Expansion Testing using Quantum Fast-Forwarding and Seed Sets

Simon Apers

CWI (The Netherlands) and ULB (Belgium)

Expansion testing aims to decide whether an n-node graph has expansion at least Φ, or is far from any such graph. We propose a quantum expansion tester with complexity $\widetilde{O}(n^{1/3}\Phi^{-1})$. This accelerates the $\widetilde{O}(n^{1/2}\Phi^{-2})$ classical tester by Goldreich and Ron [Algorithmica '02], and combines the $\widetilde{O}(n^{1/3}\Phi^{-2})$ and $\widetilde{O}(n^{1/2}\Phi^{-1})$ quantum speedups by Ambainis, Childs and Liu [RANDOM '11] and Apers and Sarlette [QIC '19], respectively. The latter approach builds on a quantum fast-forwarding scheme, which we improve upon by initially growing a seed set in the graph. To grow this seed set we use a so-called evolving set process from the graph clustering literature, which allows to grow an appropriately local seed set.

1 Introduction and Summary

The (vertex) expansion of a graph is a measure for how well connected the graph is. For an undirected graph $G = (\mathcal{V}, \mathcal{E})$, with $|\mathcal{V}| = n$ and $|\mathcal{E}| = m$, it is defined as

$$\Phi(G) = \min_{\mathcal{S}\subset\mathcal{V}:|\mathcal{S}|\leq n/2} \frac{|\partial\mathcal{S}|}{|\mathcal{S}|},$$

where $\partial\mathcal{S}$ is the set of nodes in $\mathcal{V}\backslash\mathcal{S}$ that have an edge going to $\mathcal{S}$. See [LR99] for a discussion on the relevance of expansion for a range of graph approximation algorithms, and [HLW06] for a survey on expander graphs and their applications. Since exactly determining $\Phi(G)$ is an NP-hard problem [LRV13], we consider the relaxed objective of *testing* the expansion. Goldreich and Ron [GR02, GR11] initially studied this problem in the bounded-degree model, where they proposed the following question: given query acces to G, does it have expansion at least some Φ, or is it far from any graph having expansion $\widetilde{\Omega}(\Phi^2)$? In this model, given graphs G and G' with degree bound d, G is ϵ-far from G' if at least ϵnd edges have to be added or removed from G to obtain G'. They proved an $\Omega(n^{1/2})$ lower bound on the query complexity of this problem, and proposed an elegant tester based on random walk collision counting with complexity[1]

$$\widetilde{O}(n^{1/2}\Phi^{-2}).$$

In rough strokes, the algorithm picks a uniformly random node, and counts collisions between $\widetilde{O}(n^{1/2})$ independent random walks of length $\widetilde{O}(\Phi^{-2})$ all starting from this node. If the graph is far from being an expander, then the random walk will get stuck in certain low-expansion subsets, leading to an increased number of collisions. The graph is hence rejected if the number of collisions exceeds some constant.

Simon Apers: smgapers@gmail.com, Most of the work was done while part of the CWI-Inria International Lab.

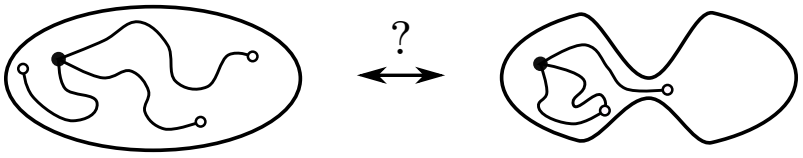

Figure 1: The GR tester counts collisions between independent random walks starting from some seed node. Low expansion of the graph results in an increased number of collisions.

Goldreich and Ron had to base the correctness of their tester on a certain unproven combinatorial conjecture. However, in later works by Czumaj and Sohler [CS10], Kale and Seshadhri [KS11] and Nachmias and Shapira [NS10] the correctness was unconditionally established. The ideas underlying this tester and its analysis were more recently extended towards testing the k-clusterability of a graph [CPS15, CKK+18], which is a multipartite generalization of the expansion testing problem.

In this work we consider the expansion testing problem in the quantum setting, where we allow to perform queries in superposition. We refer to the nice survey by Montanaro and de Wolf [MdW16] for a general overview of quantum property testing. Ambainis, Childs and Liu [ACL11] were the first to describe a quantum algorithm for expansion testing. The gist of their algorithm is to combine an appropriate derandomization of the GR tester with Ambainis' quantum algorithm for element distinctness [Amb07]. The latter allows to count collisions among the set of $\widetilde{O}(n^{1/2})$ random walk endpoints using only $\widetilde{O}(n^{1/3})$ quantum queries. The improved complexity of their quantum expansion tester is

$$\widetilde{O}(n^{1/3}\Phi^{-2}).$$

In addition they proved an $\widetilde{\Omega}(n^{1/4})$ lower bound on the quantum query complexity.

In later work of the current author together with Sarlette [AS19], as well as in the current work, a very different approach is taken. Quantum walks, which form the quantum counterpart of random walks, are used to explore the graph. Rather than picking random neighbors, a quantum walk explores a graph through quantum queries to its neighborhood. In particular, this allows to create a "quantum sample" that appropriately encodes the random walk distribution. As we detail in Section 1.1 below, we can then use standard tools from quantum algorithms to estimate the random walk collision probability. In [AS19] we introduced a new quantum walk technique called "quantum fast-forwarding" (QFF) that allows to approximately prepare these quantum samples in the square root of the random walk runtime. This yielded a new quantum expansion tester with complexity

$$\widetilde{O}(n^{1/2}\Phi^{-1}),$$

quadratically improving the dependency of the GR tester on Φ, which corresponds to the random walk runtime. Up to this work, this left the problem of quantum expansion testing with two different testers with a complementary speedup. In this work, however, we present a new quantum tester which closes this gap. Essentially we improve the QFF tester from [AS19] by initially doing some classical work in the graph: from the initial node v, we first grow a local node subset or "seed set" of size $n^{1/3}$. In earlier work by the author [A19] it was already shown that such seed sets allow to more efficiently create quantum samples, essentially by improving the projection of the initial state on the final quantum sample. Indeed, starting from this seed set, rather than directly from v, we can run the

[1] In this section we hide polynomial dependencies on $\log n$, the degree bound d_M of the graph, and the distance parameter ϵ. In the rest of the paper, $\widetilde{O}$ simply hides any poly-logarithmic dependencies.

Goldreich and Ron [GR02]	$\widetilde{O}(n^{1/2}\Phi^{-2})$ (conj.)	RW collision counting
Czumaj and Sohler [CS10], Kale and Seshadhri [KS11], Nachmias and Shapira [NS10]	$\widetilde{O}(n^{1/2}\Phi^{-2})$	prove GR conjecture
Ambainis, Childs and Liu [ACL11]	$\widetilde{O}(n^{1/3}\Phi^{-2})$ (q)	quantum element distinctness
Apers and Sarlette [AS19]	$\widetilde{O}(n^{1/2}\Phi^{-1})$ (q)	QFF
this work	$\widetilde{O}(n^{1/3}\Phi^{-1})$ (q)	QFF and seed sets

Table 1: Complexity for expansion testing. (q) denotes quantum complexity.

QFF tester with an improved complexity $\widetilde{O}(n^{1/3}\Phi^{-1})$. To prove correctness of the tester, we must ensure that if the initial node v is inside some low-expansion set, then the seed set largely remains inside that set. Thereto we borrow a so-called "evolving set process" from the local graph clustering literature [AGPT16], allowing to grow such a set in complexity $\widetilde{O}(n^{1/3}\Phi^{-1})$. This allows to prove our main result:

Theorem 1. *There exists a quantum expansion tester with complexity* $\widetilde{O}(n^{1/3}\Phi^{-1})$.

The resulting speedup combines the quantum speedups of [ACL11] and [AS19]. To summarize, we gather the different algorithms and approaches in Table 1.

1.1 QFF Tester

Our tester builds on the QFF tester from [AS19], hence we describe this tester first. Let P denote the random walk (RW) transition matrix, and $P^t|v\rangle$ the t-step RW probability distribution[2] starting from a node v. The tester builds on the observation that the squared 2-norm $\|P^t|v\rangle\|^2$ exactly equals the collision probability of a pair of random walks:

$$\|P^t|v\rangle\|^2 = \sum_{u\in\mathcal{V}} P^t(u,v)^2 = \sum_{u\in\mathcal{V}} \mathbb{P}(X_t = u \,|\, X_0 = v)^2,$$

where we let X_t denote the random walk position at time step t. Hence, we can estimate the collision probability between the t-step RW endpoints simply by estimating $\|P^t|v\rangle\|^2$. This we can do rather straightforwardly using quantum algorithms, in particular making use of quantum walks (QWs). Starting from an initial node v of the graph, a QW allows to generate the quantum sample

$$|\psi_t\rangle = P^t|v\rangle + |\Gamma\rangle,$$

which encodes the RW probability distribution $P^t|v\rangle$ as one of its component, with $|\Gamma\rangle$ some auxiliary garbage component that is orthogonal to the RW component, and which we will not care about. Given the ability to generate such quantum samples, we can then use a standard quantum routine called *quantum amplitude estimation* to estimate the norm $\|P^t|v\rangle\|^2$ of the RW component. Now, similarly to the GR tester, if we set $t \in O(\Phi^{-2})$ then we can reject the graph if $\|P^t|v\rangle\|^2$, and hence the collision probability, is larger than some threshold.

[2] We use the ket-notation $|v\rangle$ to simply denote the indicator vector on node v.

The amplitude estimation routine requires $\widetilde{O}(\|P^t|v\rangle\|^{-1})$ quantum samples, so that the complexity of this quantum expansion tester is $\widetilde{O}(\|P^t|v\rangle\|^{-1}\,\mathrm{QS}_t)$, where QS_t denotes the quantum complexity of creating the quantum sample $|\psi_t\rangle$. If the graph is regular[3] then P has a uniform stationary distribution, i.e., the vector $|\pi\rangle = n^{-1/2}\sum_{u\in\mathcal{V}}|u\rangle$ is the unique eigenvalue-1 eigenvector of P. This allows to bound

$$\|P^t|v\rangle\| \geq |\langle\pi|v\rangle| = n^{-1/2}, \tag{1}$$

so that $\widetilde{O}(\|P^t|v\rangle\|^{-1}\,\mathrm{QS}_t) \in \widetilde{O}(n^{1/2}\,\mathrm{QS}_t)$. In order to bound QS_t, we can use an existing QW approach by Watrous [Wat01] which gives $\mathrm{QS}_t \in O(t)$. Since we choose $t \in \widetilde{O}(\Phi^{-2})$, this yields a complexity $\widetilde{O}(n^{1/2}\Phi^{-2})$, thus giving no speedup with respect to the GR tester. In [AS19] however, we introduced a more involved QW technique called *quantum fast-forwarding* (QFF). Building on a Chebyshev truncation of the P^t operator, this technique allows to quadratically improve the complexity to $\mathrm{QS}_t \in O(t^{1/2})$, resulting in a complexity

$$\widetilde{O}(n^{1/2}\Phi^{-1}).$$

Given that the GR tester has complexity $\widetilde{O}(n^{1/2}\Phi^{-2})$, this yields a complementary speedup to the $\widetilde{O}(n^{1/3}\Phi^{-2})$ expansion tester in [ACL11]. Whereas their speedup follows from a quantum routine for accelerating the collision counting procedure, the speedup in the QFF tester follows from accelerating the random walk runtime.

1.2 QFF Tester with Seed Sets

In this paper we refine the QFF tester, improving its complexity to $\widetilde{O}(n^{1/3}\Phi^{-1})$. We improve its suboptimal $n^{1/2}$-dependency by initially constructing or "growing" a seed set around the initial node, from which we then run the QFF tester. This idea is derived from earlier work of the author [A19], where seed sets are used to create a superposition over the edges of a graph, leading to a similar speedup. The main insight is derived from the bound in (1), showing that the suboptimal $n^{1/2}$-dependency stems from a small projection of the initial state $|v\rangle$ onto the uniform superposition $|\pi\rangle$. Growing a seed set allows to improve this dependency: if we grow a set $\mathcal{S} \subseteq \mathcal{V}$ from v, and we use the quantum superposition $|\mathcal{S}\rangle = |\mathcal{S}|^{-1/2}\sum_{u\in\mathcal{S}}|u\rangle$ as an initial state, this bound becomes

$$\|P^t|S\rangle\| \geq |\langle\pi|\mathcal{S}\rangle| = |\mathcal{S}|^{1/2}n^{-1/2}.$$

This suggests the following new tester: (i) pick a uniformly random node v, (ii) grow a seed set $\mathcal{S}$ from v of appropriate size, and (iii) create $\widetilde{O}(n^{1/2}|\mathcal{S}|^{-1/2})$ QW samples $|\psi_t\rangle = P^t|\mathcal{S}\rangle + |\Gamma\rangle$, allowing to estimate $\|P^t|\mathcal{S}\rangle\|$. Assuming that the construction of $\mathcal{S}$ requires $|\mathcal{S}|$ queries, and momentarily ignoring the Φ-dependency, this tester has a combined complexity of $\widetilde{O}(|\mathcal{S}|+\|P^t|S\rangle\|^{-1}) \in \widetilde{O}(|\mathcal{S}|+n^{1/2}|\mathcal{S}|^{-1/2})$. If we choose $|\mathcal{S}| = n^{1/3}$, this becomes $\widetilde{O}(n^{1/3})$ as we aimed for.

Using a similar reasoning as before, we again wish to reject the graph if our estimate is larger than some threshold. Indeed, as depicted in Figure 2, if the seed set is localized in some low-expansion set, then the 2-norm $\|P^t|S\rangle\|$ will be larger than when the graph has no low-expansion sets. The difficulty however is to ensure that the seed set $\mathcal{S}$, when grown from some initial node v in a low-expansion set, effectively remains inside that set. If this is not the case, then a RW from $\mathcal{S}$ will no longer be stuck in the low-expansion set, thus no longer giving rise to an increased 2-norm. As a consequence, we cannot simply

[3] Later on we adapt the graph to ensure this.

use a breadth-first search from v, as we did in [A19]: a BFS might exit a low-expansion set more easily than a random walk. Luckily, however, the problem of locally exploring a low-expansion set (or "cluster") turns out to be well-studied under the name "local graph clustering" [ST13, ACL06, AGPT16, OSV12].

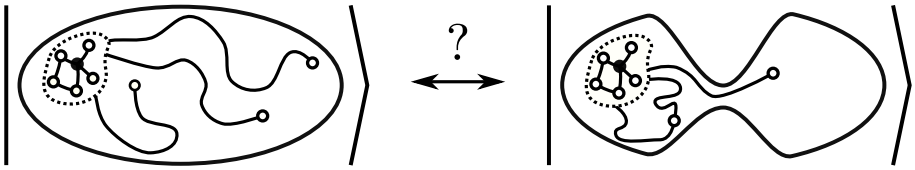

Figure 2: The new tester classically grows an appropriately local seed set around the initial node. From this set a quantum sample can be generated more efficiently. We use an evolving set process to ensure that the seed set mostly remains inside the initial low-expansion set.

In particular, we can use a so-called "evolving set process" (ESP) as used by Andersen, Oveis-Gharan, Peres and Trevisan in [AGPT16]. An ESP is a Markov chain on subsets of the nodes, which evolves by expanding or contracting its boundary based on the RW behavior on the graph. Given an initial node v inside a low-expansion set, they simulate an ESP to explicitly retrieve this set. Since we are interested in growing a potentially much smaller seed set $\mathcal{S}$ inside the cluster, we slightly adapt their algorithm, leading to the following result. The algorithm either returns a low-expansion set, allowing to immediately reject the graph, or it returns an appropriate seed set.

Proposition 1. *Fix a parameter $M \geq 0$. Given a random node v from a set $\mathcal{S}'$ of expansion $\widetilde{O}(\Phi^2)$, we can use an ESP to return a set $\mathcal{S}$ such that with constant probability either $\Phi(\mathcal{S}) < \Phi$ or $|\mathcal{S}| \geq M$ and $|\mathcal{S} \cap \mathcal{S}'|/|\mathcal{S}| \in \Omega(1)$. The complexity of generating this set is $\widetilde{O}(M\Phi^{-1})$.*

Building on this tool, we can now sketch our new quantum tester, summarized in Algorithm 1. Since the ESP process requires $\widetilde{O}(n^{1/3}\Phi^{-1})$ steps by the above proposition, and estimating $\|P^t|\mathcal{S}\rangle\|$ requires $\widetilde{O}(\|P^t|\mathcal{S}\rangle\|\Phi^{-1}) \in \widetilde{O}(n^{1/3}\Phi^{-1})$ steps, we retrieve the promised tester complexity $\widetilde{O}(n^{1/3}\Phi^{-1})$.

Algorithm 1 Quantum Expansion Tester

1: select a uniformly random starting node v
2: grow a seed set $\mathcal{S}$ from v using an ESP
3: **if** $\Phi(\mathcal{S}) < \Phi$ **then** `reject`
4: use quantum amplitude estimation to estimate $\|P^t|\mathcal{S}\rangle\|$ for $t \in \widetilde{O}(\Phi^{-1})$
5: **if** $\|P^t|\mathcal{S}\rangle\|$ too large **then** `reject` **else** `accept`

1.3 Open Questions

We finish this section by discussing some open questions related to this work.

- In [A19] a breadth-first search is used to grow a seed set $\mathcal{S}$, requiring a number of steps $\widetilde{O}(|\mathcal{S}|)$. In the current work we use a more refined ESP algorithm to grow $\mathcal{S}$, which in particular ensures that the set remains inside some low-expansion subset (say with expansion at most Φ). This procedure however requires an increased number of steps $\widetilde{O}(|\mathcal{S}|\Phi^{-1})$. We leave it as an open question whether such an appropriate set can be grown in $\widetilde{O}(|\mathcal{S}|)$ steps. The complexity of the tester then becomes $\widetilde{O}(|\mathcal{S}| +$

$n^{1/2}|\mathcal{S}|^{-1/2}\Phi^{-1})$. Setting $|\mathcal{S}| = n^{1/3}\Phi^{-2/3}$ this would lead to an improved complexity $\widetilde{O}(n^{1/3}\Phi^{-2/3})$.

- The use of an ESP for the expansion testing problem could also be useful for improving the Φ-dependency of the classical GR tester. If we could for instance grow a pair of seed sets, both of size $n^{1/2}$, that behave to some extent as "random subsets" of a local cluster, then we could simply count collisions between these sets, thus avoiding the use of random walks. A higher number of collisions would then again signal a low expansion of the graph. Ideally an ESP-like procedure would allow to grow these sets in $\widetilde{O}(n^{1/2}\Phi^{-1})$ steps, improving on the $\widetilde{O}(n^{1/2}\Phi^{-2})$ complexity of the GR tester.

- Clusterability testing, as recently studied in [CPS15, CKK+18], uses very similar techniques to the GR expansion tester. It seems feasible that we can use the techniques from this paper to similarly improve on these testers.

- Goldreich and Ron [GR02] proved a classical lower bound $\Omega(n^{1/2})$ for expansion testing, suggesting that their tester has an optimal dependency on n. In the quantum setting however, the only known lower bound is $\Omega(n^{1/4})$ as proven by Ambainis, Childs and Liu [ACL11], thus leaving a large gap to all current quantum testers, which have a $\widetilde{O}(n^{1/3})$-dependency. While our work does not provide any new insights towards closing this gap, we do feel that this is an interesting question to resolve.

2 Preliminaries

In this section we formalize the computational model, the query model and the definition of an expansion tester. We also describe some necessary random walk properties, and define the notion of a quantum walk.

2.1 Complexity, Computational Model and QRAM

The *quantum query complexity* of an algorithm simply denotes the number of quantum queries that the algorithm makes to the input (see Section 2.2 for details).

In contrast, the actual runtime or *complexity* of the algorithm is defined with respect to a computational model. We specify our computation model as a system that

1. can run quantum routines on $O(\log n)$ qubits, where n is the number of nodes in the input graph,

2. can make quantum queries to the input (see Section 2.2), and

3. has access to a quantum-read/classical-write RAM (QRAM) of $\widetilde{O}(n^{1/3})$ classical bits. A single QRAM operation corresponds to either (i) classically writing a bit to the QRAM, or (ii) making a quantum query to bits stored in the QRAM.

By the *complexity* of an algorithm we denote the total number of (i) queries, (ii) elementary classical and quantum operations (gates), and (iii) QRAM operations. By definition, the complexity of an algorithm forms an upper bound on the query complexity of an algorithm, irrespective of the computational model or QRAM assumptions.

Let $\mathcal{S} \subset \mathcal{V}$ be a subset of nodes. As in [A19], we can use the QRAM to efficiently create, and reflect around, the quantum state $|\mathcal{S}\rangle = |\mathcal{S}|^{-1/2}\sum_{v\in\mathcal{S}}|v\rangle$. To this end, we rely on the following theorem by Kerenidis and Prakash [KP17].

Theorem 2 ([KP17, Theorem 15])**.** *Assume that we are given a subset $\mathcal{S} \subset \mathcal{V}$. With a one-off preprocessing cost of $\widetilde{O}(|\mathcal{S}|)$ QRAM operations, we can repeatedly (i) create the quantum state $|\mathcal{S}\rangle$, and (ii) reflect around the quantum state $|\mathcal{S}\rangle$, both using $\widetilde{O}(1)$ QRAM operations per repetition.*

We make two final comments on our QRAM assumption, as it is often subject of debate. First, such an assumption is native to any quantum algorithm that makes use of quantum-accessible classical input or memory. We note that the actual qubits that a QRAM query acts on (i.e., the superposition of queried addresses plus the target register) is only logarithmic in the number of stored bits. In that sense our memory requirement is weaker than for instance that of Ambainis's element distinctness algorithm [Amb07] and its application in the quantum expansion tester of [ACL11], which effectively require a polynomial-sized memory of qubits. Second, as already pointed out above, the complexity is an upper bound on the query complexity, irrespective of the computational model or QRAM assumptions. In that sense, our work improves on the query complexity of the former works [ACL11, AS19], also without these assumptions.

2.2 Query Model and Property Testing

We are given query access to some undirected graph $G = (\mathcal{V}, \mathcal{E})$, with node set $\mathcal{V} = [n]$ and edge set $\mathcal{E}$. We denote $|\mathcal{E}| = m$. For any $v \in \mathcal{V}$, we let $d(v)$ denote the degree of v, the maximum degree $d_M = \max_{v\in\mathcal{V}} d(v)$, and $d(\mathcal{S}) = \sum_{v\in\mathcal{S}} d(v)$ denotes the total degree of a set $\mathcal{S} \subseteq \mathcal{V}$. We say that G has degree bound d if $d_M \leq d$. In the context of property testing of bounded-degree graphs [GR02], the following queries are allowed:

- *uniform node query*: return uniformly random node $v \in \mathcal{V}$
- *degree query*: given $v \in \mathcal{V}$, return degree $d(v)$
- *neighbor query:* given $v \in \mathcal{V}$, $k \in [d(v)]$, return k-th neighbor of v

Throughout the paper we will assume that G is regular. If this is not the case, then we can always modify the graph to ensure this: to any node i with degree $d(i) < d$ we add $d - d(i)$ parallel self loops. This effectively renders the graph regular, ensuring that a random walk converges to the uniform distribution, as we will require later. Notably this does not change the expansion of the graph. Goldreich and Ron [GR11] achieve the same effect by modifying the random walk rather than the graph, but modifying the graph will prove more elegant for our purpose.

Since we wish to study quantum algorithms, we will allow to perform degree and neighbor queries in superposition. To illustrate this, assume that a neighbor query, given $v \in \mathcal{V}$ and $k \in [d(v)]$, returns a node u. Using quantum notation, this is described as a unitary transformation

$$|v\rangle|k\rangle|x\rangle \mapsto |v\rangle|k\rangle|x + u\rangle,$$

where x is some arbitrary $\lceil \log n \rceil$-bit string, and "+" denotes addition modulo $\lceil \log n \rceil$. We can now imagine the first register being in a superposition $d(v)^{-1/2} \sum_{k\in[d(v)]} |v\rangle|k\rangle|x\rangle$, so that the query operation now becomes

$$\frac{1}{\sqrt{d(v)}} \sum_{i\in[d(v)]} |v\rangle|k\rangle|x\rangle \mapsto \frac{1}{\sqrt{d(v)}} \sum_{i\in[d(v)]} |v\rangle|k\rangle|x + u^{(k)}\rangle, \tag{2}$$

where we let $u^{(k)}$ denote the k-th neighbor of v. We will call a single such query a "quantum query". We refer the interested reader to the survey by Montanaro and de Wolf [MdW16] for more details on the quantum query model.

We will follow the property testing model for bounded-degree graphs by Goldreich and Ron [GR02]. Given two n-node graphs $G = ([n], \mathcal{E})$ and $G' = ([n], \mathcal{E}')$ with degree bound d, they define the relative distance between G and G' as the number of edges that needs to be added or removed to turn G into G', divided by the maximum number of edges nd. This is equal to $|\mathcal{E} \triangle \mathcal{E}'|/(nd)$, with $\triangle$ the symmetric difference between $\mathcal{E}$ and $\mathcal{E}'$. G is then said to be ϵ-far from G' if $|\mathcal{E} \triangle \mathcal{E}'|/(nd) \geq \epsilon$. When studying a certain property P of graphs, G is said to be "ϵ-far from having property P" if G is ϵ-far from any graph G' having property P.

2.3 Expansion Testing

We define the (vertex) expansion of a subset $\mathcal{S} \subset \mathcal{V}$ as $\Phi(\mathcal{S}) = |\partial\mathcal{S}|/|\mathcal{S}|$. Here $\partial\mathcal{S} = \{u \in \mathcal{S}^c \mid \exists v \in \mathcal{S} \text{ s.t. } (u, v) \in \mathcal{E}\}$ is the set of nodes in $\mathcal{S}^c$ that have an edge going to $\mathcal{S}$. The expansion of a graph G is then defined as

$$\Phi(G) = \min_{\mathcal{S} \subset \mathcal{V} : |\mathcal{S}| \leq n/2} \Phi(\mathcal{S}).$$

We consider the following definition of an expansion tester due to Czumaj and Sohler [CS10].

Definition 1. *An algorithm is a (Φ, ϵ)-expansion tester if there exists a constant $c > 0$, possibly dependent on d, such that given parameters n, d, and query access to an n-node graph with degree bound d it holds that*

- *if the graph has expansion at least Φ, then the algorithm outputs "`accept`" with probability at least $2/3$,*
- *if the graph is ϵ-far from any graph having expansion at least $c\Phi^2 \log^{-1}(dn)$, then the algorithm outputs "`reject`" with probability at least $2/3$.*

We note that this is a slightly more constrained definition than the one in e.g. [KS11, ACL11, AS19]. In these works the log-factor in the reject case is actually left as an additional free parameter μ, which is compensated in the runtime. While our algorithm might also work in that more general setting, we believe that the corresponding technicalities would go beyond the scope and main ideas of this paper, and we leave it as a minor open question. We also mention that in the traditional setting of property testing, the expression "$c\Phi^2 \log^{-1}(dn)$" in the second bullet should be replaced by "Φ". Although unproven, the relaxation in this definition seems necessary to allow for efficient (sublinear) testing using random walks. This is a consequence of the fact that the expansion only characterizes the random walk mixing behavior up to a quadratic factor. We stress that this quadratic gap is present in all works on expansion testing.

Apart from the vertex expansion, we also define the conductance. When studying random walks, this is often a slightly more appropriate measure. For a subset $\mathcal{S} \subset \mathcal{V}$ it is defined as $\phi(\mathcal{S}) = |\mathcal{E}(\mathcal{S}, \mathcal{S}^c)|/d(\mathcal{S})$, where $\mathcal{E}(\mathcal{S}, \mathcal{S}^c) = \{(u, v) \in \mathcal{E} \mid u \in \mathcal{S}, v \in \mathcal{S}^c\}$ denotes the set of edges between $\mathcal{S}$ to $\mathcal{S}^c$. The conductance of a graph G with m edges is then defined as $\phi(G) = \min_{\mathcal{S} \subset \mathcal{V} : d(\mathcal{S}) \leq m/2} \phi(\mathcal{S})$. If G is d-regular, as we will assume throughout the paper, this simplifies to $\phi(G) = \min_{\mathcal{S} \subset \mathcal{V} : |\mathcal{S}| \leq n/2} |\mathcal{E}(\mathcal{S}, \mathcal{S}^c)|/(d|\mathcal{S}|)$. Since $|\partial\mathcal{S}| \leq |\mathcal{E}(\mathcal{S}, \mathcal{S}^c)| \leq d|\partial\mathcal{S}|$, this allows to relate vertex expansion and conductance as follows:

$$\Phi(\mathcal{S})/d \leq \phi(\mathcal{S}) \leq \Phi(\mathcal{S}). \tag{3}$$

2.4 Random Walks

We will consider lazy random walks (RWs), described by a Markov chain on the node set. From any node the RW jumps with probability $1/2$ to any of its neighbors uniformly at random, and otherwise stands still. If we let $P(u,v)$ denote the RW transition probability from node v to node u, then $P(u,v) = 1/(2d(v))$ for $(v,u) \in E$, $P(u,v) = 1/2$ if $u = v$ and $P(u,v) = 0$ elsewhere. If the underlying graph is connected, then the RW converges to a unique limit distribution in which every node has a probability proportional to its degree. On a regular graph, this corresponds to a uniform distribution.

2.4.1 Diffusion Core

Central to the study of expansion testers is the so-called "diffusion core" of a set $\mathcal{S} \subseteq \mathcal{V}$. The diffusion core allows to lower bound the probability that a RW of given length stays entirely inside $\mathcal{S}$, as a function of its conductance $\phi(\mathcal{S})$. Let $\tau_v(\mathcal{S}^c)$ denote the escape time of $\mathcal{S}$ from v, i.e., the hitting time of a RW from $v \in \mathcal{S}$ to the complement $\mathcal{S}^c$. We then define the diffusion core of $\mathcal{S}$ as follows:

Definition 2. *For $\alpha, \beta > 0$, the (α,β)-diffusion core of $\mathcal{S}$ is defined as*

$$\mathcal{S}_{\alpha,\beta} = \{v \in \mathcal{S} \,|\, \mathbb{P}(\tau_v(\mathcal{S}^c) > \alpha\phi(\mathcal{S})^{-1}) \geq \beta\}.$$

Throughout we define the "canonical" diffusion core $\mathcal{S}_d = \mathcal{S}_{1/40,3/4}$. Using a reasoning similar to Spielman and Teng [ST13], we can lower bound the size of the diffusion core.

Lemma 1.

$$\frac{d(\mathcal{S}_{\alpha,\beta})}{d(\mathcal{S})} > 1 - \frac{\alpha}{2(1-\beta)}.$$

Proof. Let Y_v denote the event that $\tau_v(\mathcal{S}^c) > \alpha\phi(\mathcal{S})^{-1}$, let π denote the stationary distribution of the RW, and let $\pi_{\mathcal{S}}$ denote the distribution π conditioned on being in the set $\mathcal{S}$: $\pi_{\mathcal{S}}(v) = \mathbb{I}(v \in \mathcal{S})\pi(v)/\pi(\mathcal{S})$. From [ST13, Proposition 2.5] we know that $\mathbb{P}_{v\sim\pi_{\mathcal{S}}}(Y_v) \geq 1 - \alpha/2$. For all $v \notin \mathcal{S}_{\alpha,\beta}$, it holds by definition that $\mathbb{P}(Y_v) < \beta$, so that we can bound

$$\mathbb{P}_{v\sim\pi_{\mathcal{S}}}(Y_v) = \sum_{v\in\mathcal{S}} \pi_{\mathcal{S}}(v)\mathbb{P}(Y_v) < (1 - \pi_{\mathcal{S}}(\mathcal{S}_{\alpha,\beta}))\beta + \pi_{\mathcal{S}}(\mathcal{S}_{\alpha,\beta}).$$

Combined with the former inequality, and the fact that $\pi_{\mathcal{S}}(\mathcal{S}_{\alpha,\beta}) = d(\mathcal{S}_{\alpha,\beta})/d(\mathcal{S})$, this proves the claimed statement. □

This lemma implies that $d(\mathcal{S}_d)/d(\mathcal{S}) > 19/20$. As we will require this later, we also wish to prove something slightly stronger: there exists a subset $\mathcal{S}'$ of the diffusion core $\mathcal{S}_d$, from which we can bound the probability that a random walk stays inside the diffusion core, rather than only inside $\mathcal{S}$.

Lemma 2. *There exists a node subset $\mathcal{S}'$ of the diffusion core $\mathcal{S}_d$, with $d(\mathcal{S}') > d(\mathcal{S})/3$, from which a $(120\phi(\mathcal{S}))^{-1}$-step RW remains inside $\mathcal{S}_d$ with probability at least $9/10$:*

$$\forall v \in \mathcal{S}' : \quad \mathbb{P}(\tau_v(\mathcal{S}_d^c) > (120\phi(\mathcal{S}))^{-1}) \geq 9/10.$$

Proof. In the following we use the shorthand $\phi = \phi(\mathcal{S})$. We can set $\mathcal{S}'$ equal to the $(1/30, 39/40)$-diffusion core, $\mathcal{S}' = \mathcal{S}_{1/30,39/40}$. Using Definition 2 we see that $\mathcal{S}' \subseteq \mathcal{S}_d \subseteq \mathcal{S}$. From Lemma 1 we know that $d(\mathcal{S}') > d(\mathcal{S})/3$.

We will show that $\mathcal{S}'$ serves as a diffusion core for $\mathcal{S}_d$. Thereto fix any $v \in \mathcal{S}'$ and let κ denote the hitting time $\kappa = \tau_v(\mathcal{S}_d^c)$. Then we define t such that $\mathbb{P}(\kappa \leq t) > 1/10$.

Now let Y be the event that $\kappa \leq t$ and $\tau_u(\mathcal{S}^c) \leq (40\phi)^{-1}$, with $u = X_\kappa$ a random variable corresponding to the node at which the RW hits $\mathcal{S}_d^c$. Then we have that $Y \Rightarrow (\tau_v(\mathcal{S}^c) \leq t + (40\phi)^{-1})$. Since $u \notin \mathcal{S}_d$, it holds that $\mathbb{P}(\tau_u(\mathcal{S}^c) \leq (40\phi)^{-1}) > 1/4$. Combined with the assumption that $\mathbb{P}(\kappa \leq t) > 1/10$, this allows to bound $\mathbb{P}(Y) > (1/4)(1/10) = 1/40$, and therefore $\mathbb{P}(\tau_v(\mathcal{S}^c) \leq t + (2\phi)^{-1}) > 1/40$. However, since $v \in \mathcal{S}'$ we also have that $\mathbb{P}(\tau_v(\mathcal{S}^c) > 1/(30\phi)) \geq 39/40$, or equivalently $\mathbb{P}(\tau_v(\mathcal{S}^c) \leq 1/(30\phi)) \leq 1/40$. This gives a contradiction if $t + (40\phi)^{-1} \leq 1/(30\phi)$, or equivalently $t \leq 1/(120\phi)$. For such t the initial hypothesis $\mathbb{P}(\kappa \leq t) > 1/10$ must hence be false, and therefore it must hold that $\mathbb{P}(\kappa \leq (120\phi)^{-1}) \leq 1/10$ for all $v \in \mathcal{S}'$. This proves the claimed statement. □

We will also use the following lemma, which in essence was already present in [CS10, NS10, KS11]. It argues that a graph which is ϵ-far from having a certain expansion must have a large subset with low expansion.

Lemma 3. *Let G be an undirected n-node graph with degree bound d that is ϵ-far from having expansion $\geq \beta$, with $\beta \leq 1/10$. Then the following holds:*

- *There exists a subset $\mathcal{A} \subset \mathcal{V}$, with $\epsilon n/12 \leq |\mathcal{A}| \leq (1+\epsilon)n/2$, such that $\Phi(\mathcal{A}) < r_d\,\beta$, with r_d a constant dependent on d.*
- *For any $t \leq 1/(2r_d\beta)$ and distribution v having a γ-overlap with the diffusion core of $\mathcal{A}$, with $\gamma > 2(1+\epsilon)/3$, it holds that*

$$\|P^t v\|^2 \geq \frac{1}{n}\left(1 + 4\left(\frac{3\gamma}{4} - \frac{1+\epsilon}{2}\right)^2\right).$$

Proof. The first bullet is proven in [CS10, Corollary 4.6]. To prove the second bullet, we use the fact that for a general probability distribution w it holds that

$$\|w\|^2 \geq \frac{1}{n}\left(1 + \|w - u\|_1^2\right),$$

with u the uniform distribution. This bound can be found in the proof of [CS10, Lemma 4.3]. To lower bound the right hand side, we will use that

$$\|w - u\|_1 = 2\max_{\mathcal{S}\subseteq\mathcal{V}} |w(\mathcal{S}) - u(\mathcal{S})| \geq 2|w(\mathcal{A}) - u(\mathcal{A})|.$$

If $w = P^t v$, we can lower bound $(P^t v)(\mathcal{A})$ since this represents the probability that a t-step RW from v end in $\mathcal{A}$. By definition of the diffusion core, we know that a $(t \leq 1/(40\Phi(\mathcal{A})))$-step RW, starting anywhere in the diffusion core $\mathcal{A}_d$ of $\mathcal{A}$, remains inside $\mathcal{A}$ with probability at least $3/4$. Since v has a γ-overlap with $\mathcal{A}_d$, this proves that a $(t \leq 1/(40r_d\beta))$-step RW from v remains inside $\mathcal{A}$ with probability at least $3\gamma/4$. This implies that $(P^t v)(\mathcal{A}) \geq 3\gamma/4$ and hence $\|P^t v - u\|_1 \geq 2|3\gamma/4 - u(\mathcal{A})|$. By our assumption that $\gamma > 2(1+\epsilon)/3$, and since $u(\mathcal{A}) = |\mathcal{A}|/n \leq (1+\epsilon)/2$, this implies that $\|P^t v - u\|_1 \geq 2(3\gamma/4 - (1+\epsilon)/2)$. Combining this with the above inequality proves the claimed bound. □

2.5 Quantum Walks

Quantum walks (QWs) form an elegant quantum counterpart to random walks on graphs. They similarly explore a graph in a local manner, by performing queries in superposition to the neighbors of certain nodes, as illustrated in (2) in Section 2.2. In the following, let P be a symmetric random walk transition matrix (as will be the case for us), and let $\mathcal{S} \subseteq \mathcal{V}$

be the initial seed set. We denote by $|\mathcal{S}\rangle = |\mathcal{S}|^{-1/2} \sum_{v\in\mathcal{S}} |v\rangle$ the quantum state that is a uniform superposition over nodes in $\mathcal{S}$. Starting from $|S\rangle$, QWs allow to create a quantum state or "quantum sample" of the form

$$|\psi_t\rangle = P^t|\mathcal{S}\rangle + |\Gamma\rangle.$$

Here the first component forms a quantum encoding of the RW probability distribution started from a uniformly random node in $\mathcal{S}$. The second component denotes some auxiliary garbage state in which we will not be interested. In our earlier work on quantum expansion testing, we introduced a QW technique called "quantum fast-forwarding" (QFF) that allows to approximate the above quantum sample in the square root of the classical runtime. The following lemmas follow from [AS19], recalling that d_M denotes the maximum degree of the graph.

Lemma 4 (QFF)**.** *Starting from the state $|\mathcal{S}\rangle$, there exists a QW algorithm that outputs a state ϵ-close to the quantum sample $|\psi_t\rangle$ in complexity $\widetilde{O}(t^{1/2}d_M^{1/2}\log^{1/2}(1/\epsilon))$.*

Proof. We first note that, starting from $|\mathcal{S}\rangle$, the state $|\psi_t\rangle$ can be ϵ-approximated using a number of *QW steps* that scales as $\widetilde{O}(t^{1/2}\log^{1/2}(1/\epsilon))$ (hiding polylog-dependencies on t, n, ϵ). An explicit statement of this fact can be found as Theorem 7 in [AGJK20]. Second we note that a single QW step can be implemented in complexity $O(d_M^{1/2})$, as is mentioned in [AS19, A19]. Combining both facts proves the lemma. □

For clarity of exposition, we will ignore the approximation error ϵ of QFF in the rest of the paper. By linearity it suffices to set ϵ inverse polynomially small in n, and so the approximation error will only add a log-factor to the overall complexity (this in contrast to the approximation error in the lemma below, which in fact is important).

Given access to such quantum samples, we can use a standard quantum routine called "quantum amplitude estimation" to estimate $\|P^t|\mathcal{S}\rangle\|$. This leads to the following lemma, which is an immediate corollary from [AS19, Theorem 5] and Lemma 4.

Lemma 5 (2-norm estimator)**.** *There exists a QW algorithm that, with probability at least $1-\delta$, outputs an estimate a such that $\big|\|P^t|v\rangle\| - a\big| \le \epsilon$. The algorithm has complexity $\widetilde{O}(t^{1/2}d_M^{1/2}\epsilon^{-1}\log\delta^{-1})$ and uses $\widetilde{O}(\epsilon^{-1}\log\delta^{-1})$ reflections around $|\mathcal{S}\rangle$.*

As discussed in Section 2.1 (and equal to the approach in [A19]), we can use a QRAM data structure to prepare and reflect around the quantum state $|\mathcal{S}\rangle$. The effective complexity of these operations in our computation model is then $\widetilde{O}(1)$ (i.e., polylogarithmic in n).

3 Evolving Set Processes

Evolving Set Processes (ESPs) have been used for analyzing the mixing time of Markov chains [MP05], and as an algorithmic tool for performing local graph clustering [AP09, AGPT16]. Derived from some original Markov chain over a node set $\mathcal{V}$, an ESP is a Markov chain over *subsets* of the node set. For our particular case we will assume that the original Markov chain corresponds to the (lazy) RW. Given that the current state of the ESP is $\mathcal{S}\subseteq\mathcal{V}$, its next state is then determined by the following rule: draw a variable U uniformly at random from the interval $[0,1]$, and set the next state

$$\mathcal{S}' = \{v\in\mathcal{V} \,:\, P(\mathcal{S},v)\ge U\}.$$

Here $P(\mathcal{S}, v)$ denotes the probability that a single RW step from v ends up in $\mathcal{S}$, and is given by $P(v, \mathcal{S}) = |\mathcal{E}(v, \mathcal{S})|/(2d(v)) + \mathbb{I}(v \in \mathcal{S})/2$. This gives rise to an ESP transition matrix $K : 2^{\mathcal{V}} \times 2^{\mathcal{V}} \to [0, 1]$. Notice that only states in the inner or outer boundary of $\mathcal{S}$ can be added or removed: $|\mathcal{E}(v, \mathcal{S})| = d(v)$ and $\mathbb{I}(v \in \mathcal{S}) = 1$ if y and all of its neighbors lie in $\mathcal{S}$, whereas $|\mathcal{E}(v, \mathcal{S})| = \mathbb{I}(v \in \mathcal{S}) = 0$ if v nor any of its neighbors lie in $\mathcal{S}$. This process has absorbing states $\mathcal{S} = \emptyset$ and $\mathcal{S} = \mathcal{V}$, both of which have no boundary. For algorithmic purposes, it is desirable to prevent the ESP from being absorbed in the empty set. To this end, the transition probabilities can be slightly altered:

$$\hat{K}(\mathcal{S}, \mathcal{S}') = \frac{d(\mathcal{S}')}{d(\mathcal{S})} K(\mathcal{S}, \mathcal{S}').$$

Clearly the transition probability to the empty set is now equal to zero. $\hat{K}$ is again a stochastic transition matrix, and the resulting process is called the *volume-biased* ESP (yet for brevity we will simply refer to it as the ESP). We refer the reader to [AGPT16, LPW17] for more details on the ESP and its volume-biased variant.

Starting inside some low-expansion set $\mathcal{S}$, ESPs are used as a means of locally constructing or exploring $\mathcal{S}$. In our case, we only wish to retrieve a smaller subset, typically of size $|\mathcal{S}|^{1/3}$. This subset however should be sufficiently localized "inside" $\mathcal{S}$, i.e., have a sufficient overlap with the smaller diffusion core of $\mathcal{S}$. To this end we refine the ESP analysis: we use our Lemma 2 to show that also the ESP will remain in the diffusion core with large probability. The following Section 3.1 introduces some useful properties of the ESP, and in Section 3.2 we prove the main tool.

3.1 ESP Complexity and Properties

As we wish to use an ESP as an algorithmic means, it is desirable to quantify the resources needed to simulate it. Thereto we define the *cost* of a sample path

$$\mathrm{cost}(\mathcal{S}_0, \ldots, \mathcal{S}_t) = d(\mathcal{S}_0) + \sum_{i=1}^{t} \left(d(\mathcal{S}_i \triangle \mathcal{S}_{i-1}) + |\partial(\mathcal{S}_{i-1})| \right),$$

with $d(\mathcal{S})$ the total degree of a subset $\mathcal{S}$ and $\mathcal{S} \triangle \mathcal{S}'$ the symmetric difference between $\mathcal{S}$ and $\mathcal{S}'$. We also define a stopping time $\tau(T, B, \theta)$ for the ESP:

Definition 3. *The stopping time $\tau(T, B, \theta)$ is a random variable that equals the first time τ at which either $\phi(\mathcal{S}_\tau) \leq \theta$, $\tau = T$, or $\mathrm{cost}_\tau > B$.*

The following theorem from [AGPT16] bounds the complexity of sampling from the ESP with stopping rule $\tau(T, B, \theta)$.

Theorem 3 ([AGPT16, Proposition 5.3])**.** *There exists an algorithm that takes as input a node v, two integers $T, B \geq 0$ and $\theta \in [0, 1]$. Let $\mathcal{S}_0 = \{v\}$ and define the stopping time $\tau = \tau(T, B, \theta)$. The algorithm generates a sample path $(\mathcal{S}_0, \ldots, \mathcal{S}_\tau)$ of the ESP and outputs the last set $\mathcal{S}_\tau$. The complexity of the algorithm is $O(B \log m)$.*

3.2 ESP for Growing Seed Set

Using known results on ESPs, combined with our new Lemma 2, we can derive the following theorem. This constitutes the main tool that we will use to grow seed sets. We defer the proof technicalities to Appendix A.

Theorem 4. *Fix a parameter $M \geq 0$. Let $\mathcal{S} \subseteq \mathcal{V}$ be such that $\phi(\mathcal{S}) \leq \gamma^2/(2400 \log m)$. Let $\mathcal{S}' \subseteq \mathcal{S}$ be as defined in Lemma 2, and assume that $\mathcal{S}_0 = \{v\}$ for some $v \in \mathcal{S}'$. Let $\mathcal{S}_\tau$ be the set returned by the ESP with stopping rule $\tau(T, B, \theta)$ and parameters $T = 20\theta^{-2} \log m$, $B = 25M\sqrt{T \log m}$, and $\theta = \gamma$. Then with probability at least $1/5$ we have that $d(\mathcal{S}_\tau \cap \mathcal{S}_d)/d(\mathcal{S}_\tau) \geq 3/4$, and either $\phi(\mathcal{S}_\tau) \leq \gamma$ or $d(\mathcal{S}_\tau) \geq M$. The complexity of generating this set is $O(M\gamma^{-1} \log^2 m)$.*

4 Quantum Expansion Tester

We are now ready to construct our new quantum expansion tester, yielding the main contribution of this paper.

Algorithm 2 Quantum Expansion Tester

Input: parameters n and d; query access to an n-node graph G with degree bound d; expansion parameter Φ; promise parameter ϵ

Do:

1: set parameters:
$t = 16d^2\Phi^{-2} \log n$, $\delta = \epsilon/1000$, $K = 200/(\epsilon(1-\delta))$,
$\theta = \Phi/(2d)$, $B = 800\sqrt{5}\lceil n^{1/3}d \rceil d\Phi^{-1} \log m$, $T = 320\Phi^{-2}d^2 \log m$
2: **do** K **times**
3: select a uniformly random starting node v
4: run ESP from $\mathcal{S}_0 = \{v\}$ with stopping rule $\tau(T, B, \theta)$, outputting $\mathcal{S}$
5: **if** $|\mathcal{S}| \leq n/2$ and $\Phi(\mathcal{S}) \leq \Phi/2$ **then** abort and output "`reject`"
6: use 2-norm estimator to create estimate a of $\|P^t|\mathcal{S}\rangle\|$
to precision $\epsilon' = \sqrt{|\mathcal{S}|/n}(1 - \sqrt{1 + 1/256})/4$ with probability $1 - \delta$
7: **if** $a > \sqrt{|\mathcal{S}|n^{-1}(1 + n^{-1})} + \epsilon'$ **then** abort and output "`reject`"

Output: if no "`reject`", output "`accept`"

Theorem 5 (Quantum Expansion Tester)**.** *If $d \geq 3$ and $\epsilon < 1/16$, then Algorithm 2 is a (Φ, ϵ) expansion tester for $c = 1/(2400(2d)^2 r_d)$, with r_d as in Lemma 3. The complexity of the algorithm is bounded by $\widetilde{O}(n^{1/3}\Phi^{-1}d^{3/2}\epsilon^{-1})$.*

Proof. First we prove that if $\Phi(G) \geq \Phi$, then the algorithm accepts with probability at least $2/3$. Thereto note that by definition of the vertex expansion, necessarily $\Phi(\mathcal{S}) \geq \Phi(G) \geq \Phi$ if $|\mathcal{S}| \leq n/2$. Hence the algorithm cannot falsely reject in step 5. To exclude rejection in step 7, we use the result in [NS10, Proof of Theorem 2.1] showing that if $\Phi(G) \geq \Phi$ then for all nodes $v \in \mathcal{V}$ and time $t \geq 16d^2\Phi^{-2} \log n$ it holds that $\|P^t|v\rangle\| \leq \sqrt{n^{-1}(1 + n^{-1})}$. This allows to bound $\|P^t|\mathcal{S}\rangle\| \leq |\mathcal{S}|^{-1/2} \sum_{s \in \mathcal{S}} \|P^t|s\rangle\| \leq \sqrt{|\mathcal{S}|n^{-1}(1 + n^{-1})}$. Using the 2-norm estimator from Lemma 5, the estimate a will with probability $1 - \delta$ be such that $a \leq \sqrt{|\mathcal{S}|n^{-1}(1 + n^{-1})} + \epsilon'$. Step 7 will therefore reject falsely only with probability at most δ. The total probability of a faulty rejection can then be bounded by $K\delta < 1/3$. Since the algorithm accepts if it never rejects, it will correctly accept the graph with probability at least $2/3$.

Next we prove that if G is ϵ-far from having expansion $\geq c\Phi^2 \log^{-1}(dn)$, then the algorithm rejects with probability at least $2/3$. Thereto we use Lemma 3 from Section 2.4.1, which states that in this case there exist a "bad" subset $\mathcal{A}$, with $(1 + \epsilon)n/2 \geq |\mathcal{A}| \geq \epsilon n/12$, such that

$$\Phi(\mathcal{A}) < r_d c\Phi^2 \log^{-1}(dn) = \frac{1}{2400}\left(\frac{\Phi}{2d}\right)^2 \frac{1}{\log(dn)}. \tag{4}$$

From $\mathcal{A}$, we can define the diffusion core $\mathcal{A}_d$ and the subset $\mathcal{A}' \subseteq \mathcal{A}_d$ as in Lemma 2, which states that $d(\mathcal{A}') > d(\mathcal{A})/3$ and hence $|\mathcal{A}'| \geq |\mathcal{A}|/3$. The initial node v will hence be in $\mathcal{A}'$ with probability at least $|\mathcal{A}|/(3n) \geq \epsilon/36$.

Conditioning on $v \in \mathcal{A}'$, we can analyze the ESP output set $\mathcal{S}$ using Theorem 4. We choose $M = \lceil n^{1/3}d \rceil$. Using the bound (4), which by (3) implies the same upper bound for $\phi(\mathcal{A})$, $\mathcal{S}$ will with probability at least $1/5$ be such that (i) $d(\mathcal{S} \cap \mathcal{A}_d) \geq 3d(\mathcal{S})/4$, and therefore $|\mathcal{S} \cap \mathcal{A}_d| \geq 3|\mathcal{S}|/4$, and (ii) either $\phi(\mathcal{S}) \leq \Phi/(2d)$ or $d(\mathcal{S}) \geq M$. If $\phi(\mathcal{S}) \leq \Phi/(2d)$ and $|\mathcal{S}| \leq n/2$, then we have a proof that G has vertex expansion $\Phi(G) \leq \Phi/2$, and hence we reject the graph in step 5. To see this, simply note that $\phi(\mathcal{S}) \leq \Phi/(2d)$ implies that $\Phi(\mathcal{S}) \leq \Phi/2$ (again by (3)). In any other case, we know that $d(\mathcal{S}) \geq M$ and hence $|\mathcal{S}| \geq M/d$. Given such a set $\mathcal{S}$, consider the uniform distribution $\pi_{\mathcal{S}}$ over $\mathcal{S}$. Since $|\mathcal{S} \cap \mathcal{A}_d| \geq 3|\mathcal{S}|/4$, we know that $\pi_{\mathcal{S}}$ has a $3/4$-overlap with $\mathcal{A}_d$. By the second bullet of Lemma 3 this implies that for all $t \leq \log(dn)/(2r_d c\Phi^2) = 1200(2d)^2 \log(dn)\Phi^{-2}$,

$$\|P^t \pi_{\mathcal{S}}\| \geq \sqrt{\frac{1}{n}\left(1 + 4\left(\frac{3\gamma}{4} - \frac{1+\epsilon}{2}\right)^2\right)} \geq \sqrt{\frac{1}{n}\left(1 + \frac{1}{256}\right)}.$$

using that $\gamma = 3/4$ and $\epsilon \leq 1/16$. By Lemma 5, the estimate a will then with probability $1-\delta$ be such that $a \geq \sqrt{|\mathcal{S}|n^{-1}}\sqrt{1+1/256} - \epsilon'$. If $\epsilon' \leq \sqrt{|\mathcal{S}|n^{-1}}(\sqrt{1+1/256}-1)/4$, then this is strictly larger than $\sqrt{|\mathcal{S}|n^{-1}(1+n^{-1})} + \epsilon'$ for sufficiently large n, allowing to correctly reject the graph with probability at least $1-\delta$.

Now we can bound the total probability of correctly rejecting the graph in a single iteration. Thereto we multiply the probability that $v \in \mathcal{A}'$ ($\geq \epsilon/36$), the ESP process succeeds ($\geq 1/5$) and the 2-norm estimator succeeds ($\geq 1-\delta$), yielding a total rejection probability of at least $p = \epsilon(1-\delta)/180$. The total probability of correctly rejecting at least once in K iterations is therefore at least $1-(1-p)^K$. Using the elementary inequality $(1-p)^{1/p} < 1/e$ for any $0 < p \leq 1$, we can lower bound the rejection probability as

$$1 - (1-p)^K > 1 - \left(\frac{1}{e}\right)^{Kp} \geq \frac{2}{3},$$

provided that $K \geq \ln(3)/p = 180\ln(3)/(\epsilon(1-\delta))$, which is ensured by our choice of K. This concludes the proof that Algorithm 2 is a (Φ, ϵ)-expansion tester.

Towards bounding the complexity of the algorithm, we consider a single iteration of the for-loop. By Theorem 4, the complexity of simulating the ESP in step 4 is $O(M\theta^{-1}\log^2(dn))$, which is $O(n^{1/3}d^2\Phi^{-1}\log^2(dn))$. By Lemma 5, the (ϵ', δ) 2-norm estimator in step 6 has complexity

$$\widetilde{O}(t^{1/2}d^{1/2}\epsilon'^{-1}\log\delta^{-1}) \in \widetilde{O}(n^{1/3}d^{3/2}\Phi^{-1}\log\epsilon^{-1}).$$

Since we iterate the for-loop $K \in O(\epsilon^{-1})$ times, this gives the claimed complexity. □

Acknowledgements

This work greatly benefited from discussions and comments by Alain Sarlette, Anthony Leverrier, Ronald de Wolf and André Chailloux, as well as from comments and suggestions by multiple anonymous referees. Most of the work was done while part of the CWI-Inria International Lab. We acknowledge support from the QuantERA ERA-NET Cofund in Quantum Technologies implemented within the European Union's Horizon 2020 Programme (QuantAlgo project) and from the Belgian Fonds de la Recherche Scientifique - FNRS under grant no R.50.05.18.F (QuantAlgo).

References

[ACL06] Reid Andersen, Fan R.K. Chung, and Kevin Lang. Local graph partitioning using PageRank vectors. In *Proceedings of the 47th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 475–486. IEEE, 2006. doi: 10.1109/FOCS.2006.44

[ACL11] Andris Ambainis, Andrew M. Childs, and Yi-Kai Liu. Quantum property testing for bounded-degree graphs. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pages 365–376. Springer, 2011. doi: 10.1007/978-3-642-22935-0_31

[AGJK20] Andris Ambainis, András Gilyén, Stacey Jeffery, and Martins Kokainis. Quadratic speedup for finding marked vertices by quantum walks. In *Proceedings of the 52nd ACM Symposium on Theory of Computing (STOC)*, pages 412–424. ACM, 2020. doi: 10.1145/3357713.3384252

[AGPT16] Reid Andersen, Shayan Oveis Gharan, Yuval Peres, and Luca Trevisan. Almost optimal local graph clustering using evolving sets. *Journal of the ACM*, 63(2):15, 2016. doi: 10.1145/2856030

[Amb07] Andris Ambainis. Quantum walk algorithm for element distinctness. *SIAM Journal on Computing*, 37(1):210–239, 2007. doi: 10.1137/S0097539705447311

[AP09] Reid Andersen and Yuval Peres. Finding sparse cuts locally using evolving sets. In *Proceedings of the 41st ACM Symposium on Theory of Computing (STOC)*, pages 235–244. ACM, 2009. doi: 10.1145/1536414.1536449

[A19] Simon Apers. Quantum walk sampling by growing seed sets. In *Proceedings of the 27th European Symposium on Algorithms (ESA)*, volume 144 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 9:1–9:12. Springer, 2019. doi: 10.4230/LIPIcs.ESA.2019.9

[AS19] Simon Apers and Alain Sarlette. Quantum fast-forwarding Markov chains and property testing. *Quantum Information and Computation*, 19(3&4):181–213, 2019. doi: 10.5555/3370245.3370246.

[CKK+18] Ashish Chiplunkar, Michael Kapralov, Sanjeev Khanna, Aida Mousavifar, and Yuval Peres. Testing graph clusterability: Algorithms and lower bounds. In *Proceedings of the 59th Annual IEEE Symposium on Foundations of Computer Science*. IEEE, 2018. doi: 10.1109/FOCS.2018.00054

[CPS15] Artur Czumaj, Pan Peng, and Christian Sohler. Testing cluster structure of graphs. In *Proceedings of the 47th ACM Symposium on Theory of Computing (STOC)*, pages 723–732. ACM, 2015. doi: 10.1145/2746539.2746618

[CS10] Artur Czumaj and Christian Sohler. Testing expansion in bounded-degree graphs. *Combinatorics, Probability and Computing*, 19(5-6):693–709, 2010. doi: 10.1017/S096354831000012X

[GR02] Oded Goldreich and Dana Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2):302–343, 2002. doi: 10.1007/s00453-001-0078-7

[GR11] Oded Goldreich and Dana Ron. On testing expansion in bounded-degree graphs. In *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*, pages 68–75. Springer, 2011. doi: 10.1007/978-3-642-22670-0_9

[HLW06] Shlomo Hoory, Nathan Linial, and Avi Wigderson. Expander graphs and their applications. *Bulletin of the American Mathematical Society*, 43(4):439–561, 2006. doi: 10.1090/S0273-0979-06-01126-8

[KP17] Iordanis Kerenidis and Anupam Prakash. Quantum recommendation systems. In *Proceedings of the 8th Innovations in Theoretical Computer Science Conference (ITCS)*, pages 49:1–49:21. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017. doi: 10.4230/LIPIcs.ITCS.2017.49

[KS11] Satyen Kale and Comandur Seshadhri. An expansion tester for bounded degree graphs. *SIAM Journal on Computing*, 40(3):709–720, 2011. doi: 10.1137/100802980

[LPW17] David A Levin, Yuval Peres, and Elizabeth L Wilmer. *Markov chains and mixing times*. American Mathematical Society, 2017. doi: 10.1090/mbk/058

[LR99] Tom Leighton and Satish Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM*, 46(6):787–832, 1999. doi: 10.1145/331524.331526

[LRV13] Anand Louis, Prasad Raghavendra, and Santosh Vempala. The complexity of approximating vertex expansion. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pages 360–369. IEEE, 2013. doi: 10.1109/FOCS.2013.46

[MdW16] Ashley Montanaro and Ronald de Wolf. A survey of quantum property testing. *Theory of Computing Library Graduate Surveys*, (7):1–81, 2016. doi: 10.4086/toc.gs.2016.007

[MP05] Ben Morris and Yuval Peres. Evolving sets, mixing and heat kernel bounds. *Probability Theory and Related Fields*, 133(2):245–266, 2005. doi: 10.1007/s00440-005-0434-7

[NS10] Asaf Nachmias and Asaf Shapira. Testing the expansion of a graph. *Information and Computation*, 208(4):309, 2010. doi: 10.1016/j.ic.2009.09.002

[OSV12] Lorenzo Orecchia, Sushant Sachdeva, and Nisheeth K. Vishnoi. Approximating the exponential, the Lanczos method and an $\widetilde{O}(m)$-time spectral algorithm for balanced separator. In *Proceedings of the 44th ACM Symposium on Theory of Computing (STOC)*, pages 1141–1160. ACM, 2012. doi: 10.1145/2213977.2214080

[ST13] Daniel A. Spielman and Shang-Hua Teng. A local clustering algorithm for massive graphs and its application to nearly linear time graph partitioning. *SIAM Journal on Computing*, 42(1):1–26, 2013. doi: 10.1137/080744888

[Wat01] John Watrous. Quantum simulations of classical random walks and undirected graph connectivity. *Journal of Computer and System Sciences*, 62(2):376–391, 2001. doi: 10.1006/jcss.2000.1732

A Proof of ESP Algorithm

In this appendix we provide the proof of Theorem 4. We make use of several known properties that can be attributed to the output set:

- *Size*: The cost of simulating the process is related to the size of the output set.

Lemma 6 ([AGPT16, Theorem 5.4]). *For any starting set $\mathcal{S}_0$ and any stopping time τ that is upper bounded by T, it holds that*

$$\mathbb{E}[\mathrm{cost}_\tau / d(\mathcal{S}_\tau)] \leq 1 + 4\sqrt{T \log m}.$$

- *Overlap*: If a random walk has a high probability of staying inside a certain set, then with high probability the ESP will also largely remain inside that set.

  Lemma 7 ([AGPT16, Lemma 4.3]). *Consider any set $\mathcal{S} \subseteq \mathcal{V}$, a starting set $\mathcal{S}_0 = \{v\}$ for some $v \in \mathcal{S}$, and an integer $T \geq 0$. Then the following holds for all $\beta > 0$:*

$$\mathbb{P}(\min_{t \leq T} d(\mathcal{S}_t \cap \mathcal{S})/d(\mathcal{S}_t) \geq 1 - \beta\, \mathbb{P}(\tau_v(\mathcal{S}^c) \leq T)) \geq 1 - 1/\beta.$$

- *Conductance*: After T steps, the ESP encounters with high probability a set of conductance $\widetilde{O}(T^{-1/2})$.

  Lemma 8 ([AP09, Corollary 1]). *Fix any integer T, and let $\theta_T = \sqrt{4T^{-1} \log m}$. For any starting set $\mathcal{S}_0$ and constant $c \geq 0$, it holds that*

$$\mathbb{P}(\min_{t < T} \phi(\mathcal{S}_t) \leq \sqrt{c}\, \theta_T) \geq 1 - 1/c.$$

Using these lemmas, combined with our Lemma 2, we can prove the theorem below. It corresponds to Theorem 4 when setting the parameters $\alpha = 5$ and $\beta = 5/2$.

Theorem 6. *Fix constants $\alpha, \beta > 0$ and a parameter $M \geq 0$. Let $\mathcal{S} \subseteq \mathcal{V}$ be such that*

$$\phi(\mathcal{S}) \leq \frac{1}{480\alpha} \frac{\gamma^2}{\log m}.$$

Let $\mathcal{S}' \subseteq \mathcal{S}$ be as defined in Lemma 2, and assume that $\mathcal{S}_0 = \{v\}$ for some $v \in \mathcal{S}'$. Let $\mathcal{S}_\tau$ be the set returned by the ESP with stopping rule $\tau(T, B, \theta)$ and parameters $T = 4\alpha\theta^{-2} \log m$, $B = 5\alpha M \sqrt{T \log m}$, and $\theta = \gamma$. Then with probability at least $1 - 2\alpha^{-1} - \beta^{-1}$ we have that

$$\frac{d(\mathcal{S}_\tau \cap \mathcal{S}_d)}{d(\mathcal{S}_\tau)} \geq 1 - \beta/10,$$

and either

$$\phi(\mathcal{S}_\tau) \leq \gamma \quad \text{or} \quad d(\mathcal{S}_\tau) \geq M.$$

The complexity of generating this set is $O(M\gamma^{-1} \log^2 m)$.

Proof. Let X denote the event that $d(\mathcal{S}_\tau \cup \mathcal{S}_d)/d(\mathcal{S}_\tau) \geq 1 - \beta/10$, and Y the event that $\phi(\mathcal{S}_\tau) \leq \gamma$ or $d(\mathcal{S}_\tau) \geq M$. We will show that $\mathbb{P}(X) \geq 1 - \beta^{-1}$ and $\mathbb{P}(Y) \geq 1 - 2\alpha^{-1}$, so that by the union bound we find that $\mathbb{P}(X \cap Y) \geq \mathbb{P}(X) + \mathbb{P}(Y) - 1 \geq 1 - 2\alpha^{-1} - \beta^{-1}$. This proves the statements on the output set. The complexity statement follows from Theorem 3.

Towards bounding $\mathbb{P}(X)$ we combine Lemma 7 with Lemma 2. Since $v \in \mathcal{S}'$, we know that

$$\mathbb{P}(\tau_v(\mathcal{S}_d^c) \leq (120\phi(\mathcal{S}))^{-1}) \leq 1/10.$$

Our assumption on $\phi(\mathcal{S})$ implies that $T \leq (120\phi(\mathcal{S}))^{-1}$, and so by Lemma 7 we get that

$$\mathbb{P}(\min_{t \leq T} d(\mathcal{S}_t \cap \mathcal{S}_d)/d(\mathcal{S}_t) \geq 1 - \beta/10) \geq 1 - \beta^{-1}.$$

Since $\tau \leq T$, the left-hand side lower bounds $\mathbb{P}(X)$, so that $\mathbb{P}(X) \geq 1 - \beta^{-1}$.

Towards bounding $\mathbb{P}(Y)$, we condition it on the possible stopping rule outcomes:

$$\begin{aligned}\mathbb{P}(Y) = \mathbb{P}(Y|\phi(\mathcal{S}_\tau) \leq \theta)\,\mathbb{P}(\phi(\mathcal{S}_\tau) \leq \theta) + \mathbb{P}(Y|\,\mathrm{cost}_\tau > B)\,\mathbb{P}(\mathrm{cost}_\tau > B) \\ + \mathbb{P}(Y|\tau = T)\,\mathbb{P}(\tau = T).\end{aligned}$$

Since $\theta = \gamma$ we know that $\mathbb{P}(Y|\phi(\mathcal{S}_\tau) \leq \theta) = 1$. Next we wish to bound the second term by using Lemma 6. In combination with Markov's inequality this states that $\mathbb{P}(Z_\alpha) \leq 1/\alpha$, where we let Z_α denote the event that

$$\mathrm{cost}_\tau\,/d(\mathcal{S}_\tau) \geq \alpha(1 + 4\sqrt{T \log m}) = \alpha + 4B/(5M).$$

If we lower bound $\mathbb{P}(Z_\alpha) \geq \mathbb{P}(Z_\alpha|\,\mathrm{cost}_\tau > B)\,\mathbb{P}(\mathrm{cost}_\tau > B)$, then we find $\mathbb{P}(Z_\alpha|\,\mathrm{cost}_\tau > B)\,\mathbb{P}(\mathrm{cost}_\tau > B) \leq \alpha^{-1}$. With $\bar{Z}_\alpha$ the negation of Z_α, this leads to the bound

$$\mathbb{P}(\bar{Z}_\alpha|\,\mathrm{cost}_\tau > B)\,\mathbb{P}(\mathrm{cost}_\tau > B) \geq \mathbb{P}(\mathrm{cost}_\tau > B) - \alpha^{-1}.$$

Now if both $\mathrm{cost}_\tau > B$ and $\bar{Z}_\alpha$ hold, then

$$d(\mathcal{S}_\tau) > \frac{\mathrm{cost}_\tau}{\alpha + 4B/(5M)} > \frac{B}{\alpha + 4B/(5M)} > M,$$

using the bound $B > 5\alpha M$. As a consequence, if both $\mathrm{cost}_\tau > B$ and Z_α hold, then also Y holds, and so $\mathbb{P}(Y|\,\mathrm{cost}_\tau > B) \geq \mathbb{P}(Z_\alpha\ \mathrm{cost}_\tau > B)$. This gives the desired bound

$$\mathbb{P}(\mathrm{cost}_\tau > B)\,\mathbb{P}(Y|\,\mathrm{cost}_\tau > B) \geq \mathbb{P}(\mathrm{cost}_\tau > B) - \alpha^{-1}. \tag{5}$$

Finally we will upper bound $\mathbb{P}(\tau = T)$ using Lemma 8. For $c = \alpha$ this states that

$$\mathbb{P}\big(\min_{t<T}\,\phi(\mathcal{S}_t) \leq \gamma\big) \geq 1 - \alpha^{-1}.$$

Since $\min_{t<T}\,\phi(\mathcal{S}_t) \leq \gamma$ implies that $\tau < T$, this shows that

$$\mathbb{P}(\tau = T) \leq 1 - \mathbb{P}\big(\min_{t<T}\,\phi(\mathcal{S}_t) \leq \gamma\big) \leq \alpha^{-1}. \tag{6}$$

If now we combine the bounds (5) and (6) we get the final bound on $\mathbb{P}(Y)$:

$$\mathbb{P}(Y) \overset{(5)}{\geq} \mathbb{P}(\phi(\mathcal{S}_\tau) \leq \theta) + \mathbb{P}(\mathrm{cost}_\tau > B) - \alpha^{-1} \geq 1 - \mathbb{P}(\tau = T) - \alpha^{-1} \overset{(6)}{\geq} 1 - 2\alpha^{-1}. \qquad \square$$