with an application to the exclusion process. *The annals of applied probability*, pages 62–87, 1991.

[LP82] Harry R. Lewis and Christos H. Papadimitriou. Symmetric space-bounded computation. *Theoretical Computer Science*, 19(2):161 – 187, 1982.

[Mih89] Milena Mihail. Conductance and convergence of markov chains-a combinatorial treatment of expanders. In *30th Annual Symposium on Foundations of Computer Science, Research Triangle Park, North Carolina, USA, 30 October - 1 November 1989*, pages 526–531. IEEE Computer Society, 1989.

[MRSV17] Jack Murtagh, Omer Reingold, Aaron Sidford, and Salil P. Vadhan. Derandomization beyond connectivity: Undirected laplacian systems in nearly logarithmic space. In Chris Umans, editor, *58th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2017, Berkeley, CA, USA, October 15-17, 2017*, pages 801–812. IEEE Computer Society, 2017.

[Nis92] Noam Nisan. Pseudorandom generators for space-bounded computation. *Combinatorica*, 12(4):449–461, 1992.

[PS14] Richard Peng and Daniel A. Spielman. An efficient parallel solver for SDD linear systems. In David B. Shmoys, editor, *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 333–342. ACM, 2014.

[Rei05] Omer Reingold. Undirected st-connectivity in log-space. In Harold N. Gabow and Ronald Fagin, editors, *Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, May 22-24, 2005*, pages 376–385. ACM, 2005.

[RTV06] Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Pseudorandom walks on regular digraphs and the RL vs. L problem. In Jon M. Kleinberg, editor, *Proceedings of the 38th Annual ACM Symposium on Theory of Computing, Seattle, WA, USA, May 21-23, 2006*, pages 457–466. ACM, 2006.

[RV05] Eyal Rozenman and Salil P. Vadhan. Derandomized squaring of graphs. In Chandra Chekuri, Klaus Jansen, José D. P. Rolim, and Luca Trevisan, editors, *Approximation, Randomization and Combinatorial Optimization, Algorithms and*

Techniques, 8th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems, APPROX 2005 and 9th InternationalWorkshop on Randomization and Computation, RANDOM 2005, Berkeley, CA, USA, August 22-24, 2005, Proceedings, volume 3624 of *Lecture Notes in Computer Science*, pages 436–447. Springer, 2005.

[RVW00] Omer Reingold, Salil P. Vadhan, and Avi Wigderson. Entropy waves, the zig-zag graph product, and new constant-degree expanders and extractors. In *41st Annual Symposium on Foundations of Computer Science, FOCS 2000, 12-14 November 2000, Redondo Beach, California, USA*, pages 3–13. IEEE Computer Society, 2000.

[SZ95] Michael E. Saks and Shiyu Zhou. RSPACE(S) \subseteq dspace($s^{3/2}$). In *36th Annual Symposium on Foundations of Computer Science, Milwaukee, Wisconsin, USA, 23-25 October 1995*, pages 344–353. IEEE Computer Society, 1995.