Lifted Markov Chains for Fast Linear Computation

Kyomin Jung
Mathematics, MIT
kmjung@mit.edu

Devavrat Shah
EECS, MIT
devavrat@mit.edu

Jinwoo Shin
Mathematics, MIT
jinwoos@mit.edu

Abstract

We consider design of fast and robust iterative algorithm for distributed linear computation based on classical linear update method. The design of such an algorithm involves finding transition matrix $P$ of a Markov chain on network graph $G$ with fast mixing time and small size (number of non-zero entries in $P$). We present a novel method, which we call pseudo-lifting to construct such a desirable $P$ starting from a given matrix $Q$ (say, that obtained by Metropolis-Hastings rule). Under thus constructed $P$, the total distributed operations for linear computation is $\tilde{O}((|E| + Dn)D)$, where $D$ is the diameter of $G$, and $n$ is the number of nodes. This construction works for any graph and does not utilize any structural graph properties. Next, we present a hierarchical construction that cleverly utilizes the geometry of the graph to obtain $P$ with much smaller computation cost. Specifically, for graphs with doubling dimension $\rho$, it takes $\tilde{O}(D^2 n^{1 - \frac{1}{1+\rho}})$ total operations.

Our results imply an explicit construction of a Markov chain with fastest possible mixing time, of order of the graph diameter, on any graph using the pseudo-lifting – this should be of interest in its own right.

I. Introduction

Consider the following computation problem of linear computation. Given a connected network graph $G = (V,E)$, where $V = \{1, 2, \ldots, n\}$, each node $i \in V$ has a value $x_i \in \mathbb{R}$. Then our goal is to compute a linear combination of $x = (x_i)$ only by communications between adjacent nodes:

$$\langle x, \pi \rangle = \sum_i \pi_i x_i,$$

where $\pi = (\pi_i)$ are some constants such that $\pi_i \geq 0$ and $\sum_i \pi = 1$. This problem arises in many applications such as distributed estimation [22], distributed spectral decomposition [14], estimation and distributed data fusion on ad-hoc networks [18], distributed sub-gradient method for eigenvalue maximization [4], inference in Gaussian graphical models [19], and coordination of autonomous agents [12].

Thus designing distributed algorithm for fast computation of this problem is important. A popular and quite simple approach for this computation is a method based on linear iterations [23] as follows. Suppose we are given with a matrix $P = [P_{ij}] \in \mathbb{R}^{n \times n}$ such that (i) $P$ is graph conformant to $G$, i.e. $P_{ij} > 0$ only if $(i,j) \in E$, (ii) $P$ is a transition matrix i.e. for all $i \in V$, $\sum_j P_{ij} = 1$, (iii) $P$ has $\pi$ as its stationary distribution\(^1\), i.e.

$$\pi^T P = \pi^T.$$

The linear iteration algorithm is described as follows. Initially each node $i \in V$ has estimate of $\langle x, \pi \rangle$, say $y_i(0) = x_i$. At time $t = 1, 2, \ldots$ for each edge $(i,j)$ of $G$, node $i$ sends value $P_{ji}y_i(t)$ to node $j$. Then each node $j$ sums up the values received as its estimate at time $t + 1$, that is

$$y_j(t + 1) = \sum_{i=1}^{n} P_{ji}y_i(t).$$

Under the condition that $P$ is ergodic, i.e. $P$ is connected and aperiodic, it is known that [23]

$$\lim_{t \to \infty} y(t) = \lim_{t \to \infty} P^t x = \langle x, \pi \rangle 1,$$

where $1 = [1]$.

\(^1\)In this paper, $P$ can be (and will be) thought of as transition probability matrix of a Markov chain with state space $V$; therefore use of terminology stationary distribution instead of fixed point or eigenvector with eigenvalue 1.
Specifically define $\varepsilon$-computation time $T_\varepsilon(P)$ as follows:

$$T_\varepsilon(P) = \inf \left\{ t \mid \forall x \in \mathbb{R}^n, \max_{i \in V} \left| \frac{(P^t x)_i - \langle x, \pi \rangle}{\langle x, \pi \rangle} \right| \leq \varepsilon \right\}. \tag{2}$$

That is, $T_\varepsilon(P)$ is the first time $t$ so that for all $x \in \mathbb{R}^n$,

$$\frac{\|P^t x - \langle x, \pi \rangle 1\|_\infty}{\|\langle x, \pi \rangle 1\|_\infty} \leq \varepsilon.$$ 

Now consider the Markov chain whose transition matrix is given by $P$. The quantity $T_\varepsilon(P)$ is well known to be related to the mixing time $\mathcal{H}(P)$ of the Markov chain, which is defined in Section II-A. More precisely in Section II-D, we prove that

$$T_\varepsilon(P) = O \left( (\mathcal{H}(P) \log \frac{1}{\varepsilon \pi_0}) \right), \tag{3}$$

where $\pi_0 = \min_{i \in V} \pi_i$. Now the total number of operations of the linear iteration to obtain approximation of $\langle x, \pi \rangle$ scales like

$$C(\varepsilon, P) := T_\varepsilon(P) \times E(P), \tag{4}$$

where $E(P) = \{(i, j) \in E : P_{ij} > 0\}$. This is because in each time, each edge $(i, j)$ such that $P_{ij} > 0$ performs an exchange of values. Therefore the main goal of the paper is to minimize $C(\varepsilon, P)$ by designing appropriate $P$.

Now the first question is whether there is any $G$ conformant transition matrix $P$ such that $\pi^T P = \pi^T$. This question can be answered positively by the Metropolis-Hastings method [20], [9]. The Metropolis-Hastings method essentially finds such transition matrix $P_{MH}$ as follows. Initially let $\bar{P}$ be a transition matrix conformant to $G$ given by $P_{ij} \propto \frac{d_i}{\varepsilon}$, where $d_i$ is the degree of node $i$. It is known that the stationary distribution of $\bar{P}$ is proportional to $d_i$. Then by modification of the transition matrix $P$ either by adding appropriate self loops for each nodes or by probability adjustment, the Metropolis-Hastings method returns a transition matrix $P_{MH}$ that has $\pi$ as its stationary distribution and satisfies that $\pi_i(P_{MH})_{ij} = \pi_j(P_{MH})_{ji}$ for all $(i, j) \in E$.

Now to see how large $C(\varepsilon, P)$ is for such $P_{MH}$, consider the ring graph $G$ and the uniform $\pi$ on $G$. Then $P_{MH}$ is given by $P_{ij} = P_{ji} = \frac{1}{2}$ for all $(i, j) \in E$. Figure 1 (a) describes the ring graph $G$ and transition matrix $P_{MH}$. Then $|E(P_{MH})| = \Theta(n)$ and $T_\varepsilon(P_{MH}) = \Theta \left( n^2 \log \frac{n}{\varepsilon} \right)$, and hence we obtain $C(\varepsilon, P_{MH}) = \Theta \left( n^3 \log \frac{n}{\varepsilon} \right)$. This performance is clearly very poor because if every node passes its values to left, then this will take $n$ iterations and hence total cost would be $\Theta(n^2)$. Note that all the $P_{MH}$ generated by the Metropolis-Hastings method are reversible, i.e. for all $(i, j) \in E$, $\pi_i(P_{MH})_{ij} = \pi_j(P_{MH})_{ji}$. Especially when $\pi$ is uniform, $P_{MH}$ is symmetric. Boyd, Diaconis and Xiao [3] has shown that for the ring graph and uniform $\pi$, the mixing time $\mathcal{H}(P)$ is lower bounded by $\mathcal{H}(P) = \Omega(n^2)$ for any symmetric $P$. Together with (3), it implies that $C(\varepsilon, P) = \Omega(n^3)$. Hence to overcome this bound on $C(\varepsilon, P)$, we need to consider a non-reversible Markov chain $P$.

Diaconis, Holmes and Neal [7] first observed the following example called the lifted ring graph. The nodes of the lifted ring graph $G$ consist of two types: an inner circle and an outer circle. Transition on the inner circle forms a clockwise circulation and the transition on the outer circle forms a counterclockwise circulation. And the probability of changing from the inner circle to the outer circle and vice versa are $\frac{1}{n}$ at each time. Figure 1 (b) describes the lifted ring graph and its corresponding transition matrix $\bar{P}$. Then it can be checked that the uniform $\pi$ is the stationary distribution of $\bar{P}$, and that $\bar{P}$ mixes in $O \left( n \log \frac{n}{\varepsilon} \right)$ time, which implies that $C(\varepsilon, \bar{P}) = O \left( n^2 \log \frac{n}{\varepsilon} \right)$. But any reversible Markov chain defined on this ring-like graph mixes in $\Omega(n^2)$ time as in the $n$-ring graph, which implies that $C(\varepsilon, P) = \Omega(n^3)$ for any reversible $P$. Hence the non-reversible Markov chain $\bar{P}$ gives much smaller number of operations than any reversible $P$. Note that the lifted ring graph has similar structure to the $n$-ring graph. More precisely, the graph obtained by merging each inner node with the corresponding outer node forms the ring graph. The above construction suggests the following notion of Lifting Markov chain. Lifting a Markov chain involves possibly splitting each state into multiple ones; assigning a transition probability between two new states so that the projection of the new Markov chain (via mapping new states onto their original copies) results in the original Markov chain. See Section IV-A for the formal definition of lifting. This property implies that the linear iterations for the linear computation problem using lifted Markov chain can be implemented on the original network graph in a distributed manner. For example, consider the lifted ring graph on $2n$ nodes. Then each node on
the original ring graph \( G \) simulates two copied nodes of itself: one in the inner circle and the other one in the outer circle. Then all the edges of the lifted graph are located between two adjacent nodes in the original graph, hence message exchange of the lifted Markov chain can be simulated in the original network graph \( G \). Thus designing lifted Markov chain allows for fast distributed linear computation algorithm.

The work of Diaconis, Holmes and Neal was restricted to ring-like graphs. In order to distributed linear computation efficiently, we would like to build lifting for general graphs such that (i) \( T_e(\hat{P}) \) of lifted chain \( \hat{P} \) is small, and (ii) the size \(|E(\hat{P})|\) is small. In a subsequent work, Chen, Lovasz and Pak [6] provided an explicit construction to lift a general given Markov chain with almost optimal speed up in the mixing time. Specifically, for a given Markov chain \( P \), they constructed a lifted chain which has the mixing time \( \tilde{O}(1/\Phi(P))^2 \) and \( |E(P)| = \Omega(n^2/\Phi(P)) \), where \( \Phi(P) \) is the conductance of \( P \). Furthermore, they showed that their construction is the fastest mixing lifted chain achievable. For example, the chain \( \hat{P} \) in the lifted ring graph is the fastest mixing lifted chain of the \( P_{MH} \) on the ring graph because \( 1/\Phi(P_{MH}) = \Theta(n) \) in this case. But, if we follow the general construction built by Chen et. al. [6], the size of the lifted chain becomes \( \Omega(n^2/\Phi(P)) = \Omega(n^3) \), which is much larger than the size \( 2n \) of the chain in the lifted ring graph. The inefficiency of their construction in terms of size would increase the total number of operations in the linear iterations, hence our first question arises:

**Q1.** Can we construct a lifted chain which mixes as fast as that of [6], but has smaller size?

Note that Chen et. al. obtained the lower bound \( \tilde{O}(1/\Phi(P)) \) of the mixing time of the lifted Markov chain in terms of \( P \). However, for fixed \( G \) and \( \pi \) in the linear computation problem, two different Markov chains may give different bounds. For example, for the case of the \( n \)-ring graph, consider a non-reversible Markov chain \( P \) such that the probabilities of going clockwise and counterclockwise are \( 1/n, 1-1/n \) respectively. In this case, \( 1/\Phi(P) = \Theta(n^2) \), hence any lifted chain of \( P \) mixes much slower than the lifted chain in the lifted ring graph. Now we are interested in the lower bound of the mixing time of the lifted Markov chain for given underlying graph \( G \) and eigenvector \( \pi \). Its trivial lower bound is \( O(D) \), where \( D \) is a diameter of \( G \). Note that this lower bound is achievable for the case of the \( n \)-ring graph as we saw in the lifted ring graph. Also, as you see in the discussion of the Barbell graph in Section II-B, for a \( P \) defined on a given graph, \( D \) can be much smaller than \( 1/\Phi(P) \) that is the limitation of the mixing time of the original lifted chain. This motivates our second question:

**Q2.** For a given graph \( G \), can we construct a lifted Markov chain \( \hat{P} \) with the mixing time \( O(D) \) while \( \hat{P} \) having small size?

### A. Related Works

There has been a lot of past work on designing and analysis of iterative algorithms since the work by Tsitsiklis [22], and the recent excitement to design very efficient algorithms is due to its wide ranging applications in networks.

\[\text{For a function } f: \mathbb{N} \rightarrow \mathbb{R}^+, \tilde{O}(f(n)) := O(f(n)\text{poly}(\log n)).\]

\[\Phi(P) := \min_{S \subset V} \frac{Q(S,V \setminus S)}{\pi(S)\pi(V \setminus S)}, \text{ where } Q(A, B) = \sum_{i \in A, j \in B} \pi_i P_{ij}.\]
Specifically, Kempe, Dobra and Gehrke [13] obtained a randomized (gossip) algorithm for complete graph. Their algorithm achieves $C(\varepsilon, P) = O\left(n \log n \log \frac{1}{\varepsilon}\right)$. However their approach cannot be applied to general graphs. It was extended for general graph by Boyd, Ghosh, Prabhakar and Shah [5] using reversible $P$. The cost for their algorithm was $C(\varepsilon, P) = O\left(n(\log \frac{1}{\varepsilon} + \mathcal{H}(P))\right)$, where $\mathcal{H}(P)$ is the mixing time of Markov chain induced by $P$. Thus, this approach necessarily suffers from reversibility of $P$. In order to overcome this, Mosk-Aoyama and Shah [21] obtained a different gossip algorithm (based on extremal property of exponential distribution) which achieves $C(\varepsilon, P) = O\left(\frac{1}{\Phi(P)} n \log n \cdot \varepsilon^{-2}\right)$. While this algorithm removes curse of reversibility by bring the time-scaling down to $1/\Phi(P)$, it is scaling in terms of $\varepsilon$ is $1/\varepsilon^2$ instead of $\log 1/\varepsilon$. Further, as discussed above, $1/\Phi(P)$ can be arbitrarily bad compared to the diameter $D$ of graph. As we shall see, our work improves in terms of $\varepsilon$ scaling (ours is $\log 1/\varepsilon$) and graph structure.

B. Contributions

Pseudo-Lifting. We find that, in order to answer $Q1$ and $Q2$, the notion of lifting does not suffice. However, by modifying its definition, we are able to answer these questions affirmatively: this is the main insight of this paper. The new notion of lifting, called pseudo-lifting, can be defined as follows.

Definition 1 (Pseudo-Lifting): Consider a Markov chain with its transition matrix $P$. Let $\pi$ and $G = (V, E)$ be its stationary distribution and its underlying graph respectively. The Markov chain with its transition matrix $\tilde{P}$ ($\tilde{\pi}$ and $\tilde{G} = (\tilde{V}, \tilde{E})$ denote its stationary distribution and underlying graph respectively), is called a pseudo-lifting of $P$ if there exists a many-to-one function $f : \tilde{V} \rightarrow V$, $T \subset \tilde{V}$ with $|T| = |V|$ such that the following holds: (a) for any $\tilde{u}, \tilde{v} \in \tilde{V}$, $(\tilde{u}, \tilde{v}) \in \tilde{E}$ only if $(f(\tilde{u}), f(\tilde{v})) \in E$, and (b) for any $u \in V$, $\tilde{\pi}(f^{-1}(u) \cap T) = \frac{1}{2}\pi(u)$.

The property (a) in the definition implies that one can simulate the pseudo-lifted chain $\tilde{P}$ in the original graph $G$. Furthermore, the property (b) suggest that, by concentrating on set $T$, it is possible to run the linear iterations with a given $G$ and $\pi$ using $\tilde{P}$. See Section II-D for its detailed explanation. With this new concept of lifting, we show the following theorem.

Theorem 1: For a given $P$, there is an pseudo-lifted chain $\tilde{P}$ of $P$ with the mixing time $O(D)$ and size $(Dn + |E|)$, where $D$ and $E$ is the diameter and the edge-set of the underlying graph of $P$. Furthermore, it leads to an $\varepsilon$-approximation distributed algorithm for the linear computation $\langle x, \pi \rangle$ (i.e. every node know the value between $(1 \pm \varepsilon)\langle x, \pi \rangle$) with given $G$, $\pi$, $x$ and $\varepsilon > 0$ such that the running time is $T_{\varepsilon} = O\left(D \log \frac{1}{\varepsilon \pi_0}\right)$ and the total number of operations is $C(\varepsilon) = O\left(|E| + Dn \log \frac{1}{\varepsilon \pi_0}\right)$.

Note that the size guarantee $O(Dn + |E|)$ is significantly smaller than $\Omega(n^2/\Phi(P))$ which is the size of the lifted chain in [6] because $D \leq 1/\Phi(P)$ (sometimes $D$ can be much smaller than $1/\Phi(P)$ for a $P$ defined on a given graph, for example the Barbell graph discussed in Section II-B).

Pseudo-Lifting using geometry. Consider Theorem 1 for the ring graph. In this example, the total number of operations in the algorithm is $O(n^3 \log \frac{1}{\varepsilon})$, which is essentially same as $C(\varepsilon, P)$ with the reversible $P$ on this graph. The reason for this high cost compared to that with Diaconis et. al.’s lifted chain is we don’t utilize the geometry of the graph as they did. Furthermore, lifting a graph with such geometry is important because it usually suffers from the slow mixing time under the reversible random walk (for example, the expander graph mixes fast, but has no geometry). Motivated by these, we consider the geometry of graphs, and its natural and well-studied model (see [2], [10], [8]) is a class of graphs with constant doubling dimension, which is defined as follows.

Definition 2: Consider a metric space $\mathcal{M} = (\mathcal{X}, d)$, where $\mathcal{X}$ is the set of point endowed with a metric $d$. Given $x \in \mathcal{X}$, define a ball of radius $r \in \mathbb{R}_+$ around $x$ as $B(x, r) = \{y \in \mathcal{X} : d(x, y) < r\}$. Define

$$
\rho(x, r) = \inf\{K \in \mathbb{N} : \exists y_1, \ldots, y_K \in \mathcal{X}, B(x, r) \subset \bigcup_{i=1}^{K} B(y_i, r/2)\}.
$$

Then, the $\rho(\mathcal{M}) = \sup_{x \in \mathcal{X}, r \in \mathbb{R}_+} \rho(x, r)$ is called the doubling constant of $\mathcal{M}$ and $\log_2 \rho(\mathcal{M})$ is called the doubling dimension of $\mathcal{M}$. Doubling dimension of a graph $G = (V, E)$ is defined with respect to the metric induced on $V$ by the shortest path metric.

Essentially, $G$ having doubling dimension $\rho$ is equivalent to the condition that the number of nodes of which distance from a fixed node is $k$ grows like $k^\rho$. For example, the ring graph has doubling dimension 1, and the

\[\text{In fact, } \frac{1}{2} \text{ can be replaced by any constant between 0 and 1.}\]
geometric random graph on a unit square (i.e. nodes are placed randomly on the plane and two nodes are adjacent only if their distance is less than $\Theta(1/\sqrt{n})$) has constant doubling dimension with high probability. In order to utilize the doubling dimensional structure of $G$, we design a construction of the pseudo-lifted chain hierarchically. It has the smaller size than the previous construction, while maintaining the optimal mixing time. We show the following theorem which shows the advantage of this construction.

**Theorem 2:** For a given $P$, there is a pseudo-lifted chain $\tilde{P}$ of $P$ with the mixing time $O(D)$ and size $\Theta(Dn^{1-1/\epsilon})$, where $D$ and $\rho$ is the diameter and the doubling dimension of the underlying graph of $P$ respectively. Furthermore, it leads to an $\epsilon$-approximation distributed algorithm for the linear computation $\langle x, \pi \rangle$ with given $G$, $\pi$, $x$ and $\epsilon > 0$ such that the running time is $O\left(D\log \frac{1}{\epsilon^2\rho}\right)$ and the total number of operations is $O\left(D^2n^{1-1/\epsilon}\rho \log \frac{1}{\epsilon^2\rho}\right)$.

Compared to the previous construction without geometry, it achieves the reduction in size up to $\Theta(n^{1/\epsilon})$ factor. For example, the random walk in $\mathbb{R}^2$ and $\mathbb{R}^3$ dimensional grid graphs have the mixing time $\Omega(n)$ and $\Omega(n^{2/3})$ respectively, and the size $O(n)$ for both, while the pseudo-lifted chain of them have the mixing time $O(n^{1/2})$ and $O(n^{1/3})$, and the size $O(n^{7/6})$ and $O(n^{13/12})$ respectively. Note that after pseudo-lifting the size increases and the mixing time decreases. Such a trade-off always happens in the lifted chain.

**Lifting using expander.** Now, we construct a lifted chain based on the *expander* graph, instead of the complete graph in [6], to reduce the size of the lifting. Although the reduced size of this construction is still bigger than that of the pseudo-lifted chain we constructed, the primarily reason to present this construction is for the completeness and mathematical interest (our new method for analyzing mixing time of certain *hybrid Markov chains* should be of interest in its own right). Also, this construction answers directly to Q1 without the notion of pseudo-lifting. Specifically, we prove the following Theorem.

**Theorem 3:** For an arbitrary reversible Markov chain $P$, there exists its lifted chain which mixes in $\tilde{O}(1/\Phi(P))$ time and has size $\tilde{O}(|E|/\Phi(P))$, where $E$ is the edge-set of the underlying graph of $P$.

Therefore, our construction leads to the reduction in the size of the lifted chain up to $\Theta(n^2/|E|)$ with respect to the construction in [6]. Our construction builds on the technique of [6], but the usage of the expander graph in place of the complete graph requires new proof techniques for establishing the bound on the mixing time. First, we need to consider a different flow problem, and the flow shortening technique in [15]. Second, we analyze the mixing time of the hybrid-type random walk.

**Remarks.** Some important remarks are in order.

1. $P$ is constructed in centralized manner in the above Theorems. However, the problem is of distributed computation of $\langle x, \pi \rangle$. In most of the applications stated, $G$ and $\pi$ remains the same while $x$ changes. Therefore in such applications one may imagine that $P$ is constructed once and then used many times. Currently constructing $P$ in efficient distributed manner remains as an interesting open question.

2. The following spanning tree based approach may be a generic alternate approach. First we construct a spanning tree of $G$ and then use it as infrastructure to send information around directly. This approach and iterative algorithm have competed forever in the research community. Both have their pros and cons. Tree based approach is not robust for edge failure but the iterative algorithm is; tree based approach takes $O(D)$ time but usually iterative algorithm takes longer. Now our pseudo-lifting based linear computation achieves $O(D)$ time and thus competes with tree based approach.

3. Finally we note that our result imply the following nontrivial and very important implication: we provide fastest possible mixing Markov chain subject to topology constraints. This is useful for many other important applications like sampling.

**C. Organization**

In Section II, we prove Theorem 1: we explain how to build the pseudo-lifting and use it to design a distributed algorithm for the linear computation. In Section III, we describe the hierarchical construction of pseudo-lifting using geometry of $G$, and it leads the proof of Theorem 2. In Section IV, we prove Theorem 3: we explain how to construct a lifting based on expander graphs and then analyze its mixing time and size.
II. THE PSEUDO-LIFTING: PROOF OF THEOREM 1

A. Proof Strategy and Preliminaries

The following preliminary notions are necessary to describe the algorithm for Theorem 1.

Ergodic flow. The ergodic flow matrix \( Q = \{q_{ij}\} \) of \( P \) is defined as: \( q_{ij} = \pi_i p_{ij} \), where \( \pi \) is a stationary distribution of \( P \) i.e. \( \pi^T P = \pi \). It satisfies: \( \sum_{i,j} q_{ij} = 1 \), \( \sum_i q_{ij} = \pi_j \) and \( \sum_i q_{ij} = \pi_j \). Furthermore, every non-negative matrix \( Q \) with these properties defines a Markov chain with the stationary distribution \( \pi \).

Mixing time. Although there are several definitions of Mixing time based on the different measures, we primarily consider the definition of Mixing time from the stopping rule. Starting from the initial node \( x \), we prove the relation between the total number of operations and \( H \) algorithm, in Section II-C we show the bound for the mixing time \( H \) of \( \hat{P} \) we constructed. Then, in Section II-D we prove the relation between the total number of operations and \( H \).

B. Construction of Pseudo-lifting

For a given Markov Chain \( P \), we will construct the pseudo-lifted chain \( \hat{P} \) of \( P \). Let \( \pi \) and \( G = (V,E) \) be the stationary distribution and the underlying graph of \( P \) respectively. We will construct the lifted graph \( \hat{G} \) by adding vertices and edges to \( G \), and decide the values of the ergodic flows \( \hat{Q} \) on \( \hat{G} \), which defines its corresponding Markov chain \( \hat{P} \).

Let \( D \) be the diameter of \( G \). First, select an arbitrary node \( v \). Now, for each \( w \in V \), there exist paths \( P_{vw} \) and \( P_{vw'} \), from \( w \) to \( v \) and \( v' \) to \( w \) respectively. We will assume that all the paths are of length \( D \): this can be achieved by repeating same nodes. Now, we construct a pseudo-lifted graph \( \hat{G} \) starting from \( G \).

First, create a new node \( v' \) which is a copy of the chosen vertex \( v \). Then, for every node \( w \), add directed paths \( P'_{wv} \), a copy of \( P_{vw} \), from \( w \) to \( v' \). Similarly, add \( P'_{vw'} \) (a copy of \( P_{vw} \)) from \( v' \) to \( w \). Each addition creates \( D - 1 \) new interior nodes. Thus, we have essentially created a virtual star topology using the paths of the old graph and totally added \( O(nD) \) new nodes. (Every new node is a copy of an old node.) For example, see Figure 5 in the Appendix which describes the lifted graph of the ring graph \( G \) with 4 nodes.

Now, we define the ergodic flow \( \hat{Q} \) for this graph \( \hat{G} \) as follows: for an edge \((i,j)\),

\[
\hat{q}_{ij} = \begin{cases} 
\frac{\epsilon}{2D} \pi_w & \text{if } (i,j) \in E(P'_{wv}) \text{ or } E(P'_{vw}) \\
(1 - \epsilon) q_{ij} & \text{if } (i,j) \in E(G),
\end{cases}
\]

where \( \epsilon \in [0,1] \) is a constant we will decide later. It is easy to check that \( \sum_{ij} \hat{Q}_{ij} = 1 \), \( \sum_j \hat{Q}_{ij} = \sum_j \hat{Q}_{ji} \). Hence it defines a Markov chain on \( \hat{G} \). The stationary distribution of this lifting is

\[
\hat{\pi}_i = \begin{cases} 
\frac{\epsilon}{2D} \pi_w & \text{if } i \in (V(P'_{wv}) \cup V(P'_{vw})) \setminus \{w,v'\} \\
(1 - \epsilon + \frac{\epsilon}{2D}) \pi_i & \text{if } i \in V(G) \\
\frac{\epsilon}{2D} & \text{if } i = v'
\end{cases}
\]

Given the above definition of \( \hat{Q} \) and corresponding stationary distribution \( \hat{\pi} \), it satisfies the pseudo-lifting definition if we choose \( \epsilon \) such that \( 1/2 = \epsilon \left(1 - \frac{1}{2D}\right) \) and set \( T = V(G) \) (i.e. \( T \) is the set of old nodes). Clearly, the number of edges in \( \hat{G} \) is \( |E| + 2Dn \).
C. Mixing time

We claim the following bound on Mixing time of the pseudo-lifted chain we constructed.

**Lemma 4**: The mixing time of the Markov chain $\hat{P}$ defined by $\hat{Q}$ is $O(D)$.

**Proof**: Consider the following stopping rule. Walk until visiting $v'$, and toss a coin $X$ with the following probability.

$$X = \begin{cases} 
0 & \text{with probability } \frac{\varepsilon}{2D}, \\
1 & \text{with probability } \frac{\varepsilon(D-1)}{2D}, \\
2 & \text{with probability } 1 - \varepsilon + \frac{\varepsilon}{2D}, \\
3 & \text{with probability } \frac{\varepsilon(D-1)}{2D}.
\end{cases}$$

Depending on the value of $X$, the stopping node is decided as follows.

- $X = 0$: Stop at $v'$.
  - The probability for stopping at $v'$ is $\Pr[X = 0] = \frac{\varepsilon}{2D}$, which is exactly $\pi_{v'}$.
- $X = 1$: Walk a directed path $P'_{vw}$, and choose an interior node of $P'_{vw}$ uniformly at random, and stop there.
  - For a given $w$ the probability for walking $P'_{vw}$ is easy to check $\pi_w$. There are $D - 1$ many interior nodes, hence, for an interior node $i$ of $P'_{vw}$, the probability for stopping at $i$ is
    $$\Pr[X = 1] \times \pi_{w} \times \frac{1}{D - 1} = \frac{\varepsilon}{2D} \pi_{w} = \hat{\pi}_{i}.$$ 
- $X = 2$: Stop at the end node $w$ of $P'_{vw}$.
  - The probability for stopping at $w$ is
    $$\Pr[X = 2] \times \Pr[\text{walk } P'_{vw}] = \left(1 - \varepsilon + \frac{\varepsilon}{2D}\right) \times \pi_{w} = \hat{\pi}_{w}.$$ 
- $X = 3$: Walk until getting a directed path $P'_{vw}$, and choose an interior node of $P'_{vw}$ uniformly at random, and stop there.
  - Until getting a directed path $P'_{vw}$, the lifted random walk defined by $\hat{Q}$ is same as the original random walk. Since the distribution $w \in V(G)$ of the walk at the end of the previous step is exactly $\pi$, it follows that the distribution $\pi$ over the nodes of $V(G)$ is preserved under this walk till walking on $P'_{vw}$. From the same calculation as the case $X = 1$, the probability for stopping at the interior node $i$ of $P'_{vw}$ is $\hat{\pi}_{i}$.

Therefore, we have established the existence of a stopping rule that takes an arbitrary starting distribution to the stationary distribution $\hat{\pi}$. Now, this stopping rule has average length $O(D/\varepsilon)$: since the probability of getting on a directed path $P'_{vw}$ at $w$ is $\frac{\varepsilon}{2D}(1 - \varepsilon + \frac{\varepsilon}{2D}) = \Theta(\varepsilon/D)$, the expected numbers of walks until visiting $v'$ and getting a directed path when $X = 3$ are $O(D/\varepsilon) = O(D)$ in both cases. This completes the proof. 

**Example 1**: To see the importance of Lemma 4, consider the Barbell graph of $2n$ nodes shown in Figure 4: two complete graphs of $n$ nodes connected by a single edge. The size of this graph, $|E| = \Theta(n^2)$. Now, consider a random walk where the next transition is uniform among all the neighbor for each node. For such a random walk, it is easy to check that $1/\Phi(P) = \Omega(n^2)$. Therefore, the mixing time of any (original) lifted chain is at least $\Omega(n^2)$. However, this random walk is ill-designed to begin with because $1/\Phi(P)$ can be decreased up to $O(n)$ by defining its random walk in another way (i.e. increasing the probability of its linkage edge, and adding self-loops to non-linkage nodes not to change its stationary distribution). Note that the diameter $D = O(1)$. Therefore, our construction provides a pseudo-lifted Markov chain with the mixing time $O(1)$ not depending on starting $P$.

D. Proof of Theorem 1

The linear iterations for a given $G$ and $\pi$ can be implemented using $\hat{P}$ and $\hat{G}$ as follows. By each node in $G$ simulating its copies in $\hat{G}$, we can simulate the linear iterations with $\hat{P}, \hat{G}$ and $\hat{\pi}$ in the original graph $G$. Initially, $y_i(0) = x_i$ for every old node $i \in V(G)$, and $y_i(0) = 0$ otherwise. Now, only focus on the old nodes $V(G)$ in $\hat{G}$. Because $\hat{\pi}_i = \left(\frac{1}{2} + \frac{1}{4D}\right)\pi_i$ for such a node $i \in V(G)$,

$$\lim_{t \to \infty} y_i(t) = \langle y(0), \hat{\pi} \rangle = \left(\frac{1}{2} + \frac{1}{4D}\right) \langle x, \pi \rangle$$
Hence, after a sufficient large $t$, every node $i$ can estimate $\langle x, \pi \rangle$ by calculating $\frac{y_i(t)}{z + t}$. From the construction of the algorithm, it suffices to show that $C(\varepsilon, \tilde{P}) = O\left(\left(|E| + Dn\right)D \log \frac{1}{\varepsilon \pi_0}\right)$. We need the following lemma.

Lemma 5: $T(\tilde{P}) = O \left(\mathcal{H}(\tilde{P}) \log \frac{1}{\varepsilon \pi_0}\right)$. Hence, $T(\tilde{P}) = O \left(D \log \frac{1}{\varepsilon \pi_0}\right)$.

Proof: Here, we need the $\varepsilon$-mixing time $\tau(\varepsilon)$ based on the total variance distance, and it is defined as follows:

$$\tau(\varepsilon) = \min \left\{ t : \forall i \in G, \frac{1}{2} \sum_{j \in G} |P_{ij}^t - \pi_j| \leq \varepsilon \right\}.$$ 

The following relation between two different mixing time $\tau(\varepsilon)$ and $\mathcal{H}$ is known (see [17]):

$$\tau(\varepsilon) = O \left(\mathcal{H} \log \frac{1}{\varepsilon}\right).$$

If $t$ is larger than $\tau(\varepsilon \pi_0/4)$ of $\tilde{P}$, which is $O \left(\mathcal{H}(\tilde{P}) \log \frac{1}{\varepsilon \pi_0/4}\right) = O \left(D \log \frac{1}{\varepsilon \pi_0}\right)$ from Lemma 4,

$$|y_i(t) - \langle y(0), \pi \rangle| = \left| \sum_j \tilde{P}_{ij}^t y_j(0) - \sum_j y_j(0) \pi_j \right| \leq \sum_j y_j(0) \left| \tilde{P}_{ij}^t - \pi_j \right| \leq \sum_j y_j(0) \varepsilon \pi_0 = \varepsilon \langle y(0), \pi \rangle,$$

where (a) is from $|\tilde{P}_{ij}^t - \pi_j| \leq \sum_j |\tilde{P}_{ij}^t - \pi_j| \leq 2 \times \frac{\varepsilon \pi_0}{2} = \frac{\varepsilon \pi_0}{2}$, and (b) is because $\pi_j > \frac{1}{2} \pi_j \geq \frac{1}{2} \pi_0$ for every old node $j \in V(G)$, and $y_j(0) = 0$ otherwise. This completes the proof.

Note that the relation $T_\varepsilon(P) = O \left(\mathcal{H}(P) \log \frac{1}{\varepsilon \pi_0}\right)$ holds for any Markov chain $P$. From Lemma 5 and the fact $|\tilde{E}| = |E| + 2Dn$, $C(\varepsilon, \tilde{P}) = T_\varepsilon(\tilde{P}) \times |\tilde{E}| = O \left(\left(|E| + Dn\right) D \log \frac{1}{\varepsilon \pi_0}\right)$. This completes the proof of Theorem 1.

III. HIERARCHICAL CONSTRUCTION FOR THE PSEUDO-LIFTING: PROOF OF THEOREM 2

A. Motivation

Fig. 2. For a given line graph with $n$ nodes, (a) is the star topology which used in the construction of the pseudo-lifted graph in Section II. (b) is the hierarchical star topology which will be used in this section for the new construction of pseudo-lifting.

The basic idea for the construction of the pseudo-lifting in Section II-B is creating a virtual star topology using paths from every node to a fixed one, and the length of paths grows the size of the lifted chain. However, as you see the example of the line graph in Figure 2, one can create the hierarchical star topology (or say tree topology), and intuitively the total sum of lengths of paths becomes smaller. To create such a topology, we need to decide which nodes would become middle ones (between a root and leaves). A natural candidate for them is the $R$-net $Y \subset V$ of graph $G$ defined as follows:

(i) For every $v \in V$, there exists $u \in Y$ such that the shortest path distance between $u, v$ is at most $R$.
(ii) The distance between any two $y, z \in Y$ is more than $R$.

Such an $R$-net can be found in $G$ greedily. To guarantee the existence of a good $R$-net, we need geometric structure for the graph $G$, and the doubling dimension of $G$ introduced earlier is a natural approach for it.
B. Construction of Pseudo-Lifting using geometry

For a given Markov Chain $P$, we will construct the pseudo-lifted chain $\hat{P}$ of $P$ using the hierarchical star topology. Denote $\pi$ and $G = (V, E)$ be the stationary distribution and the underlying graph of $P$ again. As the previous construction in Section II-B, we will construct the lifted graph $\hat{G}$ by extending $G$, and define the ergodic flows $\hat{Q}$ on $\hat{G}$, which leads to its corresponding Markov chain $\hat{P}$.

Given a $R$-net $Y$, match each node $w$ to the nearest $y \in Y$ (breaking ties arbitrarily). Let $C_y = \{w \mid w \text{ matched to } y\}$ for $y \in Y$. Clearly, $V = \cup_{y \in Y} C_y$. Finally, for each $y \in Y$ and for any $w \in C_y$ we have paths $P_{wy}$, $P_{yw}$ between $w$ and $y$ of length exactly $R$. Also, for each $y \in Y$, there exits $P_{yy}$, between $y$ and $v$ of length exactly $D$ (we allow the repetition of nodes to hit this length exactly).

Now, we construct the lifted graph $\hat{G}$. As the construction in Section II-B, select an arbitrary node $v \in V$ and create its copy $v'$. For each $y \in Y$, create two copies $y_1'$ and $y_2'$. Now, add directed paths $P'_{wy}$, a copy of $P_{wy}$, from $w$ to $y_1'$ and add $P'_{yy}$, a copy of $P_{yy}$, from $y_1'$ to $v'$. Similarly, add $P'_{yy}$ and $P'_{y_2}$ between $v'$, $y_2'$ and $y_2$, $w$. This construction adds $2D|Y| + nR$ edges to $G$ to give $\hat{G}$. Now, the ergodic flow $\hat{Q}$ on $\hat{G}$ is defined as follows: for any $(i, j)$ of $\hat{G}$,

\[
\hat{Q}_{ij} = \begin{cases} 
\frac{\pi}{2(R+D)} \pi_w & \text{if } (i, j) \in E(P'_{wy}) \text{ or } E(P'_{yy}) \\
\frac{\pi}{2(R+D)} \pi(C_y) & \text{if } (i, j) \in E(P'_{yy}) \text{ or } E(P'_{y_2}) \\
(1 - \varepsilon)Q_{ij} & \text{if } (i, j) \in E(G)
\end{cases}
\]

where $\pi(C_y) = \sum_{w \in C_y} \pi_w$ and $\varepsilon \in [0, 1]$ is a constant decided later. It can be checked that $\sum_{ij} \hat{Q}_{ij} = 1$, $\sum_{ij} \hat{Q}_{ij} = \sum_{ji} \hat{Q}_{ji}$. Hence it defines a Markov chain on $\hat{G}$. The stationary distribution of this lifted chain is

\[
\hat{\pi}_i = \begin{cases} 
\frac{\pi}{2(R+D)} \pi_w & \text{if } i \in (V(P'_{wy}) \cup V(P'_{yy})) \setminus \{w, y_1', y_2'\} \\
\frac{\pi}{2(R+D)} \pi(C_y) & \text{if } i \in (V(P'_{yy}) \cup V(P'_{y_2})) \setminus \{v'\} \\
(1 - \varepsilon)(1 - \frac{\pi}{2(R+D)}) \pi_i & \text{if } i \in V(G) \\
\frac{\pi}{2(R+D)} & \text{if } i = v'
\end{cases}
\]

To establish this lifting as the pseudo-lifting of the original Markov chain $P$, consider $T = V(G)$ and $\varepsilon$, where $\frac{1}{2} = \varepsilon \left(1 - \frac{1}{2(R+D)}\right)$. The $\hat{G}$ has exactly $|E| + 2Rn + 2D|Y|$ edges as noted earlier. We state the following result about the mixing time, and see the Appendix for its proof which can be done similarly as the proof of Lemma 4.

**Lemma 6:** The mixing time of the Markov chain $\hat{P}$ defined by $\hat{Q}$ is $O(D)$.

C. Proof of Theorem 2: Application to Constant Doubling Dimension Graphs

We design an algorithm and use the proof strategy as same as those of Theorem 1 in Section II. The difference is we need to analyze the size of the lifted chain using a constant doubling dimension $\rho$ of $G$, and we show the following lemma

**Lemma 7:** Given a graph $G$ with a constant doubling dimension $\rho$ and its diameter $D$, the hierarchical construction gives a pseudo-lifted graph $\hat{G}$ with its size $|\hat{E}| = O(Dn^{1-\frac{1}{\rho+1}})$.

**Proof:** The property of doubling dimension implies that there exists an $R$-net $Y$ such that $|Y| \leq (2D/R)^\rho$ (cf. [2]). Consider $R = D2^{\frac{\rho}{\rho+1}}n^{-\frac{1}{\rho+1}}$. This is an appropriate choice because $R = D2^{\frac{\rho}{\rho+1}}n^{-\frac{1}{\rho+1}} > Dn^{-\frac{1}{\rho+1}} > n^{\frac{1}{\rho}} > \frac{1}{\rho}$ (the second inequality is from $n \leq D^\rho$). Given this, the size of the lifted graph $\hat{G}$ is

\[|\hat{E}| = |E| + 2Rn + 2D|Y| \leq |E| + 2D \left(2^{\frac{\rho}{\rho+1}}\right)n + 2D \left(2^{\frac{\rho}{\rho+1}}\right)^\rho = |E| + O(Dn^{1-\frac{1}{\rho+1}}).
\]

Since $|E| = O(n)$ and $D = \Omega(n^{1/\rho})$, we have that $|\hat{E}| = O(Dn^{1-\frac{1}{\rho+1}})$.

Therefore, using Lemma 6,7 and doing the similar analysis in Section II-D, the total number of operations need for the algorithm is

\[C(\varepsilon, \hat{P}) = T_\varepsilon(\hat{P}) \times |\hat{E}| = O(D \log \frac{1}{\varepsilon \pi_0}) \times O(Dn^{1-\frac{1}{\rho+1}}) = O(D^2 n^{1-\frac{1}{\rho+1}} \log \frac{1}{\varepsilon \pi_0}).\]
IV. EFFICIENT LIFTING USING EXPANDERS: PROOF OF THEOREM 3

In what follows, we will consider only lazy \(^5\) reversible \(^6\) \(P\).

A. Preliminaries

Lifting. The following definition of lifting, introduced in [7], [6].

Definition 4 (Lifting): Consider a Markov chain with transition matrix \(P\) and stationary distribution \(\pi\) defined on graph \(G = (V, E)\). A Markov chain with transition matrix \(\hat{P}\), stationary distribution \(\hat{\pi}\) on graph \(\hat{G} = (\hat{V}, \hat{E})\) is called lifting of \(P\) if there exists a many-to-one function \(f : \hat{V} \rightarrow V\) such that the following holds: (a) for any \(\hat{u}, \hat{v} \in \hat{V}\), \((\hat{u}, \hat{v}) \in \hat{E}\) only if \((f(\hat{u}), f(\hat{v})) \in E\); (b) for any \(u, v \in V\), \(\pi(u) = \hat{\pi}(f^{-1}(u))\), and \(Q(u, v) = \hat{Q}(f^{-1}(u), f^{-1}(v))\). Here \(Q\) and \(\hat{Q}\) are ergodic flow matrix for \(P\) and \(\hat{P}\) respectively.

Multi-commodity Flows. We consider a multi-commodity flow problem on \(G\) with the capacity constraint on edge \((u, v) \in E\) given by \(Q_{uv}\). A flow from a source \(s\) to a destination \(t\), denoted by \(f\), is defined as a non-negative function on edges of \(G\) so that

\[\sum_j f(ji) = \sum_j f(ij)\]

for every node \(i \neq s, t\). The value of the flow is defined by

\[\text{val}(f) = \sum_j f(sj) - \sum_j f(js) = \sum_j f(st) - \sum_j f(tj)\].

A multi-commodity flow is a collection \(f = (f^{st})\) of flows, where each \(f^{st}\) is a flow from \(s\) to \(t\). Define the congestion of a multi-commodity flow \(f\) as

\[\max_{(i,j)\in E} \frac{\sum_{s,t} f^{st}(ij)}{Q_{ij}}\].

Expander. The expander graphs are sparse graphs which have high connectivity properties, quantified using the edge expansion \(h(G)\) as defined as

\[h(G) = \min_{1 \leq |S| \leq |V|/2} \frac{|\partial(S)|}{|S|}\],

where \(\partial(S)\) is the set of edges with exactly one endpoint in \(S\). For constants \(d\) and \(c\), a family \(\mathcal{G} = \{G_1, G_2, \ldots\}\) of \(d\)-regular graphs is called a \((d, c)\)-expander family if \(h(G) > c\) for every \(G \in \mathcal{G}\). There are many explicit constructions of a \((d, c)\)-expander family available in recent times. We will use a \((d, c)\)-expander graph \(G^{Ex} = (V, E^{Ex})\) (i.e. \(V^{Ex} = V\)), and a transition matrix \(P^{Ex}\) defined on this graph. For a given \(\pi\), we can define a reversible \(P^{Ex}\) so that its stationary distribution is \(\pi\) as follows,

\[P^{Ex}_{ij} = \begin{cases} \frac{\pi_i}{\pi_j} & \text{if } (i, j) \in E^{Ex} \\ 1 - \frac{\pi_0}{\pi_i} & \text{if } i = j \end{cases}\].

In the case of \(\pi_{max} = O(\pi_0)\), it is easy to check that \(\Phi(P^{Ex}) = \Theta(h(G)) = \Omega(1)\), where \(\Phi(P^{Ex})\) is the conductance of \(P^{Ex}\). Hence, \(\lambda_{P^{Ex}} = \Omega(1)\), and the random walk defined by \(P^{Ex}\) mixes fast. In this section, we will consider only such \(\pi\).

B. Construction

We use the multi-commodity flow based construction which was introduced in [6]. They essentially use a multi-commodity flow between source-destination pairs for all \(s, t \in V\). Instead, we will use a balanced multi-commodity flow between source-destination pairs that are obtained from an expander. Thus, the essential change in our construction is the use of an expander in place of a complete graph used in [6]. A caricature of this lifting is

\(^5\)\(P\) is lazy if \(P \geq I/2\). We can assume this without loss of generality for the purpose of mixing time because if \(P\) is not such then we can modify it as \((I + P)/2\); the mixing time of \((I + P)/2\) is within constant factor of the mixing time of \(P\).

\(^6\)The construction in this section even works for a non-reversible \(P\). However, we only consider a reversible \(P\) to simplify the statements of Theorem 3 and Lemma 9 and 10.
explained in Figure 3. However, this change makes analysis of mixing time lot more challenging and requires us to use different analysis technique. Further, we use arguments based on the classical linear programming to derive the bound on the size of lifting.

Fig. 3. A caricature of lifting using expander. Let line graph $G$ be a line graph with 4 nodes. We wish to use an expander $G^{Ex}$ with 4 nodes, shown on the top-right side of the figure. $G$ is lifted by adding paths that correspond to edges of expander. For example, an edge $(2, 4)$ of expander is added as path $(2, 3', 4)$. We also draw the lifting in [6] which uses the complete graph.

To this end, we consider the following multi-commodity flow: let $G^{Ex} = (V, E^{Ex})$ be an expander with a transition matrix $P^{Ex}$ and a stationary distribution $\pi$ as required – this is feasible since we have assume $\pi_{max} = O(\pi_0)$.

We note that this assumption is used only for existence of expander. Consider a multi-commodity flow $f = (f_{st})_{(s,t) \in E^{Ex}}$ so that
(a) $val(f_{st}) = \pi_s P_{st}^{Ex} = Q_{st}^{Ex}, \forall (s, t) \in E^{Ex}$,
(b) $\sum_{s,t} f_{st}(ij) \leq K \pi_{ij}, \forall (i, j) \in E$;

Lemma 8: There is a feasible multi-commodity flow in the above flow problem with congestion($K$) and path-length at most $\tilde{C} = O(1/\Phi(P))$. The proof of this lemma is in the Appendix. Now, we can think of this multi-commodity flow as a weighted collection of directed paths $\{(P_r, w_r) : 1 \leq r \leq N\}$, where the total weight of paths from node $s$ to $t$ is $\pi_s P_{st}^{Ex}$, where $(s, t) \in E^{Ex}$. Let $\ell_r$ be the length of path $P_r$. From Lemma 8, we have the following:

$$\sum_r w_r = 1, \quad \ell_r \leq \tilde{C},$$

$$\sum_{r: P_r \text{ starts at } i} w_r = \pi_i, \quad \sum_{r: P_r \text{ ends at } i} w_r = \pi_i, \quad \text{for } i \in V$$

$$\sum_{r: (i,j) \in E(P_r)} w_r \leq \tilde{C} \pi_{ij}, \quad \text{for } (i, j) \in E.$$

Using such a collection of weighted paths, we construct the desired lifting next. As Figure 3, we construct the lifted graph $\tilde{G} = (\tilde{V}, \tilde{E})$ from $G$ by adding a directed path $P_r'$ of length $\ell_r$ connecting $i$ to $j$ if $P_r$ goes from $i$ to $j$. Subsequently, $\ell_r - 1$ new nodes are added to the original graph. The ergodic flow on an edge $(i, j)$ of the lifted chain is defined by

$$\tilde{Q}_{ij} = \begin{cases} w_r/2\tilde{C} & \text{if } (i, j) \in E(P_r') \\ Q_{ij} - \sum_{r: i,j \in E(P_r')} w_r/2\tilde{C} & \text{if } (i, j) \in E(G) \end{cases}$$

$$\pi_{max} = \max_{i \in V} \pi_i.$$
It is easy to check it defines a Markov chain on \( \hat{G} \), and a natural way of mapping the paths \( P'_{r} \) onto the paths \( P_{r} \) collapses the random walk on \( \hat{G} \) onto the random walk on \( G \). The stationary distribution of the lifted chain is

\[
\hat{\pi}_i = \begin{cases} 
\frac{w_r/2\hat{C}}{\pi_r} & \text{if } i \in V(P'_{r}) \setminus V(G) \\
\pi_i - \sum_{r: P_{r} \text{ thru } i} w_r/2\hat{C} & \text{if } i \in V(G)
\end{cases}
\]

Thus, the above stated construction is a valid lifting of given Markov chain \( P \) defined on \( G \).

### C. Performance of lifting: Mixing time and Size

We first state the result about the mixing time of the above constructed lifted Markov chain, whose proof is presented in the Appendix.

**Lemma 9:** The mixing time, \( \hat{\tau} \) of the lifted Markov chain represented by \( \hat{Q} \) defined on \( \hat{G} \) is \( \hat{O}(1/\Phi(P)) \).\(^8\)

Next, we state the bound on the size of the above lifted chain as follow: again, its proof is in the Appendix.

**Lemma 10:** The size of the lifted Markov chain can be bounded above as \( \hat{O}(|E|/\Phi(P)) \).\(^9\)

### REFERENCES


\(^8\)The precise bound is \( O(\mathcal{C} \log \frac{1}{\mathcal{P}}) \), where \( \mathcal{C} \) is defined in the Appendix, and this bound holds even for a non-reversible \( P \) although we assumed \( P \) is reversible at the beginning of this section.

\(^9\)The precise bound is \( O(\mathcal{E}(\mathcal{C})) \), where \( \mathcal{C} \) is defined in the Appendix, and this bound holds even for a non-reversible \( P \) although we started assumed \( P \) is reversible at the beginning of this section.
D. Figure for Example 1

Fig. 4. The Barbell graph with $2n$ nodes: here $n = 6$.

E. Figure for Example of Construction in Proof of Theorem 1

Fig. 5. The pseudo-lifted chain $\tilde{P}$ of the reversible chain $P$ on a ring graph $G$ with 4 nodes. $G$ has $\{v, w, x, y\}$ as the set of nodes, and its diameter $D$ is 2. To construct the pseudo-lifted graph $\tilde{G}$, select a node $v$ of $G$. For a vertex $x$ of $G$, find a path $P(x \rightarrow w \rightarrow v)$ from $x$ to $v$, and add its copy $P'$ twice for both directions. Every dotted line and vertex are copies of there original one. Similarly, add copies of path with length 2 for other vertices (including $v$) as well. Finally, $\{v', v_1, v_2, v_3, v_4, v_5, v_6\}$ and $\{w_1, w_2\}$ are set of copies of $v$ and $w$ respectively.
APPENDIX

A. Proof of Lemma III-B

Consider the following stopping rule. Walk until visiting \( v' \), and toss a coin \( X \) with the following probability.

\[
X = \begin{cases} 
0 & \text{with probability } \frac{\varepsilon}{2(R+D)} \\
1 & \text{with probability } \frac{\varepsilon D}{2(R+D)} \\
2 & \text{with probability } \frac{\varepsilon(R-1)}{2(R+D)} \\
3 & \text{with probability } 1 - \varepsilon(1 - \frac{\varepsilon}{2(R+D)}) \\
4 & \text{with probability } \frac{\varepsilon}{2(R+D)} \\
5 & \text{with probability } \frac{\varepsilon D}{2(R+D)} 
\end{cases}
\]

Depending on the value of \( X \),
- \( X = 0 \): Stop at \( v' \).
- \( X = 1 \): Walk on a directed path \( P'_{vy} \), and choose its interior node uniformly at random, and stop there.
- \( X = 2 \): Walk until getting a directed path \( P'_{yw} \), and choose its interior node uniformly at random, and stop there.
- \( X = 3 \): Walk until getting an old node in \( V(G) \), and stop there.
- \( X = 4 \): Walk until getting a directed path \( P'_{wy} \), and choose its interior node uniformly at random, and stop there.
- \( X = 5 \): Walk until getting a directed path \( P'_{yw} \), and choose its interior node uniformly at random, and stop there.

It can be checked, using arguments similar to that in proof of Lemma 4, that the distribution of the stopped node is precisely \( \hat{\pi} \). Also, we can show that the expected length of this stopping rule is \( O(R + D) = O(D) \). This is primarily true because the probability of getting on a directed path \( P'_{wy} \) at \( w \) is \( \Theta(\varepsilon/(R + D)) \).

B. Proof of Lemma 8

In [6], they consider the following multi-commodity flow problem on \( G \) with the capacity constraint on edge \((u, v) \in E \) given by \( Q_{uv} \).

\[
\begin{align*}
\text{minimize} & \quad K \\
\text{subject to} & \quad \text{val}(f_{st}) = \pi_s \pi_t, \quad \forall s, t, \\
& \quad \sum_{s,t} f_{st}(ij) \leq KQ_{ij}, \quad \forall (i, j) \in E, \\
& \quad \sum_{t} \text{cost}(f_{st}) \leq K\pi_s, \quad \sum_{s} \text{cost}(f_{st}) \leq K\pi_t, \quad \forall s, t.
\end{align*}
\]

Let \( C \) be the optimal solution of the above problem. It is easy to see that \( C \geq 1/\Phi(P) \). Further, if \( P \) is reversible, then result of Leighton and Rao [16] on the approximate multi-commodity implies that

\[
C = O\left(\frac{1}{\Phi(P)} \log \frac{1}{\pi_0}\right) = \tilde{O}\left(\frac{1}{\Phi(P)}\right)
\]

Let the optimal multi-commodity flow of the above problem be \( F_1 \), and we can think of \( F_1 \) as a weighted collection of directed paths. In [6], authors modified \( F_1 \), and got a new multi-commodity flow \( F_2 \) that has the same amount of \( s - t \) flows as \( F_1 \), while its congestion and path length at most \( 12C \).

To prove Lemma 8, we will study the existence of the specific \( k \)-commodity flow with short path lengths. For this, we will use a balanced multi-commodity flow, which is a multi-commodity flow with the following condition for the amount of flows:

\[
\text{val}(f_{st}) = g(s, t), \quad \forall s, t,
\]

and \( g(s, t) \) satisfies the balanced condition:

\[
\sum_{t} g(s, t) \leq \pi_s, \quad \sum_{s} g(s, t) \leq \pi_t, \quad \forall s, t.
\]
Therefore, $F_1$ and $F_2$ are also balanced multi-commodity flows with $g(s, t) = \pi_s \pi_t$. Given a multi-commodity flow $f$, let $C(f)$ be its congestion and $D(f)$ be the length of the longest flow-path. Then, the flow number $T$ is defined follows:

$$T = \min_f \{ \max \{C(f), D(f)\} \},$$

where the minimum is taken over all balanced multi-commodity flows with $g(s, t) = \pi_s \pi_t$. Hence, as stated earlier, $T \leq 12C$ (from $F_2$). The following claim appears in [15]:

Claim 11: (Claim 2.2 in [15]) For any $g(s, t)$ satisfying the balanced condition, there exists a balanced multi-commodity flow $f$ with $g(s, t)$ such that $\max\{C(f), D(f)\} \leq 2T$.

Therefore, Lemma 8 is derived directly from Claim 11, because the flow number is less than $12C = \tilde{O}(1/\Phi(P))$ and the flow considered in Lemma 8 is a balanced multi-commodity flow.

C. Proof of Lemma 9

1. Preliminaries: If $P$ is reversible, one can view $P$ as a self-adjoint operator on a suitable inner product space and this permits us to use the well-understood spectral theory of self-adjoint operators. It is well-known that $P$ has $n = |V|$ real eigenvalues $1 = \lambda_0 > \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n-1} > -1$. The $\varepsilon$-mixing time $\tau_2(\varepsilon)$ is related as

$$\tau_2(\varepsilon) \leq \left[ \frac{1}{\lambda_P} \log \frac{1}{\varepsilon \sqrt{\pi_0}} \right],$$

where $\lambda_P = 1 - \max\{ |\lambda_1|, |\lambda_{n-1}| \}$. The $\lambda_P$ is also called the spectral gap. When, $P$ is non-reversible we consider $PP^\ast$. It is easy to see that the Markov chain with $PP^\ast$ as transition matrix is reversible. Let $\lambda_{PP^\ast}$ be the spectral gap of this reversible Markov chain. Then, the mixing time of original Markov chain (with transition matrix $P$) is bounded above as:

$$\tau_2(\varepsilon) \leq \left[ \frac{2}{\lambda_{PP^\ast}} \log \frac{1}{\varepsilon \sqrt{\pi_0}} \right].$$

2. Some lemmas: First, we state two useful Lemmas.

Lemma 12: Let $P_1, P_2$ be reversible Markov chains with their stationary distributions $\pi_1, \pi_2$ respectively. Suppose there exist positive constants $\alpha, \beta, c, d$ such that $P_1 \geq \alpha P_2$, $P_1 \geq \beta I$, and $c\pi_2 \leq \pi_1 \leq d\pi_2$. Then,

$$\lambda_{P_1} \geq \min\left( \frac{\alpha c}{d^2} \lambda_{P_2}, 2\beta \right).$$

Proof: From the min-max characterization of the spectral gap (see, e.g., page 176 in [11]) for the reversible Markov chain, it follows that

$$\lambda_{P_1} = \inf_{\psi: V \to \mathbb{R}} \left( \sum_{i,j \in V} \psi(i) - \psi(j) \right)^2 \frac{(\pi_1)_i (P_1)_{ij}}{\sum_{i,j \in V} \left( \psi(i) - \psi(j) \right)^2 (\pi_1)_i (\pi_1)_j} \geq \left( \frac{\alpha c}{d^2} \right) \inf_{\psi: V \to \mathbb{R}} \left( \sum_{i,j \in V} \psi(i) - \psi(j) \right)^2 \frac{(\pi_2)_i (P_2)_{ij}}{\sum_{i,j \in V} \left( \psi(i) - \psi(j) \right)^2 (\pi_2)_i (\pi_2)_j} = \left( \frac{\alpha c}{d^2} \right) \lambda_{P_2}.$$

The smallest eigenvalue of $P_1$ is greater than $2\beta - 1$ because $P_1 \geq \beta I$. So, the distance between the smallest eigenvalue and $-1$ is greater than $2\beta$. This completes the proof of Lemma 12.

Lemma 13: Let $P_1, P_2$ be Markov chains with their stationary distributions $\pi_1, \pi_2$ respectively. Now, suppose $P_2$ is reversible as well. ($P_1$ is not necessarily reversible.) If there exist positive constants $\alpha, \beta, c, d$ such that $P_1 \geq \alpha P_2$, $P_1 \geq \beta I$ and $c\pi_2 \leq \pi_1 \leq d\pi_2$. Then,

$$\lambda_{P_1 P_1^\ast} \geq \min\left( \frac{\alpha \beta c}{d^2} \lambda_{P_2}, 2\beta^2 \right).$$

Proof: $P_1 P_1^\ast$ is a reversible Markov chain which has $\pi_1$ as its stationary distribution. Because $P_1^\ast \geq \beta I$, $P_1 P_1^\ast \geq \alpha P_2 P_1^\ast \geq \alpha \beta P_2$. Also, $P_1 P_1^\ast \geq \beta^2 I$. Now, the proof of Lemma 13 follows from Lemma 12.
3. Completing proof of Lemma 9: By property of expander, we have $\lambda_{PEx} = \Omega(1)$. Therefore, to prove Lemma 9, it is sufficient to show that

$$\hat{H} = O\left( \frac{\hat{C}}{\lambda_{PEx}} \log(1/\pi_0) \right).$$

First, note that for any node $i \in V$ (i.e. original node $i$ in graph $G$),

$$\frac{1}{2} \pi_i \leq \hat{\pi}_i \leq \pi_i. \quad \tag{8}$$

Now, under the lifted Markov chain the probability of getting on any directed path $P'_r$ starting at $i$ is

$$\hat{P}_{ij} = \frac{\hat{Q}_{ij}}{\hat{\pi}_i} = \frac{w_r}{2\hat{C}\hat{\pi}_i}.$$  

Hence the probability of getting on any directed path starting at $i$ is

$$\sum_{r: P'_r \text{ starts at } i} \frac{w_r}{2\hat{C}\hat{\pi}_i} = \frac{1}{2\hat{C}\hat{\pi}_i} \sum_{r: P'_r \text{ starts at } i} w_r = \frac{\pi_i}{2\hat{C}\hat{\pi}_i}. \quad \tag{9}$$

From (8), this is bounded between $\frac{1}{2\hat{C}}$, and $\frac{1}{\hat{C}}$.

To study the $\hat{H}$, we will focus on the induced random walk (or Markov chain) on original nodes $V \subset \hat{V}$ by the lifted Markov chain $\hat{P}$. Let $\hat{P}^V$ be the transition matrix of this induced random walk. Then,

$$\hat{P}_{ij}^V = \hat{P}_{ij} + \sum_{r: P'_r \text{ goes from } i \text{ to } j} \frac{w_r}{2\hat{C}\hat{\pi}_i}. \quad \tag{10}$$

Now, $\hat{P}^V \geq \hat{P} \geq I/4$, because $\hat{P}_{ii} = Q_{ii}/\hat{\pi}_i \geq \hat{Q}_{ii}/2\hat{\pi}_i = P_{ii}\pi_i/2\hat{\pi}_i \geq P_{ii}/2 \geq I/4$. Here we have assumed that $P \geq I/2$ as discussed earlier. Now,

$$\hat{P}_{ij}^V \geq \frac{1}{2\hat{C}\hat{\pi}_i} \sum_{r: P'_r \text{ goes from } i \text{ to } j} w_r = \frac{\pi_i P_{ij}^{Ex}}{2\hat{C}\hat{\pi}_i} \geq \frac{1}{2\hat{C}} P_{ij}^{Ex}. \quad \tag{11}$$

And, its stationary distribution $\hat{\pi}^V$ is: $\hat{\pi}_i^V = \frac{\hat{\pi}_i}{\pi_i}$. Therefore, by (8) we have $\frac{1}{2} \pi_i \leq \hat{\pi}_i \leq 2\pi_i$. Now, we can apply Lemma 13 to obtain the following:

$$\lambda_{\hat{P}^V(\hat{\pi}^V)} = \Omega\left( \frac{1}{\hat{C}} \lambda_{P_{Ex}} \right). \quad \tag{9}$$

Now, we are ready to design a stopping rule $\Gamma$, that will imply that the desired bound on $\hat{H}$ as claimed in Theorem 3. Now, the stopping rule:

(i) Walk until visiting old nodes of $V \subset \hat{V}$ for $t$ times, where $t = \left\lceil 2\log(2/\pi_0^V)/\lambda_{\hat{P}^V(\hat{\pi}^V)} \right\rceil$. Let this $t$-th old node be denoted by $X$.

(ii) Stop at $X$ with probability $1/2$.

(iii) Otherwise, continue walking until getting onto any directed path $P'_r$; choose an interior node $Y$ of $P'_r$ uniformly at random and stop at $Y$.

Given (9) and the relation between mixing time and $\lambda_{\hat{P}^V(\hat{\pi}^V)}$, it follows that after time $t$ as defined above the Markov chain, restricted to old nodes $V$, has distribution close to $\hat{\pi}^V$. Therefore,

$$|\Pr(X = w) - \hat{\pi}_w^V| \leq \frac{\hat{\pi}_w^V}{2}, \quad \forall w \in V.$$  

According to the above stopping rule, we stop at an old node $w$ with probability $1/2$. Therefore, for any $w \in V$, we have that the stopping time $\Gamma$ stops at $w$ with probability at least $\hat{\pi}_w^V/4 \geq \pi_w/8 \geq \pi_w/8$. With probability $1/2$, the rule does not stop at the node $X$. Let $w^k$ be the $k$-th point in the walk starting from $X$. Because at any old node $\hat{i}$, the probability of getting on any directed path is between $\frac{1}{2\hat{C}}$ and $\frac{1}{\hat{C}}$, a coupling argument shows that for
If \( w \) is a new point on the directed path \( P' \) which connects the old node \( i \) to \( j \). Then,

\[
\Pr(\Gamma \text{ stop at } w) \geq \frac{1}{2} \sum_{k=0}^{\infty} \text{Prob}(w^k = i|w^0, \ldots, w^k \text{ are old points}) \times \Pr(\text{at } i, \text{ get on the path } P') \times \frac{1}{\ell_r}
\]

\[
\geq \frac{1}{2} \sum_{k=0}^{\infty} \left( 1 - \frac{1}{C} \right)^k \frac{1}{2} \frac{\pi_i}{2C\pi_i} \frac{w_r}{C} \sum_{k=0}^{\infty} \left( 1 - \frac{1}{C} \right)^k
\]

\[
= \frac{w_r}{16C} \frac{1}{8\pi_w}
\]

The average length of this stopping rule is \( O(t + \tilde{C}) \). By (9),

\[
O(t + C) = O \left( \left[ \frac{2}{\lambda_{\tilde{P}V}(\tilde{P}V)} \log(2/\pi_0) \right] + \tilde{C} \right) = O \left( \frac{\tilde{C}}{\lambda_{PEx}} \log(1/\pi_0) \right).
\]

Thus, we have established that the stopping rule \( \Gamma \) has average length \( O(C\log 1/\pi_0) \) and the distribution of the stopping nodes is \( \Omega(\tilde{\pi}) \). Therefore, using the fill-up lemma stated in [1], it follows that \( \tilde{\mathcal{H}} = O(C\log 1/\pi_0) \).

**D. Proof of Lemma 10**

Here, we establish the bound on the size of lifted chain as described in Section IV. We want to establish that the size of the lifted chain in terms of number of edges, i.e. \(|\tilde{E}| = \tilde{O}(|E|/\Phi(P)) \). Note that, the lifted graph \( \tilde{G} \) is obtained by adding the paths that appeared in the solution of the multi-commodity flow problem. Therefore, to establish the desired bound we need to establish bound on the number of distinct paths as well as their lengths.

To this end, let us re-formulate the multi-commodity flow based on expander \( G^{Ex} \) as follows. For each \((s, t) \in E^{Ex}\), we add a flow between \( s \) and \( t \). Let this flow be routed along possibly multiple paths. Let \( P_{stj} \) denote the \( j^{th} \) path from \( s \) to \( t \), it’s length \( \ell_{stj} \) is at most \( \tilde{C} \) as per discussion in Lemma 8 and \( x_{stj} \) be the amount of flow sent along this path. Let the overall solution, denoted by \( \{(P_{r}, w_r)\} \) gives a feasible solution in the following polytope with \( x_{stj} \) as its variables:

\[
\sum_j x_{stj} = \pi_s P_{st}^{Ex}, \quad \forall(s, t) \in E^{Ex}
\]

\[
\sum_{st \in E^{Ex}} \sum_{j \in E_{P_{stj}}} x_{stj} \leq \tilde{C}Q_e, \quad \forall e \in E
\]

\[
x_{stj} \geq 0 \quad \forall s, t, j.
\]

Clearly, any feasible solution in this polytope, say \( \{(P_{r}, w_r)\} \), will work for our lifting construction. Now, the size of its support set is \(|\{(P_{r}, w_r)\}| \). If we consider the extreme point of this polytope, the size of its support set is at most \(|E^{Ex}| + |E| = O(|E|) \) because the extreme point is an unique solution of sub-collection of linear constraints in this polytope. Hence, if we choose such an extreme point \( \{(P_{r}, w_r)\} \) for our lifting, the size of our lifted chain \( |\tilde{E}| \) is at most \( O(C|E|) \) since each path is of length \( O(\tilde{C}) \). Thus, we have established that the size of the lifted Markov chain is at most \( O(C|E|) = \tilde{O}(|E|/\Phi(P)) \).