Planar Graphs: Random Walks and Bipartiteness Testing*

Artur Czumaj† Morteza Monemizadeh‡ Krzysztof Onak§ Christian Sohler¶

July 8, 2014

Abstract

We initiate the study of property testing in *arbitrary planar graphs*. We prove that *bipartiteness* can be tested in constant time: for every planar graph G and $\varepsilon > 0$, we can distinguish in constant time between the case that G is bipartite and the case that G is ε-far from bipartite. The previous bound for this class of graphs was $\tilde{O}(\sqrt{n})$, where n is the number of vertices, and the constant-time testability was only known for planar graphs with *bounded degree*. Our approach extends to arbitrary minor-free graphs.

Our algorithm is based on random walks. The challenge here is to analyze random walks for graphs that have good separators, i.e., bad expansion. Standard techniques that use a fast convergence of random walks to a uniform distribution do not work in this case. Informally, our approach is to self-reduce the problem of finding an odd-length cycle in a multigraph G induced by a collection of cycles to the same problem on another multigraph G' induced by a set of shorter odd-length cycles, in such a way that when a random walk finds a cycle in G' with probability $p > 0$, then it does so in G with probability $\lambda(p) > 0$. This reduction is applied until the cycles collapse to self-loops, in which case they can be easily detected.

While the analysis presented in this paper applies only to testing bipartiteness, we believe that the techniques developed will find applications to testing other properties in arbitrary planar (or minor-free) graphs, in a similar way as in the past the advances in testing bipartiteness led to the development of testing algorithms for more complex graph properties.

*A preliminary version of this paper appeared in *Proceedings of the 52th IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 423–432, Palm Springs, CA, October 22–25, 2011. IEEE Computer Society Press, Los Alamitos, CA, 2011.

†Department of Computer Science, Centre for Discrete Mathematics and its Applications, University of Warwick. Email: A.Czumaj@warwick.ac.uk. Research supported in part by the *Centre for Discrete Mathematics and its Applications (DIMAP)*, EPSRC award EP/D063191/1, by EPSRC award EP/G064679/1, and by a Weizmann-UK Making Connections Grant "The Interplay between Algorithms and Randomness."

‡Department of Computer Science, Goethe-Universität Frankfurt, Germany and Department of Computer Science, University of Maryland at College Park, USA. Email: monemizadeh@em.uni-frankfurt.de and monemi@umiacs.umd.edu, Supported in part by MO 2200/1-1.

§IBM T.J. Watson Research Center. Email: krzysztof@onak.pl. Supported in part by a Simons Postdoctoral Fellowship and NSF grants 0732334 and 0728645.

¶Department of Computer Science, TU Dortmund. Email: christian.sohler@tu-dortmund.de. Supported in part by the German Research Foundation (DFG), grant So 514/3-2.

1 Introduction

Property testing studies relaxed decision problems in which one wants to distinguish objects that have a given property from those that are far from this property (see, e.g., [7]). Informally, an object $\mathcal{X}$ is ε-far from a property $\mathcal{P}$ if one has to modify at least an ε-fraction of $\mathcal{X}$'s representation to obtain an object with property $\mathcal{P}$, where ε is typically a small constant. Given oracle access to the input object, a typical property tester achieves this goal by inspecting only a small fraction of the input. Property testing is motivated by the need to understand how to extract information efficiently from massive structured or semi-structured data sets using small random samples.

One of the main and most successful directions in property testing is *testing graph properties*, as introduced in papers of Goldreich et al. [8, 9]. There are two popular models for this task, which make different assumptions about how the input graph is represented and how it can be accessed.

For a long time, the main research focus has been on the *adjacency matrix model*, designed typically for *dense* graphs [8]. In this model, after a sequence of papers, it was shown that testability of a property in constant time is closely related to Szemerédi partitions of the graph. More precisely, a property is testable in constant time[1] if and only if it can be reduced to testing finitely many Szemerédi partitions [1].

The *adjacency list model* has been designed mostly for *sparse* graphs. In the most standard scenario, the model is studied with an additional restriction that the degree of the graph is at most a certain predefined constant d [9]. Unlike in the adjacency matrix model, it is not yet completely understood what graph properties are testable in constant time in the adjacency list model. Known examples include all hyperfinite properties [16] (see also [3] and [12] for previous general results), connectivity, k-edge-connectivity, the property of being Eulerian [9], and the property of having a perfect matching [17]. On the other hand, some properties testable in constant time in the dense graph model, such as bipartiteness and 3-colorability, are known to require a superconstant number of queries [4, 9].

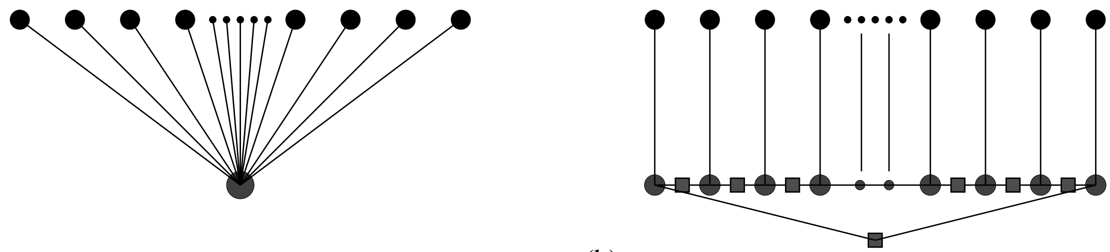
Even less is known about efficiently testable properties for sparse graphs that do not have a degree bound. It turns out the constant degree bound in the adjacency list model is essential for many of the results mentioned above. All constant-time testers mentioned above use the fact that in a graph with constant maximum degree for a small sample set one can explore all vertices that have constant distance to one of the sample vertices and then decide based on the obtained information. It is known that connectivity, k-edge-connectivity, and Eulerian graphs are testable in constant time [15]. However, no general results characterizing constant-time testable properties are known.

Bipartiteness. The problem of testing bipartiteness has been a great benchmark of the capabilities of property testing algorithms in various graph models. It was one of the first problems studied in detail in both the dense graph model [8] and the sparse graph model [9, 10]. Bipartiteness is known to be testable in $\widetilde{O}(1/\varepsilon^2)$ time in the dense graph model [2]. However, in the sparse graph model, it requires $\Omega(\sqrt{n})$ queries [9] and is testable in $\widetilde{O}(\sqrt{n}\cdot\varepsilon^{-O(1)})$ time [10], where n is the number of vertices. Kaufman et al. [13] show that the property is still testable in $\widetilde{O}(\sqrt{n}\cdot\varepsilon^{-O(1)}))$ time in the adjacency list model for graphs that have constant *average* degree.

Czumaj et al. [5] show in the bounded-degree model that if the underlying graph is planar, then any hereditary graph property[2], including bipartiteness, is testable in constant time. This approach can be generalized to any class of graphs that can be partitioned into constant-size components by removing εn edges

[1]Throughout the paper we say that a property is *testable in constant time* if there is a testing algorithm whose number of queries to the input is independent of the input size, possibly depending only on the proximity parameter ε.

[2]A graph property is *hereditary* if it is closed under vertex removals.



(a) (b)

Figure 1: An example of the process of splitting a vertex that reduces any graph into a graph of maximum degree at most 3 and that maintains planarity. For the graph in (a), Figure (b) depicts the splitting that is invariant to being bipartite.

of the graph, for any $\varepsilon > 0$. Graphs satisfying this property are called *hyperfinite*, and they include all bounded-degree minor-free graphs.

Hassidim et al. [12] show that in fact, the distance to most hereditary properties can be approximated in constant time in such graphs. These results are generalized in the recent work of Newman and Sohler [16], who show that in hyperfinite graphs, one can approximate the distance to *any graph property* in constant time. In particular, this implies that *any graph property is testable in constant time in hyperfinite graphs*, and therefore, in bounded-degree planar graphs.

The central goal of this paper is to initiate the research on the complexity of testing graph properties in general unbounded degree minor-free graphs. Our main technical contribution is the design and analysis of a constant time algorithm testing bipartiteness in arbitrary planar graphs. We show (in Theorem 4) that *a long enough constant-length random walk from a random vertex discovers an odd-length cycle in a graph far from bipartite with constant probability*. The result extends to an arbitrary family of minor-free graphs (Theorem 24 in Section 6).

1.1 Techniques

Our approach is based on a new analysis technique for random walks in planar (and minor-free) graphs. We first show that a planar graph that is far from bipartiteness has a linear number of edge-disjoint cycles of constant odd-length. Then we show a contraction procedure that preserves up to a constant factor the probability of discovering an odd-length cycle by a random walk. We show that after a constant number of contractions we obtain a multigraph in which the probability of discovering an odd-length cycle is lower bounded by a positive constant.

1.2 Approaches that do not work

Given that bipartiteness can be tested in constant time in planar graphs of bounded degree [5], it may seem that there is a simple extension of this result to arbitrary degrees. We now describe two natural attempts at reducing our problem to testing bipartiteness in other classes of graphs. We explain why they fail. We hope that this justifies our belief that new techniques are necessary to address the problem.

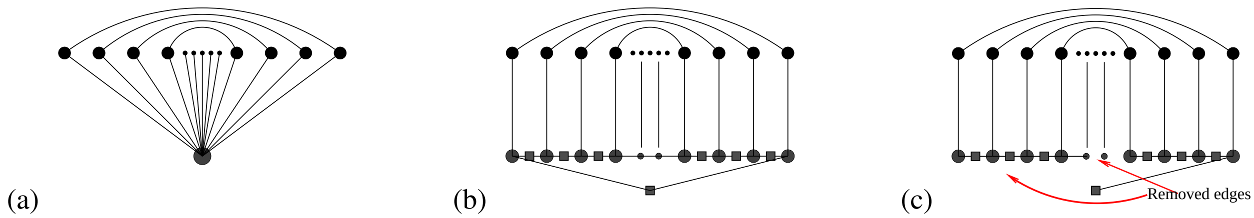


Figure 2: An example showing that the splitting construction from Figure 1 can reduce the distance from being bipartite. The planar graph in (a) (in which the i^{th} top vertex from the left is connected by an edge to the i^{th} top vertex from the right) has $\Theta(n)$ edge-disjoint cycles of length 3 and is ε-far from bipartite (one has to remove at least $\frac{n-1}{2}$ edges to obtain a bipartite graph). However, after the splitting, the obtained graph (Figure (b)) can be made bipartite just by removal of two edges: Figure (c) depicts a bipartite graph obtained after removal of such two edges: one of the two edges at the bottom and the middle edge in the split part.

1.2.1 Transforming into a constant-degree planar graph

The first and possibly the most natural approach to designing a constant-time algorithm for testing bipartiteness in arbitrary planar graphs would be to extend the known constant-time algorithm for *bounded-degree* planar graphs [5]. This could be achieved by first transforming an input planar graph G with an arbitrary maximum-degree into a planar graph $G^\star$ with bounded-degree and then running the tester for $G^\star$ to determine the property for G. However, we do not see any transformation that could work well and we do not expect any such transformation to exist.

For example, it is known that one can transform any graph into one with maximum degree at most 3 by splitting every vertex of degree $d > 3$ into d vertices of degree 3. It is also easy to ensure that this reduction maintains the planarity, and also the property of being bipartite (see Figure 1). However, there are two properties that are not maintained: one is the distance from being bipartite (see Figure 2) and another is that the access to the neighboring nodes requires more than constant time (though this can be "fixed" if one allows each vertex to have its adjacency list ordered consistently with some planar embedding). In particular, Figure 2 depicts an example of a planar graph that is originally ε-far from bipartite, but after the transformation it suffices to remove 2 edges to obtain a bipartite graph.

1.2.2 Substituting high-degree vertices with expanders

Another transformation of the graph is considered by Kaufman et al. [13]. They replace every high degree vertex with a constant-degree bipartite expander. While they prove that this construction preserves the distance, it is clear that it cannot preserve the planarity, since planar graphs are not expanders. For expanders, testing bipartiteness requires $\Omega(\sqrt{n})$ queries [9].

2 Preliminaries

Bipartiteness. A graph is *bipartite* if one can partition its vertex set into two sets A and B such that every edge has one endpoint in A and one endpoint in B. We also frequently use the well known fact that a graph is bipartite if and only if it has no odd-length cycle.

We now formally introduce the notion of being far from bipartiteness.[3] The notion is parameterized by

[3]The standard definition of being ε-far (see for example the definition in [13]) expresses the distance as the fraction of edges

a distance parameter $\varepsilon > 0$.

Definition 1 A graph $G = (V, E)$ is *ε-far from bipartite* if one has to delete more than $\varepsilon|V|$ edges from G to obtain a bipartite graph.

Property testing. We are interested in finding a *property testing algorithm* for bipartiteness in planar graphs, i.e., an algorithm that inspects only a very small part of the input graph, and accepts bipartite planar graphs with probability at least $\frac{2}{3}$, and rejects planar graphs that are ε-far away from bipartite with probability at least $\frac{2}{3}$, where ε is an additional parameter.

Our algorithm always accepts every bipartite graph. Such a property testing algorithms is said to have *one-sided error*.

Access model. The access to the graph is given by an *oracle*. We consider the oracle that allows two types of queries:

- *Degree queries:* For every vertex $v \in V$, query the degree of v.
- *Neighbor queries:* For every vertex $v \in V$, query its i^{th} neighbor.

Observe that by first querying the degree of a vertex, we can always ensure that the i^{th} neighbor of the vertex exists in the second type of query. In fact, in the algorithm that we describe in this paper, the neighbor query can be replaced with a weaker type of query: *random neighbor query*, which returns a random neighbor of a given vertex v; each time the neighbor is chosen independently and uniformly at random.

The *query complexity* of a property testing algorithm is the number of oracle queries it makes.

Basic properties of planar graphs. We extensively use the following well-known properties of planar graphs. The graph $G'(V', E')$ obtained by the *contraction* of an edge $(u, v) \in E$ into vertex u is defined as follows: $V' = V \setminus \{v\}$ and $E' = \{(x, y) \in E : x \neq v \wedge y \neq v\} \cup \{(x, u) : (x, v) \in E \wedge x \neq u\}$. A graph G' that can be obtained from a graph G via a sequence of edge removals, vertex removals, and edge contractions is called a *minor* of G.

We use the following well-known property of planar graphs.

Fact 2 *Any minor of a planar graph is planar.*

Furthermore, we use the following upper bound on the number of edges in a planar graph, which follows immediately from Euler's formula.

Fact 3 *For any planar graph $G = (V, E)$ (with no self-loops or parallel edges), $|E| \leq 3|V| - 6$.*

We remark that for any class of graphs $\mathcal{H}$ that is defined by a finite collection of forbidden minors, similar statements are true, i.e., if $G \in \mathcal{H}$, then any minor of G also belongs to $\mathcal{H}$ and if $G = (V, E) \in \mathcal{H}$, then G has $O(|V|)$ edges (where the constant in the big O notation depends on the set of forbidden minors).

that must be modified in $G = (V, E)$ to obtain a bipartite graph. Compared to our Definition 1, instead of deleting $\varepsilon|V|$ edges, one can delete $\varepsilon|E|$ edges. For any class of graphs with an excluded minor, the number of edges in the graph is upper bounded by $C \cdot |V|$, where C is a constant. Moreover, unless the graph is very sparse (i.e., most of its vertices are isolated, in which case even finding a single edge in the graph may take a large amount of time), the number of edges in the graph is at least $\Omega(|V|)$. Thus, under the standard assumption that $|E| = \Omega(|V|)$, the ε in our definition and the ε in the previous definitions remain within a constant factor. We use our definition of being ε-far for simplicity; our analysis can be extended to the standard definition of being ε-far in a straightforward way.

Graph structures. We consider three types of graph structures:

Simple graphs: We use the standard font, e.g., $G = (V, E)$, to denote simple graphs, which have no multiple edges and no self-loops. Unless we say otherwise, throughout the paper we refer to simple graphs as graphs without explicitly mentioning that they are simple.

Multigraphs: We use the calligraphic font, e.g., $\mathcal{G} = (V, E)$, to denote graphs that may have multiple edges connecting the same pair of vertices and an arbitrary number of self-loops at each vertex.

Parity graphs: Finally, we use the blackboard bold font, e.g., $\mathbb{G} = (V, E, w, \ell)$, to denote *parity graphs* that are multigraphs with the following additional weights and labels. First, each vertex v of a parity graph $\mathbb{G}$ has a non-negative real weight $w(v)$. Second, each edge e (including self-loops) is labeled as either even or odd, i.e., it has a label $\ell(e) \in \{\mathbf{0}, \mathbf{1}\}$. Parallel edges (or self-loops) may have distinct labels.

In this paper, all of the above types of graph structures are planar.

Cycles. A *cycle* of length k is a sequence of vertices $(v_0, \dots, v_k)$ with $(v_i, v_{i+1}) \in E$, for $0 \le i < k$, $v_0 = v_k$, and $v_i \ne v_j$ for $0 < i < j \le k$. We stress that, in contrast to the standard definition, the sequence (u, v, u), for distinct $u, v \in V$, denotes a cycle of length 2 and the sequence (u, u), $u \in V$, denotes a self-loop, i.e., a cycle of length 1. For a set $\mathcal{C}$ of cycles on a set of vertices V, we use $\mathcal{G}(\mathcal{C})$ to denote the multigraph induced by $\mathcal{C}$, i.e., the multigraph $\mathcal{G}(U, E)$ where the multiset E is the set of edges of the cycles in $\mathcal{C}$ and $U \subseteq V$ is the set of vertices that are incident to edges in $\mathcal{C}$. If the cycles in $\mathcal{C}$ are edge disjoint, then $\mathcal{G}(\mathcal{C})$ is a graph.

We also consider *parity cycles* $\langle (v_0, \dots, v_k), w, \ell \rangle$ where $(v_0, \dots, v_k)$ is a cycle in which every vertex v_i, $0 \le i < k$, is assigned a non-negative weight $w(v_i)$ and every edge $e = (v_i, v_{i+1})$ is labeled as either even or odd, i.e., $\ell(e) \in \{\mathbf{0}, \mathbf{1}\}$. The parity of a parity cycle is the parity of the set of labels, i.e., the parity of the sum of labels. We refer to parity cycles with odd parity as *odd parity cycles*. The *total weight* $W(\mathfrak{c})$ of a parity cycle $\mathfrak{c} = \langle (v_0, \dots, v_k), w, \ell \rangle$ is the sum of weights of its vertices, i.e., $W(\mathfrak{c}) = \sum_{i=0}^{k-1} w(v_i)$.

For a set $\mathbb{C}$ of parity cycles, we write $\mathbb{G}(\mathbb{C})$ to refer to the induced parity graph with the following properties. First, the underlying multigraph is $\mathcal{G}(\mathcal{C})$, where $\mathcal{C}$ is the set of cycles in $\mathbb{C}$ with weights and labels removed. Second, the parities of edges from $\mathbb{C}$ are preserved in $\mathbb{G}(\mathbb{C})$. Third, the weight of each vertex v in $\mathbb{G}(\mathbb{C})$ is defined as the sum of v's weights in all cycles $\mathfrak{c} \in \mathbb{C}$ that contain it, i.e.,

$$w(v) = \sum_{\mathfrak{c} \in \mathbb{C} \text{ s.t. } v \text{ lies on } C} w_{\mathfrak{c}}(v)$$

where $w_{\mathfrak{c}}$ is the weight function for a parity cycle $\mathfrak{c}$.

Notation. In the remainder of the paper we use several constants depending on ε. We use lower case Greek letters to denote constants that are typically smaller than 1 (e.g., $\delta_i(\varepsilon)$) and lower case Latin letters to denote constants that are usually larger than 1 (e.g., $f_i(\varepsilon)$). All these constants are always positive.

3 Algorithm Random Bipartiteness Exploration

We first describe our algorithm for testing bipartiteness of planar graphs with arbitrary degree and provide the high level structure of its analysis. The technical details of the two main lemmas appear in Sections 4 and 5.

Random-Bipartiteness-Exploration (G, ε):

- Repeat $f(\varepsilon)$ times:
  - Pick a random vertex $v \in V$.
  - Perform a random walk of length $g(\varepsilon)$ from v.
  - If the random walk found an odd-length cycle, then **reject**.
- If none of the random walks found an odd-length cycle, then **accept**.

Theorem 4 *Let G be a planar graph. There are positive functions f and g such that*

- *if G is bipartite, then* **Random-Bipartiteness-Exploration**(G, ε) *accepts G, and*
- *if G is ε-far from bipartite, then* **Random-Bipartiteness-Exploration**(G, ε) *rejects G with probability at least* 0.99.

We start by observing that the first claim is obvious: if G is bipartite, then every cycle in G is of even length and hence **Random-Bipartiteness-Exploration** always accepts. Thus, to prove Theorem 4, it suffices to show that if G is ε-far from bipartite, then **Random-Bipartiteness-Exploration** rejects G with probability at least 0.99. Therefore, from now on, we assume that the input graph G is ε-far from bipartite for some constant $\varepsilon > 0$. Furthermore, note that it suffices to show that a *single* random walk of length $g(\varepsilon)$ finds an odd-length cycle with probability at least $5/f(\varepsilon)$. Indeed, this would immediately imply that $f(\varepsilon)$ independent random walks detect at least one odd-length cycle with probability at least $1 - (1 - 5/f(\varepsilon))^{f(\varepsilon)} \geq 1 - e^{-5} \geq 0.99$. Therefore, in the remainder of the paper, we analyze algorithm **Random-Walk**(G, t) below. We have to prove that there are functions $t(\varepsilon)$ and $\lambda(\varepsilon)$ such that for every planar graph G that is ε-far from bipartite, **Random-Walk**$(G, t(\varepsilon))$ finds an odd-length cycle with probability at least $\lambda(\varepsilon) = 5/f(\varepsilon)$. This would imply Theorem 4.

Random-Walk (G, t):

- Pick a random vertex $v \in V$.
- Perform a random walk of length t from v.
- If the random walk found an odd-length cycle, then **reject**.
- If not, then **accept**.

In order to prove that algorithm **Random-Walk** finds an odd-length cycle in any planar graph G that is ε-far from bipartite, we design a graph transformation that approximately preserves the probability of detecting an odd length cycle. Roughly speaking, a graph G' is *random walk invariant* with respect to a graph G if finding an odd length cycle by random walks in G is not (much) harder than finding an odd length cycle by random walks in G', i.e., by increasing the length of the walk by some multiplicative constant, we can make sure that the success probability drops at most by a multiplicative constant. The main goal of the proof is to construct a sequence $G = G_0, \ldots, G_k$ of graphs such that G_{i+1} is random walk invariant with respect to G_i, for all $0 \leq i < k$, and such that in G_k we can easily find an odd-length cycle. Unfortunately, the construction does not turn out to be straightforward. In particular, we need two different types of reductions and we have to extend the definition of random walk invariance to parity graphs.

Our first step is to formalize the notion of random walk invariance.

Definition 5 (Random walk invariant) Let $r, t > 0$ be integral constants and let $\xi > 0$ be an arbitrary constant. Let $G = (V, E)$ be a graph and let $H = (U, F)$ be another graph on vertex set $U \subseteq V$. Let λ denote the probability that **Random-Walk**(H, t) finds an odd-length cycle. The graph H is called (r, t, ξ)-*random walk invariant* with respect to G if the probability that **Random-Walk**(G, $r \cdot t$) finds an odd-length cycle is at least $\xi \cdot \lambda$.

Next we define the first type of reduction (see Section 4 for a proof).

Lemma 6 (First reduction) *In every planar graph $G = (V, E)$ that is ε-far from bipartite, there exists a collection $\mathcal{C}$ of $\Omega_\varepsilon(|V|)$ edge-disjoint odd-length cycles of length at most $k = k(\varepsilon) = O_\varepsilon(1)$ such that for every $t > 0$, there is $\xi = \xi(t, \varepsilon)$ such that $\mathcal{G}(\mathcal{C})$ is $(1, t, \xi)$-random walk invariant with respect to G.*

Lemma 6 states that it suffices to prove that **Random-Walk** finds with probability at least $\lambda(\varepsilon)$ an odd-length cycle in any planar graph that is induced by a set $\mathcal{C}$ of $\Omega_\varepsilon(n)$ short odd-length cycles. It is tempting to attempt to prove that the algorithm finds one of the cycles in $\mathcal{C}$. Unfortunately, this approach applied directly is doomed to fail, as one can see in Figure 3. Instead, we prove a sufficient result that **Random-Walk** finds *a* short odd-length cycle that is a *combination of the cycles from* $\mathcal{C}$ (with probability at least $\lambda(\varepsilon)$).

Armed with the first reduction we can concentrate on graphs that are induced by sets of short odd-length cycles. More precisely, we focus on parity graphs induced by short odd parity cycles. In order to prove that algorithm **Random-Walk** finds an odd-length cycle in the original graph, we define a second type of reduction. Given a graph $\mathbb{G}(\mathbb{C})$ induced by a set of odd parity cycles $\mathbb{C}$ of length at most k, we find another set of odd parity cycles, $\mathbb{C}'$, inducing a parity graph $\mathbb{G}(\mathbb{C}')$ that is random walk invariant with respect to the first parity graph (for some appropriate generalization of random walks to parity graph). The number of parity cycles in $\mathbb{C}'$ decreases by at most a constant with respect to $\mathbb{C}$ and the length of each cycle other than a self-loop decreases by at least 1. The main high level idea of the second part of the proof is to apply this reduction several times until we have reached a graph that contains $\Omega_\varepsilon(n)$ self-loops (which are odd parity cycles) and no other edges. In this resulting graph, an odd parity cycle is easily detected by a random walk.

The set $\mathbb{C}'$ is obtained from $\mathbb{C}$ by cycle removals and edge contractions. Since we use edge contractions, edges of parity cycles may represent paths of the original graph in the course of the reduction. Since we are only interested in odd-length cycles, we need to remember the parity of paths represented by a single edge, which is achieved by labels **0** and **1**. An odd parity cycle was constructed from a cycle of odd length in the initial set of cycles provided by the first reduction.

Furthermore, edge contractions may lead to situations where a pair of vertices appears as consecutive vertices on more than one cycle. This is why parity graphs build on multigraphs, which allow for parallel edges and self-loops. Edge contractions also lead to the situation that some vertices represent *a set of vertices* in the original graph. Vertex weights are used to address this situation. The starting point in a modified random walk procedure is selected with probability proportional to vertex weights. In our construction, the cycles in the set $\mathbb{C}$ are weighted. As described in Section 2, this defines vertex weights of the induced graph $\mathbb{G}(\mathbb{C})$ (the weight of a vertex is simply the sum of the weights of the vertex in the incident cycles). If we contract an edge (u, v) of a cycle into v, we add the weight of u to the weight of v to make sure that the total weight of the cycle is preserved. This way, we also make sure that the weight of any set of vertices

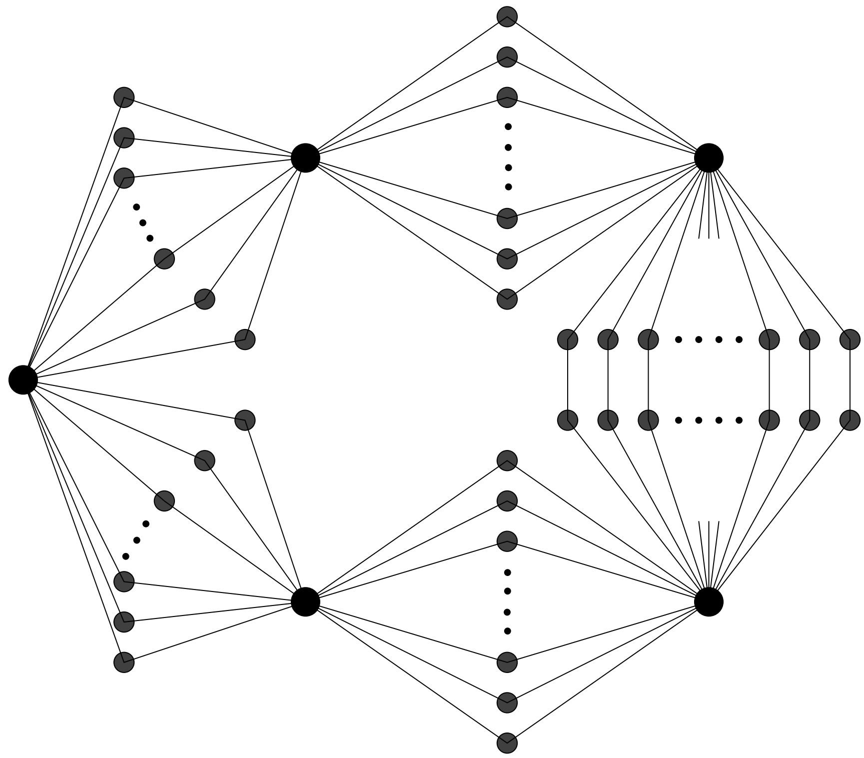


Figure 3: A planar graph G with $\frac{n-5}{6}$ edge-disjoint odd-length cycles. Each of the five high degree vertices has degree exactly $\frac{2(n-5)}{6}$, and the edge-disjoint cycles are of length 11 each. Observe that if $\mathcal{C}$ is any fixed set of $\frac{n-5}{6}$ edge-disjoint cycles of length 11, then the probability that a single constant-length random walk discovers one of the cycles in $\mathcal{C}$ is $n^{-\Omega(1)}$. Nevertheless, a single random walk of length 12 finds *an* odd-length cycle with probability at least 2^{-11}.

is preserved in the edge contraction (i.e., if a vertex disappears, its weight is assigned to the vertex that represents the disappearing vertex). We make sure that initially the total weight of each cycle is $\Theta_\varepsilon(1)$. The assignment of weights may be interpreted as the share of the weight of a vertex (and thus of the probability to start in at this cycle), a given cycle has.

Since the result of the first reduction is a set of constant-length cycles, we need to assign weights so that the total weight of each cycle is bounded away from 0 and is at most some constant, and so that each vertex has weight 1 (since we are sampling from the uniform distribution).

Lemma 7 *Let $\mathcal{C}$ be a set of edge-disjoint cycles of length at most k such that $\mathcal{G}(\mathcal{C})$ is a planar graph (with no self-loops or multiedges). The cycles in $\mathcal{C}$ can be assigned weights $w_\mathfrak{c}$ such that each cycle has total weight in the range $[1/2, k]$ and each vertex in the induced graph $\mathcal{G}(\mathcal{C})$ has weight* 1.

Proof. As long as the set of cycles is non-empty, we repeat the following procedure: Since $\mathcal{G}(\mathcal{C})$ is planar, Fact 3 ensures that there is a non-isolated vertex v of degree at most 5. Let t_v be the number of cycles that share this vertex; we know that $1 \le t_v \le 2$. For each cycle $\mathfrak{c} \in \mathcal{C}$ that shares vertex v, we set the weight of v in $\mathfrak{c}$ to $1/t_v$ and we remove $\mathfrak{c}$ from $\mathcal{C}$.

This construction ensures that no vertex has weight greater than 1 and every cycle has at least one vertex of weight at least $1/2$. To make the weights of all vertices equal to 1, for any vertex whose weight is smaller than 1, we increase it to 1 and assign it to any of the cycles originally in $\mathcal{C}$. Since each cycle $\mathfrak{c}$ has length at most k, we get $W(\mathfrak{c}) \le k$. ■

We now describe a modified version of algorithm **Random-Walk** for parity graphs. We refer to it as **RWPG** (Random Walk for Parity Graphs).

RWPG $(\mathbb{G}, t)$:

- Pick a random vertex v in $\mathbb{G}$ with probability proportional to its weight.
- Starting from v, run a random walk of length t (the details of how each step is selected are below).
- If the random walk found an odd cycle, then **reject**.
- Otherwise, **accept**.

In the process above, at each vertex the random walk selects every incident edge with the same probability, with one technical exception: a self-loop's ends are treated as independent edges, i.e., a self-loop is followed with probability twice the probability of following an incident edge that connects two different vertices.

The definition of random walk invariance extends to parity graphs in a straightforward way.

Definition 8 (Random walk invariant for parity graphs) Let $r, t > 0$ be integral constants and let $\xi > 0$ be an arbitrary constant. Let $\mathbb{G} = (V, E, w, \ell)$ and $\mathbb{G}' = (V', E', w', \ell')$ be parity graphs, where $V' \subseteq V$. Let λ denote the probability that **RWPG**$(\mathbb{G}', t)$ finds an odd parity cycle. The multigraph $\mathbb{G}'$ is called (r, t, ξ)-*random walk invariant* with respect to $\mathbb{G}$, if the probability that **RWPG**$(\mathbb{G}, r \cdot t)$ finds an odd parity cycle is at least $\xi \cdot \lambda$.

We introduce an auxiliary notion of a *witness cycle* that helps simplify the statements of our reductions.

Definition 9 (Witness cycles) A parity cycles is a *witness cycle* if it has odd parity. Furthermore, for α and β such that $0 < \alpha \le \beta$, we say that a parity cycle is an (α, β)-*witness cycle* if it is a witness cycle and the total weight of the cycle is in the range $[\alpha, \beta]$.

We next state the properties of our second reduction (for a proof of Lemma 10, see Section 5).

Lemma 10 (Second reduction) *Let $\mathbb{C}$ be a set of (α, β)-witness cycles on a vertex set V, where α and β are real numbers such that $0 < \alpha \le \beta$. If $\mathbb{G}(\mathbb{C})$ is planar and if each witness cycle in $\mathbb{C}$ has length at most k, $k \ge 2$, then there exists a set of (α, β)-witness cycles $\mathbb{C}'$ on V that satisfies the following conditions:*

- $\mathbb{G}(\mathbb{C}')$ *is planar,*
- *every cycle in $\mathbb{C}'$ has length at most $k - 1$,*
- $|\mathbb{C}'| \ge \sigma|\mathbb{C}|$, *with $\sigma = \sigma(k)$, and*
- *there is $\xi = \xi(k) > 0$ such that $\mathbb{G}(\mathbb{C}')$ is $(3, 3^{k-1}, \xi\alpha/\beta)$-random walk invariant with respect to $\mathbb{G}(\mathbb{C})$.*

We also use the following observation.

Observation 11 *Let $\mathbb{G} = (V, E, w, \ell)$, $\mathbb{G}' = (V', E', w', \ell')$ and $\mathbb{G}' = (V'', E'', w'', \ell'')$ be parity graphs with $V'' \subseteq V' \subseteq V$. If $\mathbb{G}'$ is $(r, r't, \xi)$-random walk invariant with respect to $\mathbb{G}$ and $\mathbb{G}''$ is (r', t, ξ')-random walk invariant with respect to $\mathbb{G}'$, then $\mathbb{G}''$ is $(rr', t, \xi\xi')$-random walk invariant with respect to $\mathbb{G}$.*

Given Lemma 10 and Observation 11, we can now prove Theorem 4.

Let G be any planar graph ε-far from bipartite. By our previous discussion, it suffices to prove that for a sufficiently large $t = t(\varepsilon)$, **Random-Walk**(G, t) finds an odd-length cycle with probability at least $\eta = \eta(\varepsilon) > 0$.

Let n be the number of vertices in G. By Lemma 6, there is a set $\mathcal{C}$ of $\Omega_\varepsilon(n)$ odd-length cycles, each of length at most $k = k(\varepsilon)$, such that $\mathcal{G}(\mathcal{C})$ is $(1, t, \zeta)$-random walk invariant with respect to G, where $\zeta = \zeta(\varepsilon) > 0$ and t is arbitrary. We apply Lemma 7 to $\mathcal{C}$ and label each edge with $\mathbf{1}$ (odd parity) in order to obtain a set $\mathbb{C}$ of $(1/2, k)$-witness cycles. There is a one-to-one correspondence between every path in $\mathcal{G}(\mathcal{C})$ and $\mathbb{G}(\mathbb{C})$. In particular, **Random-Walk** finds an odd-length cycle in $\mathcal{G}(\mathcal{C})$ with the same probability as **RWPG** finds an odd parity cycle in $\mathbb{G}(\mathbb{C})$.

Let $\mathbb{C}_k = \mathbb{C}$. We repeatedly apply Lemma 10. First, we obtain $\mathbb{C}_{k-1}$ from $\mathbb{C}_k$, then $\mathbb{C}_{k-2}$ from $\mathbb{C}_{k-1}$, and so on, until we obtain a set $\mathbb{C}_1$. Each of the sets $\mathbb{C}_i$ consists of $(1/2, k)$-witness cycles. In each step, their number decreases by at most a factor of σ, a positive constant depending on ε. Thus, after k rounds, $\Omega_\varepsilon(n)$ witness cycles remain in $\mathbb{C}_1$. By the properties guaranteed by Lemma 10, the length of all cycles in $\mathbb{C}_1$ is 1. Hence, every vertex in $\mathbb{G}(\mathbb{C}_1)$ is incident to odd parity self-loops. Therefore, independently of the choice of the starting vertex, **RWPG** discovers an odd parity cycle in a single step.

We set $t = 3^{k-1}$, which implies that t is a function of only ε. Lemma 10 and Observation 11 guarantee that $\mathbb{G}(\mathbb{C}_1)$ is $(t, 1, \xi)$-random walk invariant with respect to $\mathbb{G}(\mathbb{C}_k)$ for a positive value $\xi = \xi(k)$. Again, for our choice of k, ξ depends only on ε.

It follows from our discussion that **RWPG**$(\mathbb{G}(\mathbb{C}), t)$ discovers an odd parity cycle with probability at least ξ. Also **Random-Walk**$(\mathcal{G}(\mathcal{C}), t)$ discovers an odd-length cycle with probability at least ξ, which implies in turn that **Random-Walk**(G, t) discovers an odd-length cycle with probability at least $\xi \cdot \zeta > 0$, where both ξ and ζ depend only on ε. This completes the proof of Theorem 4.

4 Proof of Lemma 6: Large subgraphs induced by odd-length cycles

In this section we give a proof of Lemma 6. We begin with a useful graph decomposition lemma that follows directly from the Klein-Plotkin-Rao decomposition theorem [14].

Lemma 12 [14] *Let $G = (V, E)$ be a planar graph and let δ be a parameter, $0 < \delta < 1$. There is a set of at most $\delta|E|$ edges in G whose deletion decomposes G into connected components, where the distance (in the original graph) between any two nodes in the same component is $O(1/\delta^2)$.*

Lemma 12 leads to the following key property of planar graphs that are ε-far from bipartite.

Lemma 13 *Let G be a planar graph. If G is ε-far from bipartite, then G has at least $\frac{\varepsilon n}{q(\varepsilon)}$ edge-disjoint odd-length cycles of length at least 3 and at most $\frac{q(\varepsilon)}{2}$ each, where $q(\varepsilon) = O(1/\varepsilon^2)$.*

Proof. We define $q(\varepsilon)$ such that $q(\varepsilon) \geq 6$ and $\frac{q(\varepsilon)}{6}$ is an upper bound on the diameter (in the original graph) of components in the decomposition given by Lemma 12 with $\delta = \frac{\varepsilon}{2}$.

We find the cycles one by one. Suppose that we have already found in G a set of k edge-disjoint odd-length cycles of length at most $\frac{q(\varepsilon)}{2}$ each, where $k < \frac{\varepsilon n}{q(\varepsilon)}$. We show the existence of one more such cycle, which by induction yields the lemma.

Let $G^\star$ be the subgraph of G obtained by removing the k edge-disjoint odd-length cycles of length at most $\frac{q(\varepsilon)}{2}$ each. Since $k < \frac{\varepsilon n}{q(\varepsilon)}$, $G^\star$ is obtained by removing less than $\frac{\varepsilon n}{2}$ edges, and hence $G^\star$ is $\varepsilon/2$-far from bipartite. Apply Lemma 12 to $G^\star$ with $\delta = \frac{\varepsilon}{2}$ and let H be the resulting decomposition. Since $G^\star$ is $\varepsilon/2$-far from bipartite, H is not bipartite. Let us consider a connected component C_H of H that is not bipartite and let v be a vertex from C_H. Build a BFS tree from v in $G^\star$. Since C_H is not bipartite, there must exist two vertices u_1 and u_2 in C_H that have the same distance from v and that are connected by an edge in H (otherwise, we could define a bipartition of C_H by the parity of the distance from v in the BFS tree). Let v' be the last common vertex on the paths from v to u_1 and from v to u_2 in the BFS tree. The tour that starts at v', goes to u_1 via the BFS tree edges, then takes the edge connecting u_1 and u_2, and finally returns to v' via the BFS tree edges is an odd-length cycle of length at most $\frac{q(\varepsilon)}{3} + 1 \le \frac{q(\varepsilon)}{2}$. ■

Our next lemma states that in order to show that a t-step random walk from a random start vertex finds an odd-length cycle in a planar graph G that is ε-far from bipartite, it suffices to show that the random walk finds an odd-length cycle in a subgraph of G induced by a linear number of edge-disjoint odd-length cycles of constant length. The combination of the lemma with Lemma 13 immediately implies Lemma 6.

Lemma 14 *Let $G = (V, E)$ be a planar graph that has a set $\mathcal{C}^\star$ of at least αn edge-disjoint odd-length cycles in G with $\alpha > 0$, and let $t > 0$ be an integer. Then there is $\zeta = \zeta(t, \alpha) > 0$ and there is a subset $\mathcal{C}$ of $\mathcal{C}^\star$ of size at least $\frac{1}{2}\alpha n$ such that $\mathcal{G}(\mathcal{C})$ is $(1, t, \zeta)$-random walk invariant with respect to G.*

Proof. To construct the subset $\mathcal{C}$, we first delete some cycles from $\mathcal{C}^\star$. The process of deleting the cycles is based on the comparison of the original degree of the vertices with the current degree in $\mathcal{G}(\mathcal{C}^\star)$. To implement this scheme, we write $\deg_G(v)$ to denote the degree of v in the original graph G and we use the term *current degree* of a vertex v to denote its current degree in the graph $\mathcal{G}(\mathcal{C}^\star)$ induced by the *current* set of cycles $\mathcal{C}^\star$ (where "current" means at a given moment in the process).

We repeat the following procedure as long as possible: if there is a vertex $v \in V$ with current degree in $\mathcal{G}(\mathcal{C}^\star)$ at most $\frac{1}{12}\alpha \deg_G(v)$, then we delete from $\mathcal{C}^\star$ all cycles going through v in $\mathcal{C}^\star$. To estimate the number of cycles deleted, we charge to v the number of deleted cycles in each such operation.

Let $\mathcal{C}$ be the remaining set of cycles from $\mathcal{C}^\star$. Observe that each $v \in V$ can be processed above at most once. Indeed, once v has been used, it becomes isolated, and hence it is not used again. Therefore, at most $\frac{1}{1}2\alpha \deg_G(v)$ cycles from $\mathcal{C}^\star$ can be charged to any single vertex. This, together with the inequality $\sum_{v \in V} \deg_G(v) \le 6n$ by planarity of $\mathcal{G}(\mathcal{C}^\star)$, implies that the total number of cycles removed from $\mathcal{C}^\star$ is upper bounded by $\sum_{v \in V} \frac{1}{12}\alpha \deg_G(v) \le \frac{1}{2}\alpha n$. Since $|\mathcal{C}^\star| \ge \alpha n$, we conclude that $|\mathcal{C}| \ge |\mathcal{C}^\star| - \frac{1}{2}\alpha n \ge \frac{1}{2}\alpha n$.

We have constructed a subset $\mathcal{C}$ of $\mathcal{C}^\star$ of size at least $\frac{1}{2}\alpha n$ such that the degree of every vertex v of $\mathcal{G}(\mathcal{C})$ is greater than $\frac{1}{12}\alpha \deg_G(v)$ (i.e., at least an $\frac{\alpha}{12}$ fraction of its original degree in G). We now use this property to show that if the probability that a t-step random walk starting from a random vertex in $\mathcal{G}(\mathcal{C})$ finds an odd-length cycle is p, then a t-step random walk starting from a random vertex in G finds an odd-length cycle with probability at least $p \cdot \zeta$, for appropriately chosen $\zeta = \zeta(t, \alpha)$. This suffices to prove the theorem.

Let us consider a fixed sequence of $t+1$ vertices $\langle x_0, x_1, \ldots, x_t \rangle$ in $\mathcal{G}(\mathcal{C}^\star)$ with $(x_i, x_{i+1}) \in E_{\mathcal{C}}$, $0 \le i \le t-1$, that contains an odd-length cycle $\mathfrak{c}$. Our claim is that, if the probability of this fixed sequence to be chosen as a t-step random walk in $\mathcal{G}(\mathcal{C})$ is p', then it is at least $\zeta \cdot p'$ in G. This suffices to finish the proof of the theorem.

Since $\mathcal{G}(\mathcal{C})$ may have fewer vertices than G, the probability of choosing x_0 as a start vertex may be larger. However, since G is planar (and has no self-loops), every cycle in G has at least 3 edges. Furthermore, as a subgraph of a planar graph, $\mathcal{G}(\mathcal{C})$ is also planar. Since $\mathcal{C}$ is of size at least $\alpha n/2$, $\mathcal{G}(\mathcal{C})$ has at least $3\alpha n/2$ edges. Now it follows from Fact 3 that $\mathcal{G}(\mathcal{C})$ has at least $\alpha n/2$ vertices. Thus, the probability of choosing x_0 in G is no smaller than $\alpha/2$ times the probability of choosing it in $\mathcal{G}(\mathcal{C})$.

Since x_0 is a start vertex and is not isolated in $\mathcal{G}(\mathcal{C})$, we must have $\deg_{\mathcal{G}(\mathcal{C})}(x_0) > \frac{\alpha}{12}\deg_G(x_0)$. Therefore, when the random walk in G chooses a neighbor of x_0, it chooses x_1 with probability at least $\frac{\alpha}{12}$ times the probability that the random walk in $\mathcal{G}(\mathcal{C})$ chooses x_1. The same arguments can be used to argue that if the random walk in G has chosen any vertex x_i, then it also chooses x_{i+1} with probability at least $\frac{\alpha}{12}$ times the respective probability for the random walk in $\mathcal{G}(\mathcal{C})$. Therefore, if the random walk in $\mathcal{G}(\mathcal{C})$ is chosen with probability p' then the probability of choosing the same random walk in G is at least $(\frac{\alpha}{12})^t \cdot p'$. Summing up over all sequences of $t+1$ vertices $\langle x_0, x_1, \ldots, x_t\rangle$ in $\mathcal{G}(\mathcal{C})$ with $(x_i, x_{i+1}) \in E_{\mathcal{C}}$, $0 \le i \le t-1$, that contain an odd length cycle, we obtain the claim with $\zeta = \frac{\alpha}{2} \cdot (\frac{\alpha}{12})^t$. ■

5 Proof of Lemma 10: Parity graphs induced by witness cycles

We continue with the assumption that the input graph G is planar and ε-far from bipartite. Recall that by Lemma 6, it suffices to consider graphs that are induced by a collection of $\Omega_\varepsilon(n)$ odd-length cycles. To show that in such a graph a random walk finds an odd-length cycle with constant probability, we introduce another reduction. We start by describing the intuition behind the proof.

5.1 Overview

Even though Lemma 10 applies to parity graphs, we describe the intuition behind its proof for multigraphs induced by a collection of odd-length cycles.

We first observe that if all vertices in $\mathcal{G}(\mathcal{C})$ have constant degree, then a single random walk finds an odd-length cycle with constant probability. This is because with constant probability the starting vertex is on some odd-length cycle $\mathfrak{c}$ from $\mathcal{C}$ and in each step we follow $\mathfrak{c}$ with constant probability. Since $\mathfrak{c}$ is an odd-length cycle of length at most $t(\varepsilon)$, the observation follows. Thus, the hard part is to deal with vertices whose degree is not constant. An example that captures many of the difficulties of the problem is given in Figure 3. In this example, we have many "parallel" cycles that intersect at several vertices of high degree. However, many vertices of the cycles also have constant degree. By the linear bound on the number of edges in a planar graph, this is the case for any planar graph.

Our main idea is to identify paths of length two with the middle vertex of constant degree (or with high degree, but with a constant number of distinct neighbors), and then contract the middle vertex into one of the end vertices. The motivation behind this contraction is that if a random walk takes the first edge of such a path, then with constant probability it also follows the second edge. Thus, from the point of the analysis of the random walk, we can view this path as a single edge. Furthermore, since many cycles must have at least one subpath whose middle vertex has constant degree, we can make sure that a constant fraction of cycles has at least one subpath that is contracted. If we keep only the cycles of $\mathcal{C}$ that have been contracted, we end up with a linear-size set $\mathcal{C}'$ of cycles with length reduced by at least 1. Furthermore, since planarity is closed under edge contraction and edge removal, the multigraph $\mathcal{G}(\mathcal{C}')$ is planar again. This allows us to repeatedly apply our reduction until all cycles collapse to self-loops. Such a repeated application may be necessary since $\mathcal{G}(\mathcal{C}')$ may still have vertices of high degree. Finally, since a constant fraction of edges belongs to self-loops, a random walk traverses a self-loop with constant probability. Hence, by the properties of our reductions, a random walk finds an odd-length cycle in the original graph with constant probability.

5.2 The framework

We begin by introducing several basic definitions and concepts.

5.2.1 Cycle contractions

In our analysis, we apply *a sequence of contractions* to the parity graph induced by a set of witness cycles. This operation contracts some paths of length 2 to edges and simplifies the structure of the parity graph, as needed in our analysis.

We begin with definitions used. If $\mathcal{C}$ is a set of cycles on V, then for every $v \in V$, we denote by $\mathcal{C}_v$ the set of cycles in $\mathcal{C}$ passing through v. Analogously, we use $\mathbb{C}_v$ to denote the set of parity cycles in $\mathbb{C}$ passing through v.

Definition 15 (Well-contractible vertices with respect to cycles) Let $\mathcal{C}$ be any set of cycles on a vertex set V. We say that a vertex $v \in V$ is *well-contractible with respect to* $\mathcal{C}$ if $\mathcal{C}_v \neq \emptyset$ and there are two vertices $x, y \in V$ (x and y do not have to be distinct) such that $v \notin \{x, y\}$ and every cycle in $\mathcal{C}_v$ enters v through vertex x and leaves v through vertex y. (See Figures 4(a) and 5(a).)

Note that if a vertex has a self-loop, then it is not well-contractible. The above definition extends to parity cycles in a natural way.

Definition 16 (Well-contractible vertices with respect to parity cycles) Let $\mathbb{C}$ be any set of parity cycles on a vertex set V. Let $\mathcal{C}$ be the set of cycles obtained by removing weights and labels from each cycle in $\mathbb{C}$. A vertex $v \in V$ is *well-contractible with respect to* $\mathbb{C}$ if v is well-contractible with respect to $\mathcal{C}$.

We now describe cycle contractions.

Definition 17 (Cycle contraction for a parity cycles) Let $\mathfrak{c} = \langle (v_0, \ldots, v_t), w, \ell \rangle$ be a parity cycle of length t, where $t \geq 2$. Let v be one of the vertices on $\mathfrak{c}$, and let x and y be the neighbors of v on $\mathfrak{c}$ (it may be the case that $x = y$). The *contraction of* $\mathfrak{c}$ *at* v, denoted $\mathfrak{c}|_v$, is the parity cycle obtained from $\mathfrak{c}$ as follows:

- Vertex v is removed from the cycle, i.e., the edges (x, v) and (v, y) are replaced with an edge (x, y) (see Figures 4 and 5).
- The weight of both x and y is increased by half of the weight of v in $\mathfrak{c}$ (if $x = y$, then the weight of x is simply increased by the weight of v).
- The label of the new edge is the parity of labels of the edges being removed, i.e., it is $\mathbf{0}$ if the edges had the same labels, and $\mathbf{1}$, otherwise.

Definition 18 (Cycle contraction for a set of parity cycles) Let $\mathbb{C}$ be a set of parity cycles and let v be well-contractible with respect to $\mathbb{C}$. The set $\mathbb{C}|_v = (\mathbb{C} \setminus \mathbb{C}_v) \cup \{\mathfrak{c}|_v : \mathfrak{c} \in \mathbb{C}_v\}$ is a *contraction of* $\mathbb{C}$ *at* v.

Let $Q = \{u_1, \ldots, u_k\}$ be a set of vertices such that *(i)* for each $u \in Q$, either $\mathbb{C}_u = \emptyset$ or u is well-contractible with respect to $\mathbb{C}$, and *(ii)* such that the subset of Q consisting of vertices well-contractible with respect to $\mathbb{C}$ forms an independent set in $\mathbb{G}(\mathbb{C})$. Let $\mathbb{C}_0^\star = \mathbb{C}$ and for each $i \in \{1, \ldots, k\}$, let

$$\mathbb{C}_i^\star = \begin{cases} \mathbb{C}_{i-1}^\star & \text{if } \mathbb{C}_{u_i} = \emptyset, \\ \mathbb{C}_{i-1}^\star\big|_{u_i} & \text{otherwise.} \end{cases}$$

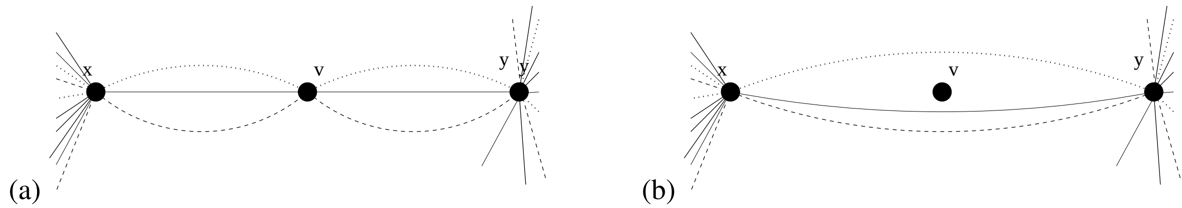


Figure 4: (a) A well-contractible vertex v (with $x \neq y$) with three cycles going through v, and (b) a contraction at v. If initially the weights of vertices x, v, y were $w(x), w(v), w(y)$, then after contraction at v, the weights changed to $w(x) + \frac{1}{2}w(v)$, 0, and $w(y) + \frac{1}{2}w(v)$, respectively.

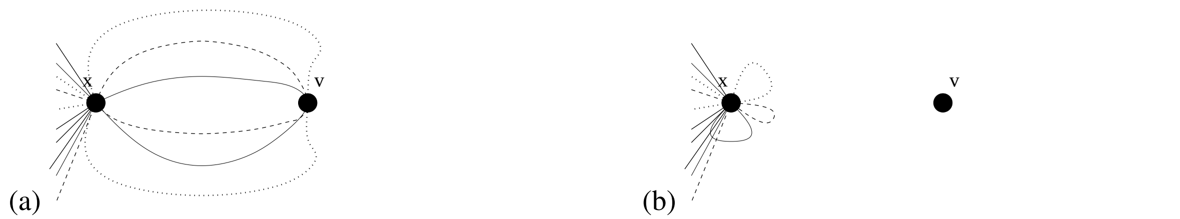


Figure 5: (a) A well-contractible vertex v (with $x = y$) with three cycles going through v, and (b) a contraction at v.

We call $\mathbb{C}_k^\star$ a *contraction of* $\mathbb{C}$ *at* Q and denote it by $\mathbb{C}|_Q$.

Intuitively, $\mathbb{C}|_Q$ is a parallel contraction at all vertices in Q. It is easy to show that the definition of $\mathbb{C}|_Q$ is correct, which involves showing that the expression $\mathbb{C}_{i-1}^\star\big|_{u_i}$ is used only when u_i is still well-contractible and that $\mathbb{C}_k^\star$ is independent of the order in which vertices of Q are considered.

The following observation is crucial for our second reduction. It follows from the fact that planarity is closed under contraction of edges and removal of edges (and vertices).

Observation 19 *If $\mathbb{C}$ is a collection of parity cycles such that $\mathbb{G}(\mathbb{C})$ is planar and $\mathbb{C}'$ is obtained by removals of parity cycles from $\mathbb{C}$ and cycle contractions at well-contractible vertices, then $\mathbb{G}(\mathbb{C}')$ is planar.*

5.3 Shortening cycles

The key idea used in our proof of Lemma 10 is to show that given a set of witness cycles $\mathbb{C}$, each of length at most k, we can replace it with a set of witness cycles $\mathbb{C}'$, each of length at most $k-1$, such that $\mathbb{G}(\mathbb{C}')$ is random walk invariant to $\mathbb{G}(\mathbb{C})$. We obtain $\mathbb{C}'$ in two steps. First, we delete some cycles from $\mathbb{C}$ and obtain a set $\mathbb{C}^\star$ for which there is a set Q of vertices that are well-contractible in $\mathbb{G}(\mathbb{C}^\star)$ and such that every cycle in $\mathbb{C}^\star$ either contains at least one vertex in Q or is a self-loop. In the second step, we exploit this and some further properties of $\mathbb{C}^\star$ to prove that by performing cycle contractions at the vertices of Q, we obtain the desired set $\mathbb{C}'$.

This idea is described in a generic form as a procedure **Cycle-Shortening** below. The exact choices of $\mathbb{C}^\star \subseteq \mathbb{C}$ and Q are explained later in the analysis.

Cycle-Shortening (set $\mathbb{C}$ of parity cycles of length not greater than k)

- Choose an appropriate subset $\mathbb{C}^\star$ of $\mathbb{C}$ such that every cycle in $\mathbb{C}^\star$ either is a self-loop or has a well-contractible vertex with respect to $\mathbb{C}^\star$.
- Perform cycle contraction at an appropriate independent set of well-contractible vertices Q to ensure that the length of each parity cycle in $\mathbb{C}^\star$ other than a self-loop reduces by at least 1.
- Return the resulting set $\mathbb{C}' = \mathbb{C}^\star|_Q$.

We now prove auxiliary lemmas that enable the construction of the set Q as well as the subset $\mathbb{C}^\star \subseteq \mathbb{C}$ of cycles that are contracted to obtain $\mathbb{C}'$ from $\mathbb{C}$. The graph properties that we explore in the next few lemmas are not related to edge parities or vertex weights. Because of that, we focus on proving them for the underlying unweighted and unlabeled cycles, and multigraphs induced by them.

Lemma 20 *Let $\mathcal{C}$ be a set of cycles on a vertex set V, with each cycle of length at least 2 and at most k. Let D be a constant and let Q be the set of vertices in V that have at most D distinct neighbors in $\mathcal{G}(\mathcal{C})$. If every cycle in $\mathcal{C}$ contains at least one vertex from Q, then there is a subset $\mathcal{C}^\star \subseteq \mathcal{C}$ such that $|\mathcal{C}^\star| \geq D^{-2k}|\mathcal{C}|$ and every cycle $\mathfrak{c} \in \mathcal{C}^\star$ has a well-contractible vertex with respect to $\mathcal{C}$.*

Proof. Initially, we set $\mathcal{C}^\star = \mathcal{C}$. We process vertices in Q one by one and prune $\mathcal{C}^\star$ in order to ensure that in the end, each vertex in Q is either well-contractible with respect to $\mathcal{C}^\star$ or is not contained in a cycle of $\mathcal{C}^\star$.

To measure the progress, we assign to each cycle $\mathfrak{c} \in \mathcal{C}$ its *age* $A(\mathfrak{c})$ and a corresponding *value* that equals $D^{2A(\mathfrak{c})}$. In the beginning, the age of each cycle is 0. Whenever we make a vertex v well-contractible, the age of every cycle containing this vertex increases by 1. Since the length of each cycle is bounded by k, the age of each cycle is at most k.

Consider a vertex $v \in Q$ that is contained in at least one cycle of the current set $\mathcal{C}^\star$. To make v well-contractible we proceed as follows. For every pair x and y of vertices adjacent to v in $\mathcal{G}(\mathcal{C}^\star)$ (including the case that $x = y$), compute the sum of the values of all cycles that go through v by first traversing x and then going through v directly to y. Let x and y be the pair that maximizes the sum of the values. We remove from $\mathcal{C}^\star$ all cycles going through v that do not use edges (x, v) and (v, y) to visit v. Observe that by definition, v then becomes a well-contractible vertex. If we apply this construction to all vertices in Q then each vertex in Q eventually either becomes well-contractible or is not contained in any remaining cycle. Therefore, if $\mathcal{C}^\star$ denotes the set of cycles at the end of the process, then since each cycle in $\mathcal{C}^\star$ contains at least one vertex from Q and since the length of the cycle is at least 2, it contains at least one contractible vertex.

It remains to show that $|\mathcal{C}^\star| \geq D^{-2k}|\mathcal{C}|$. Let us first consider the operation above for a single vertex v in Q that is contained in at least one cycle of the current $\mathcal{C}^\star$. Since v has degree at most D, the sum of the values of the cycles going through v left after the operation above is at least $1/D^2$ fraction of the sum of the values of all cycles going through v before the operation. However, after this operation, we also increase the age of each remaining cycle going through v by one, which implies that the sum of the values of these cycles increases by a factor of D^2. Hence, by the guaranteed ratio between the values in the removed and remaining cycles, after taking this increased age into account, we conclude that the operation above does not decrease the sum of the values of the cycles in the set.

With the invariant above, we can finalize our analysis. Initially, for the original set $\mathcal{C}$ of cycles, the sum of the values of the cycles equals $\sum_{\mathfrak{c} \in \mathcal{C}} D^{2A(\mathfrak{c})} = |\mathcal{C}|$. After applying the process above to all vertices in Q, the sum of the values does not decrease, i.e., $\sum_{\mathfrak{c}\in\mathcal{C}^\star} D^{2A(\mathfrak{c})} \geq |\mathcal{C}|$ for final values of $A(\mathfrak{c})$. Since $A(\mathfrak{c}) \leq k$, we conclude that $|\mathcal{C}| \leq \sum_{\mathfrak{c}\in\mathcal{C}^\star} D^{2A(\mathfrak{c})} \leq \sum_{\mathfrak{c}\in\mathcal{C}^\star} D^{2k} = D^{2k} \cdot |\mathcal{C}^\star|$. ■

To apply Lemma 20, we need to ensure that for sets $\mathcal{C}$ with planar $\mathcal{G}(\mathcal{C})$, the set Q of vertices with few distinct neighbors intersects with a constant fraction of the cycles in $\mathcal{C}$. We show that this property can indeed be achieved after a proper modification of the set of cycles.

Levels. Let $D \geq 5$ be an arbitrary constant. Let $\mathcal{C}$ be a set of cycles on a set V of vertices such that $\mathcal{C}$ contains no self-loops and $\mathcal{G}(\mathcal{C})$ is planar. We assign *levels* to vertices in V and cycles in $\mathcal{C}$ using the following algorithm:

Assigning-Levels (set $\mathcal{C}$ of cycles with no self-loops such that $\mathcal{G}(\mathcal{C})$ is planar)

- $t = 1$.
- Repeat until $\mathcal{C}$ is empty:
  - Let Q_t be the set of all vertices that have at most D distinct neighbors in $\mathcal{G}(\mathcal{C})$.
  - Let $\mathcal{C}_t$ be the set of all cycles in $\mathcal{C}$ that go through at least one vertex from Q_t.
  - Assign level t to every vertex in Q_t.
  - Assign level t to every cycle in $\mathcal{C}_t$.
  - $\mathcal{C} = \mathcal{C} \setminus \mathcal{C}_t$.
  - $t = t + 1$.

Since the graph $\mathcal{G}(\mathcal{C})$ is planar and has no self-loops, we can construct a *simple* planar graph by removing repeated edges in $\mathcal{G}(\mathcal{C})$. This operation preserves the number of distinct neighbors of each vertex. It now follows from Fact 3 that at least one vertex in $\mathcal{G}(\mathcal{C})$ has to have at most 5 distinct neighbors. This implies that the set Q_t is non-empty as long as $\mathcal{C}$ is nonempty, because $D \geq 5$.

Note that our construction ensures that while a vertex $v \in Q_i$ may belong to cycles in any $\mathcal{C}_j$ with $j \leq i$, the intersection of Q_i with the set of vertices present in cycles of $\bigcup_{j>i} \mathcal{C}_j$ is empty.

To relate the notion of the levels to Lemma 20, we notice that if there is a level i for which Q_i intersects with a constant fraction of cycles in $\mathcal{C}$, then we can directly invoke Lemma 20 to find a large subset of cycles that can be contracted simultaneously. However, vertices from every level may have small intersection with $\mathcal{C}$. Therefore, we need to incorporate a more involved procedure to ensure that a large intersection is obtained.

Let T be the maximum level assigned to any vertex in the algorithm above. For any vertex $v \in V$, let $\mathcal{C}_{t,v}$ be the set of all cycles in $\mathcal{C}_t$ that contain v. We apply the following filtering procedure.

Cycle-Level-Filtering (set $\mathcal{C}$ of cycles of length at most k)

- **for** $t = T$ **downto** 1 **do**
  - **if** $|\mathcal{C}_t| < \frac{1}{k}|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}|$ **then** delete $\mathcal{C}_t$ from $\mathcal{C}$
    else delete from $\mathcal{C}$ all cycles in $\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}$

Our goal is to analyze the properties of the set $\mathcal{C}$ obtained at the end of cycle level filtering. Let $\mathcal{C}^0$ be the original set $\mathcal{C}$, i.e., the input for the cycle level filtering procedure. We prove the following key lemma.

Lemma 21 *At the end of* **Cycle-Level-Filtering***, the size of $\mathcal{C}$ is at least $\frac{1}{2k^2}$ times the size of the original set $\mathcal{C}^0$ of cycles. Furthermore, every cycle in the obtained set $\mathcal{C}$ of cycles contains at least one vertex that has at most D distinct neighbors in $\mathcal{G}(\mathcal{C})$.*

Proof. We use charging arguments. For every $\mathfrak{c} \in \mathcal{C}$, we introduce the *cost of* $\mathfrak{c}$, which is initially set to 1. The cost of a cycle $\mathfrak{c}$ can be modified only on the levels when one of its vertices appears. Every time a cycle $\mathfrak{c}$ appears in the term $\frac{1}{k}|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}|$, we increase the cost of $\mathfrak{c}$ by 1. Therefore, when we reach the level of $\mathfrak{c}$, its cost is at most k, because its length is bounded by k. When $\mathfrak{c} \in \mathcal{C}_t$, we increase its cost by k^2. Observe that this ensures that the cost of any cycle $\mathfrak{c}$ is never greater than $k + k^2$.

We prove that the following inequality holds after any loop in **Cycle-Level-Filtering**:

$$\sum_{\mathfrak{c}\in\mathcal{C}} \text{cost}(\mathfrak{c}) \quad \geq \quad |\mathcal{C}^0|. \tag{1}$$

For any cycle $\mathfrak{c}$ and any $t \in \{1,\ldots,T\}$, let $\text{cost}^{\langle t\rangle}(\mathfrak{c})$ denote the cost of $\mathfrak{c}$ after the iteration with value t and let $\mathcal{C}^{\langle t\rangle}$ denote the set $\mathcal{C}$ after the iteration with value t. To initialize, we set $\mathcal{C}^{\langle T+1\rangle}$ to be the initial $\mathcal{C} = \mathcal{C}^0$ and for all cycles $\mathfrak{c}$, we set $\text{cost}^{\langle T+1\rangle}(\mathfrak{c}) = \text{cost}(\mathfrak{c})$. Furthermore, for all $t \in \{1,\ldots,T+1\}$, let $\mathcal{C}_j^{\langle t\rangle} = \mathcal{C}_j \cap \mathcal{C}^{\langle t\rangle}$ and $\mathcal{C}_{j,u}^{\langle t\rangle} = \mathcal{C}_{j,u} \cap \mathcal{C}^{\langle t\rangle}$.

Inequality (1) trivially holds in the beginning, because $\mathcal{C}^{\langle T+1\rangle} = \mathcal{C}^0$ and $\text{cost}^{\langle T+1\rangle}(\mathfrak{c}) = 1$ for every $\mathfrak{c}$.

We now prove that $\sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \text{cost}^{\langle t\rangle}(\mathfrak{c}) \geq \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t+1\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c})$ for all $t \in \{1,\ldots,T\}$, which implies inequality (1).

We consider two cases:

- If $|\mathcal{C}_t^{\langle t+1\rangle}| < \frac{1}{k}\left|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}\right|$, then $\mathcal{C}^{\langle t\rangle} = \mathcal{C}^{\langle t+1\rangle} \setminus \mathcal{C}_t^{\langle t+1\rangle}$, $\mathcal{C}_t^{\langle t\rangle} = \emptyset$, and for every $\mathfrak{c} \in \bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j\,v}^{\langle t+1\rangle}$, we have $\text{cost}^{\langle t\rangle}(\mathfrak{c}) = \text{cost}^{\langle t+1\rangle}(\mathfrak{c}) + 1$. Therefore,

$$\begin{aligned}
\sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \text{cost}^{\langle t\rangle}(\mathfrak{c}) \quad &= \quad \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c}) + \left|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}\right| \\
&> \quad \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c}) + k\cdot|\mathcal{C}_t^{\langle t+1\rangle}| \\
&\geq \quad \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c}) + \sum_{\mathfrak{c}\in\mathcal{C}_t^{\langle t+1\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c}) \\
&= \quad \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t+1\rangle}} \text{cost}^{\langle t+1\rangle}(\mathfrak{c}),
\end{aligned}$$

  where the first inequality follows from the assumption that $\left|\mathcal{C}_t^{\langle t+1\rangle}\right| < \frac{1}{k}\left|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}{}_v^{\langle t+1\rangle}\right|$ and the second one from the fact that $\text{cost}^{\langle t+1\rangle}(\mathfrak{c}) \leq k$ for every $\mathfrak{c} \in \bigcup_{j=1}^{t}\mathcal{C}_j^{\langle t+1\rangle}$.

- If $\left|\mathcal{C}_t^{\langle t+1\rangle}\right| \geq \frac{1}{k}\left|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}\right|$, then

$$\begin{aligned}
\sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \mathrm{cost}^{\langle t\rangle}(\mathfrak{c}) &= \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}) + k^2|\mathcal{C}_k^{\langle k+1\rangle}| \\
&\geq \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}) + \frac{k^2}{k}\left|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}\right| \\
&\geq \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t\rangle}} \mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}) + \sum_{\mathfrak{c}\in\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}} \mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}) \\
&= \sum_{\mathfrak{c}\in\mathcal{C}^{\langle t+1\rangle}} \mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}),
\end{aligned}$$

  where in the first inequality we used the assumption that $|\mathcal{C}_t^{\langle t+1\rangle}| \geq \frac{1}{k}|\bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}|$, and in the second inequality we used the fact that $\mathrm{cost}^{\langle t+1\rangle}(\mathfrak{c}) \leq k$ for every $\mathfrak{c} \in \bigcup_{v\in Q_t}\bigcup_{j=1}^{t-1}\mathcal{C}_{j,v}^{\langle t+1\rangle}$.

We can now apply inequality (1) to conclude the first claim. Since for any cycle $\mathfrak{c}$, $\mathrm{cost}(\mathfrak{c}) \leq k + k^2 \leq 2k^2$, it follows that at the end of filtering, $|\mathcal{C}^0| \leq \sum_{\mathfrak{c}\in\mathcal{C}} \mathrm{cost}(\mathfrak{c}) \leq |\mathcal{C}| \cdot 2k^2$. This is the desired lower bound on the size of the final $\mathcal{C}$: $|\mathcal{C}| \geq \frac{1}{2k^2}|\mathcal{C}^0|$.

We now prove the second claim, i.e., we show that every cycle remaining in $\mathcal{C}$ has at least one vertex with at most D distinct neighbors in $\mathcal{G}(\mathcal{C})$. Our construction, together with the observation made earlier that $Q_i \cap \bigcup_{j>i}\mathcal{C}_j = \emptyset$, ensures that for every t, at the end of the execution of the loop with value t, either $\mathcal{C}_t = \emptyset$ or $\bigcup_{v\in Q_t}\bigcup_{j\neq t}\mathcal{C}_{j,v} = \emptyset$. Let us consider any t with $\mathcal{C}_t \neq \emptyset$ (there must be such a t since $\mathcal{G}(\mathcal{C})$ is not empty). Then since $\bigcup_{v\in Q_t}\bigcup_{j\neq t}\mathcal{C}_{j,v}$ is empty, for every vertex that is contained in a cycle from $\mathcal{C}_t$, all its neighbors in $\mathcal{G}(\mathcal{C})$ are also vertices that are contained in a cycle from $\mathcal{C}_t$. Therefore, our construction of **Assigning-Levels** $(\mathcal{C})$ ensures that every cycle in $\mathcal{C}_t$ has a vertex from Q_t, and every vertex from Q_t has at most D distinct neighbors in the final $\mathcal{G}(\mathcal{C})$. ■

Equipped with Lemmas 20 and 21, we can proceed to the proof of the next lemma that takes care of all properties that are necessary for our cycle shortening. We introduce one more definition that is useful in situations when we obtain edges of different parities between a pair of vertices. We ensure that for the set of cycles that we obtain, for each pair of connected vertices, either all edges between them have the same parity or the numbers of edges with different parities are within a constant factor from each other.

Definition 22 A pair $\langle x, y\rangle$ of vertices is called *τ-parity balanced for* $\mathbb{C}$, where $\tau \in (0,1]$, if either all parallel edges (x,y) in $\mathbb{G}(\mathbb{C})$ have the same parity (i.e., all are odd, or all are even), or the ratio between the number of *odd* parallel edges (x,y) and the number of *even* parallel edges (x,y) in $\mathbb{G}(\mathbb{C})$ lies in $[\tau, 1/\tau]$.

We now prove the main technical lemma that states all desired properties of the set Q consisting of well-contractible vertices. First of all, we require that every vertex from Q be a well-contractible vertex with respect to a pruned set of parity cycles. Second, we require that every parity cycle that is not a self-loop has a well-contractible vertex. Both properties ensure that we reduce the length of every parity cycle by at least 1. Furthermore, we need to make sure that a random walk behaves similarly in both graphs. To this end, we first need to make sure that the distribution of the starting vertices is similar in the sense that the weight of a vertex in the contracted graph is similar to the sum of weights of the corresponding vertices of the original graph (which is enforced by the definition of vertex weights) and that the overall weight does not decrease too much. The latter property is ensured by the fact that the size of the new set of parity cycles is at least a constant fraction of the original set. Finally, we also want to make sure that moving to neighboring vertices has a comparable probability distribution. For this purpose, we make sure that all vertices of the parity cycles we keep have a degree that is at most a constant factor smaller than in the original graph. All these properties are stated formally in the following lemma.

Lemma 23 *Let $\mathbb{C}$ be a set of witness cycles on a vertex set V with planar $\mathbb{G}(\mathbb{C})$ and with each cycle in $\mathbb{C}$ of length at most k. Then there are positive τ, ϑ, and σ depending only on k, a subset $\mathbb{C}^\star \subseteq \mathbb{C}$, and an independent set $Q \subseteq V$ in $\mathbb{G}(\mathbb{C}^\star)$ such that*

- *every vertex from Q is well-contractible with respect to $\mathbb{C}^\star$,*
- *every cycle $\mathfrak{c} \in \mathbb{C}^\star$ that is not a self-loop has a well-contractible vertex from Q,*
- *all pairs $\langle x, y\rangle$, where x and y are vertices of $\mathbb{G}(\mathbb{C}^\star)$, are τ-parity balanced for $\mathbb{C}^\star$,*
- *every vertex in $\mathbb{G}(\mathbb{C}^\star)$ has degree at least ϑ times its degree in $\mathbb{G}(\mathbb{C})$, and*
- *$|\mathbb{C}^\star| \geq \sigma|\mathbb{C}|$.*

Proof. Let $\mathbb{C}_0^\triangle$ be the set of self-loops in $\mathbb{C}$. Let $\mathbb{C}_1^\triangle = \mathbb{C} \setminus \mathbb{C}_0^\triangle$ be the subset of witness cycles in $\mathbb{C}$ of length at least 2. We first apply procedures **Assigning-Levels** and **Cycle-Level-Filtering** to $\mathbb{C}_1^\triangle$ with $D = 5$, ignoring the fact that cycles in $\mathbb{C}_1^\triangle$ have labels and weights. Let $\mathbb{C}_2^\triangle \subseteq \mathbb{C}_1^\triangle$ be the resulting set of witness cycles. By Lemma 21, $\left|\mathbb{C}_2^\triangle\right| \geq \left|\mathbb{C}_1^\triangle\right|/(2k^2)$. Ignoring weights and labels again, we now apply Lemma 20 to $\mathbb{C}_2^\triangle$, obtaining a set $\mathbb{C}_3^\triangle$ such that each cycle in $\mathbb{C}_3^\triangle$ has a well-contractible vertex with respect to $\mathbb{C}_3^\triangle$. Furthermore, it holds that $\left|\mathbb{C}_3^\triangle\right| \geq D^{-2k} \cdot \left|\mathbb{C}_2^\triangle\right| \geq \left|\mathbb{C}_1^\triangle\right| / \left(2k^2 D^{2k}\right)$.

Let $\mathbb{C}'$ be the larger of $\mathbb{C}_3^\triangle$ and $\mathbb{C}_0^\triangle$. $\mathbb{C}'$ is either a set of witness cycles, each having a well-contractible vertex with respect to $\mathbb{C}'$, or a set of self-loops. Since $\left|\mathbb{C}_0^\triangle\right| + \left|\mathbb{C}_1^\triangle\right| = |\mathbb{C}|$, it is easy to show that $|\mathbb{C}'| \geq |\mathbb{C}| / \left(4k^2 D^{2k}\right)$.

In the contraction procedure, we do not want to use all well-contractible vertices, but only an independent subset of them, so we can perform all contractions independently.[4] For the rest of the proof, let $\widetilde{Q}$ be a *maximal* independent subset of well-contractible vertices in $\mathbb{G}(\mathbb{C}')$ with respect to $\mathbb{C}'$. Let $\mathfrak{c}$ be any cycle from $\mathbb{C}'$ that is not a self-loop and let v be a well-contractible vertex on $\mathfrak{c}$. We want to show that at least one vertex in $\widetilde{Q}$ lies on $\mathfrak{c}$. If $v \in \widetilde{Q}$, we are done. Otherwise, since $\widetilde{Q}$ is maximal, a neighbor v' of v belongs to $\widetilde{Q}$. By the definition of a well-contractible vertex, since $\mathfrak{c}$ passes through v, it has to pass also through v'. Therefore, every cycle in $\mathbb{C}'$ that is not a self-loop has a well-contractible vertex in $\widetilde{Q}$ with respect to $\mathbb{C}'$.

We now introduce additional notation. Let $\tau = \vartheta = \frac{1}{24k^3D^{2k}}$. In the following, $\widetilde{\mathbb{C}}$ denotes an arbitrary subset of $\mathbb{C}'$. Let $\text{ODD}_{\widetilde{\mathbb{C}}}(x, y)$ be the number of edges (x, y) with parity $\mathbf{1}$ in $\mathbb{G}(\widetilde{\mathbb{C}})$. Similarly, let $\text{EVEN}_{\widetilde{\mathbb{C}}}(x, y)$ be the number of edges (x, y) with parity $\mathbf{0}$ in $\mathbb{G}(\widetilde{\mathbb{C}})$. Putting this notation to use, $\langle x, y\rangle$ is *τ-parity balanced for* $\widetilde{\mathbb{C}}$ if and only if $\text{ODD}_{\widetilde{\mathbb{C}}}(x, y) = 0$, or $\text{EVEN}_{\widetilde{\mathbb{C}}}(x, y) = 0$, or if $\tau \leq \frac{\text{ODD}_{\widetilde{\mathbb{C}}}(x,y)}{\text{EVEN}_{\widetilde{\mathbb{C}}}(x,y)} \leq \frac{1}{\tau}$. For the original subset $\mathbb{C}'$, we may omit the subscript, i.e., $\text{ODD}(x, y) = \text{ODD}_{\mathbb{C}'}(x, y)$ and $\text{EVEN}(x, y) = \text{EVEN}_{\mathbb{C}'}(x, y)$.

[4] As an example, consider a chain $x_1, \ldots, x_i$ of distinct adjacent contractible vertices. In this case, we could consider only every other vertex from the chain, say, $x_1, x_3, \ldots$, to make sure that the contractions can be performed independently.

For each vertex v of $\mathbb{G}(\mathbb{C})$, we write $\deg(v)$ to denote the degree of v in $\mathbb{G}(\mathbb{C})$ (self-loops contribute 2 to the count). To define $\mathbb{C}^\star$, we consider a process that gradually deletes some cycles from $\mathbb{C}'$. At any given step of this process, we use the term *current degree* of a vertex v to refer to the degree of v in the graph $\mathbb{G}(\mathbb{C}')$ induced by the *current* set of cycles $\mathbb{C}'$. The process is as follows.

Repeat as long as possible:

- If there is a vertex $v \in V$ with the current degree in $\mathbb{G}(\mathbb{C}')$ at most $\vartheta \cdot \deg(v)$, then
  - delete from $\mathbb{C}'$ all parity cycles passing through v in $\mathbb{C}'$.
- If there is a pair $\langle x, y\rangle \in V^2$ that is not τ-parity-balanced for the current $\mathbb{C}'$, then
  - If $\text{ODD}_{\mathbb{C}'}(x,y) < \text{EVEN}_{\mathbb{C}'}(x,y)$, then
    delete from $\mathbb{C}'$ all cycles with an odd-parity edge between x and y
    going through $\text{ODD}_{\mathbb{C}'}(x,y)$ in $\mathbb{C}'$;
  - Otherwise, delete from $\mathbb{C}'$ all cycles with an even-parity edge between x and y

We define $\mathbb{C}^\star$ to be the final $\mathbb{C}'$ produced by the above process. Moreover, we define Q to be the subset of $\widetilde{Q}$ consisting of vertices that are well-contractible with respect to $\mathbb{C}^\star$. It consists of vertices in $\widetilde{Q}$ that did not become isolated in the process of pruning $\mathbb{C}'$.

To estimate the total number of cycles deleted, we charge to v the number of deleted cycles in each call to the first operation above and we charge to the pair $\langle x, y\rangle$ the number of deleted cycles in each call to the second operation. Observe that each vertex $v \in V$ has been processed (charged to) above at most once. Indeed, once a vertex v has been used, all incident cycles are deleted, the vertex disappears from $\mathbb{G}(\mathbb{C}')$, and hence it cannot be used again. Therefore, at most $\vartheta \cdot \deg(v)$ cycles from $\mathbb{C}'$ can be charged to any single vertex. Similarly, we observe that each pair of vertices $\langle x, y\rangle$ has been processed (charged to) by the process above at most once. This is because if $\langle x, y\rangle$ has been processed, then either $\text{ODD}_{\mathbb{C}'}(x,y)$ or $\text{EVEN}_{\mathbb{C}'}(x,y)$ becomes 0, and hence $\langle x, y\rangle$ has to remain τ-parity-balanced for all future $\mathbb{C}'$. Therefore, at most $\tau \cdot \max\{\text{ODD}(x,y), \text{EVEN}(x,y)\} \le \tau \cdot (\text{ODD}(x,y) + \text{EVEN}(x,y))$ cycles from $\mathbb{C}'$ can be charged to any pair $\langle x, y\rangle$.

The two bounds above imply that the total number of cycles removed from $\mathbb{C}'$ is at most $\sum_{v\in V} \vartheta \cdot \deg(v) + \sum_{x,y\in V} \tau \cdot (\text{ODD}(x,y) + \text{EVEN}(x,y))$. Since each cycle in $\mathbb{C}$ is of length at most k, $\mathbb{G}(\mathbb{C})$ has at most $k|\mathbb{C}|$ edges. Because of that, $\sum_{v\in V}\deg(v) \le 2k|\mathbb{C}|$ and $\sum_{x,y\in V}(\text{ODD}(x,y) + \text{EVEN}(x,y)) \le k|\mathbb{C}|$. Therefore, the total number of cycles removed is at most $(2\vartheta + \tau)k|\mathbb{C}|$. Thus, the total number of cycles in $\mathbb{C}^\star$ is lower bounded by $|\mathbb{C}'| - (2\vartheta+\tau)k|\mathbb{C}| \ge \frac{1}{4k^2D^{2k}}|\mathbb{C}| - \frac{1}{8k^2D^{2k}}|\mathbb{C}| = \frac{1}{8k^2D^{2k}}|\mathbb{C}|$.

In summary, we have constructed a subset $\mathbb{C}^\star$ of $\mathbb{C}$ and a set $Q \subseteq V$ that satisfy the following properties:

- Q is an independent set in $\mathbb{G}(\mathbb{C}^\star)$ (because a superset of Q was an independent set in a parity graph that contained $\mathbb{G}(\mathbb{C}^\star)$).
- Every vertex from Q is a well-contractible vertex with respect to $\mathbb{C}^\star$.
- Every cycle $\mathfrak{c} \in \mathbb{C}^\star$ (since $\mathfrak{c}$ is also a cycle in $\mathbb{C}'$) contains a well-contractible vertex from Q or is a self-loop.
- All pairs $\langle x, y\rangle$, where x and y are vertices in $\mathbb{G}(\mathbb{C}^\star)$, are $\frac{1}{24k^3D^{2k}}$-parity balanced with respect to $\mathbb{C}^\star$.

- Every vertex in $\mathbb{G}(\mathbb{C}^\star)$ has degree at least $\frac{1}{24k^3D^{2k}}$ times its degree in $\mathbb{G}(\mathbb{C})$.
- $|\mathbb{C}^\star| \ge \frac{1}{8k^2D^{2k}}|\mathbb{C}|$.

This completes the proof of the lemma. ■

5.4 Putting things together: proving Lemma 10

We now have all tools necessary to prove Lemma 10, which is the last missing link in the proof of Theorem 4.

Proof of Lemma 10. The set $\mathbb{C}'$ of parity cycles is a result of applying procedure **Cycle-Shortening** (see p. 15) with $\mathbb{C}^\star$ and Q given by Lemma 23, that is, $\mathbb{C}' = \mathbb{C}^\star|_Q$. Observe first that contraction preserves the weight and parity of a parity cycle, so all parity cycles in $\mathbb{C}'$ are (α, β)-witness cycles. Moreover, the first three desired properties of $\mathbb{C}' = \mathbb{C}^\star|_Q$ follow directly from Lemma 23. It remains to prove that $\mathbb{G}(\mathbb{C}')$ is $(3, 3^{k-1}, \xi\alpha/\beta)$-random walk invariant with respect to $\mathbb{G}(\mathbb{C})$ for some positive $\xi = \xi(k)$. Let τ, ϑ, and σ be the positive constants from Lemma 23, which depend only on k. We prove that the statement holds for $\xi = \xi(k) = \vartheta\sigma \cdot (\vartheta^2\tau/(2(1+\tau)))^t$, where $t = 3^{k-1}$ is the random-walk length in $\mathbb{G}(\mathbb{C}')$.

Let us fix an arbitrary walk $\langle x_0, x_1, \dots, x_t\rangle$ in $\mathbb{G}(\mathbb{C}')$ with a sequence of parities $\langle \chi_0, \dots, \chi_{t-1}\rangle$, where $\chi_i \in \{\mathbf{0}, \mathbf{1}\}$ is the parity of the edge (x_i, x_{i+1}) in the walk. Our goal is to compare the probability that **RWPG** chooses the walk $\langle x_0, x_1, \dots, x_t\rangle$ with the prescribed parities of edges on $\mathbb{G}(\mathbb{C}')$ with the probability that **RWPG** chooses a walk with a prefix of the form $\langle ?, x_0, ?, x_1, ?, x_2, \dots, x_{t-1}, ?, x_t\rangle$ on $\mathbb{G}(\mathbb{C})$, where each question mark denotes either a vertex from Q or no vertex at all.[5] Additionally, we require that for any $0 \le i < t$, the parity of the selected path $\langle x_i, ?, x_{i+1}\rangle$ is χ_i. Such a prefix has length at most $2t + 1 \le 3t$ and therefore it may occur as a prefix of a $3t$-step random walk. Note the following key property of our construction: if the walk $\langle x_0, x_1, \dots, x_t\rangle$ with the prescribed parities of edges discovers an odd parity cycle in $\mathbb{G}(\mathbb{C}')$, then so does $\langle ?, x_0, ?, x_1, ?, x_2, \dots, x_{t-1}, ?, x_t\rangle$ in $\mathbb{G}(\mathbb{C})$.

Let us introduce more notation. For any vertex $u \in V$, let Ψ_u be the set of well-contractible vertices in Q that are contracted into u during the cycle contractions that form $\mathbb{C}'$. If one of contracted paths is of the form $\langle u, v, u\rangle$ as in Figure 5, then we assume that Ψ_u is a multiset and contains two copies of v. We also define Υ_u to be the version of Ψ_u with no vertex multiplicities.

Let $w(u)$, $w^\star(u)$, and $w'(u)$ denote the weight of a vertex $u \in V$ in $\mathbb{G}(\mathbb{C})$, $\mathbb{G}(\mathbb{C}^\star)$, and $\mathbb{G}(\mathbb{C}')$, respectively, provided u belongs to a given graph. Then, for every vertex u in $\mathbb{G}(\mathbb{C}')$ we have:

$$w'(u) = w^\star(u) + \tfrac{1}{2} \sum_{v \in \Psi_u} w^\star(v). \tag{2}$$

Let W and W' be the total weight of vertices in $\mathbb{G}(\mathbb{C})$ and $\mathbb{G}(\mathbb{C}')$, respectively. Note that the total weight of vertices in $\mathbb{G}(\mathbb{C}^\star)$ is the same as in $\mathbb{G}(\mathbb{C}')$, because contractions preserve the total weight. Moreover, $W' \ge \sigma W \cdot (\alpha/\beta)$, because the total number of cycles in $\mathbb{C}'$ is at least $\sigma|\mathbb{C}|$ and the average total weight of a cycle cannot decrease by a factor greater than β/α.

Furthermore, let d_u, $d_u^\star$, and d_u' be the degree of a vertex u in $\mathbb{G}(\mathbb{C})$, $\mathbb{G}(\mathbb{C}^\star)$, and $\mathbb{G}(\mathbb{C}')$, respectively, provided u belongs to a given graph. We assume that a self-loop contributes 2 to the degree of the vertex it is incident to. Observe that by Lemma 23, for every u in the vertex set of $\mathbb{G}(\mathbb{C}^\star)$, $d_u^\star \ge \vartheta \cdot d_u$. Furthermore, $d_u' = d_u^\star$ for every vertex u in $\mathbb{G}(\mathbb{C}')$ (i.e., for every vertex that does not become isolated as a result of cycle contractions).

[5]For example, $\langle ?, x_0, ?, x_1\rangle$ denotes one of the following: a path $\langle x_0, x_1\rangle$, or a path $\langle v, x_0, x_1\rangle$ with an arbitrary vertex $v \in Q$, or a path $\langle x_0, u, x_1\rangle$ with an arbitrary vertex $u \in Q$, or a path $\langle v, x_0, u, x_1\rangle$ with an arbitrary pair of vertices $v, u \in Q$.

Starting the walk. A necessary condition for **RWPG** on $\mathbb{G}(\mathbb{C}')$ to visit $\langle x_0, x_1, \ldots, x_t\rangle$ in a random walk is that x_0 is chosen as the starting vertex. This happens with probability $w'(x_0)/W'$. The corresponding random walk on $\mathbb{G}(\mathbb{C})$ is initiated correctly when one of the following disjoint events occurs:

(i) x_0 is chosen as the starting vertex by **RWPG** on $\mathbb{G}(\mathbb{C})$,

(ii) a vertex from Ψ_{x_0} is chosen as the starting vertex of the random walk by **RWPG** on $\mathbb{G}(\mathbb{C})$ and then the random walk moves to x_0 in a single step.

The first event happens with probability $w(x_0)/W \geq w^\star(x_0)/W$. The second event happens with probability

$$P = \sum_{u\in\Upsilon_{x_0}} \frac{w(u)}{W} \cdot \mathbf{Pr}[\text{single step of random walk on } \mathbb{G}(\mathbb{C}) \text{ moves from } u \text{ to } x_0] \ .$$

Note first that $w(u) \geq w^\star(u)$ for any $u \in \Upsilon_{x_0}$. Then a single step of a random walk moves from u to x_0 with probability $d_u^\star/d_u \geq \vartheta$ if u gets completely contracted into x_0 (as in Figure 5) or with probability $(d_u^\star/2)/d_u \geq \vartheta/2$ if the weight of u is split between two different vertices (as in Figure 4). In the former case, u appears twice in Ψ_{x_0}. Using this information, we can write

$$P \geq \sum_{u\in\Psi_{x_0}} \frac{w^\star(u)}{W} \cdot \frac{\vartheta}{2} \ .$$

The sum of probabilities of both events is at least

$$\frac{w^\star(x_0)}{W} + \sum_{u\in\Psi_{x_0}} \frac{w^\star(u)}{W} \cdot \frac{\vartheta}{2} \geq \frac{\vartheta}{W} \cdot \left(w^\star(x_0) + \sum_{u\in\Psi_{x_0}} w^\star(u) \right) \geq \vartheta\sigma \cdot (\alpha/\beta) \cdot \frac{w'(x_0)}{W'} \ ,$$

i.e., the probability that the random walk on $\mathbb{G}(\mathbb{C})$ gets initiated in a manner corresponding to the walk on $\mathbb{G}(\mathbb{C}')$ decreases by a multiplicative factor of at most $\vartheta\sigma \cdot \alpha/\beta$, compared to the walk on $\mathbb{G}(\mathbb{C}')$.

Continuing the walk. Next, we assume that the random walk reached a vertex x_i in both **RWPG** on $\mathbb{G}(\mathbb{C}')$ and **RWPG** on $\mathbb{G}(\mathbb{C})$, and we compare the probability that **RWPG** on $\mathbb{G}(\mathbb{C}')$ takes an edge (x_i, x_{i+1}) with a given label $\chi_i \in \{\mathbf{0}, \mathbf{1}\}$ to the probability that **RWPG** on $\mathbb{G}(\mathbb{C})$ takes a path $\langle x_i, ?, x_{i+1}\rangle$ with the same parity χ_i.

Let k_i be the number of edges of parity χ_i that connect x_i and x_{i+1} in $\mathbb{G}(\mathbb{C}')$ (as always, we count self-loops as two edges, because, intuitively, they can be followed in two different directions). The probability that **RWPG** on $\mathbb{G}(\mathbb{C}')$ follows one of them is k_i/d'_{x_i}.

We now want to bound from below the probability that **RWPG** on $\mathbb{G}(\mathbb{C})$ follows a path $\langle x_i, ?, x_{i+1}\rangle$. Each of the k_i direct parity-χ_i edges in $\mathbb{G}(\mathbb{C}')$ either already existed in $\mathbb{G}(\mathbb{C})$ (let $k_i^\star$ denote the number of edges of this kind) or is a result of contracting some path $\langle x_i, v, x_{i+1}\rangle$. For every $v \in Q$, let $k_{i,v}^\star$ be the number of parity-χ_i edges that are a result of contracting paths $\langle x_i, v, x_{i+1}\rangle$ (if $x_i = x_{i+1}$, each path is accounted for twice). Clearly, $k_i^\star + \sum_{v\in Q} k_{i,v}^\star = k_i$.

We first bound from below the probability that **RWPG** on $\mathbb{G}(\mathbb{C}^\star)$ follows a path $\langle x_i, ?, x_{i+1}\rangle$. This event occurs if one of the following disjoint events happens:

(i) the random walk selects one of the $k_i^\star$ direct edges,

(ii) for some $v \in Q$, the random walk selects one of the $k^\star_{i,v}$ edges from x_i to v that lay on a parity-χ_i path $\langle x_i, v, x_{i+1}\rangle$ that gets contracted into an edge (x_i, x_{i+1}) in $\mathbb{G}(\mathbb{C}')$ and then selects an arbitrary edge (v, x_{i+1}) that has complementary parity.

The probability of the first event is $k^\star_i/d^\star_{x_i}$. For any $v \in Q$ with $k^\star_{i,v} > 0$, the probability of the second event is at least $k^\star_{i,v}/d^\star_{x_i} \cdot (1/2) \cdot \tau/(1+\tau)$. The first factor in the last expression is the probability of selecting one of the suitable edges to v. The second factor is a lower bound for the probability of selecting at v an edge leading to x_{i+1}, because v is well-contractible with respect to $\mathbb{C}^\star$ and at least half of all the edges incident to v lead to x_{i+1}. Finally, the last factor is a lower bound for the fraction of edges (v, x_{i+1}) that have the desired parity resulting in a path of parity χ_i. This follows from the fact that $\langle v, x_{i+1}\rangle$ is τ-parity balanced with respect to $\mathbb{C}^\star$ and has to have at least one edge of the desired parity, because the edge that was taken from x_i to v_i on some parity cycle has a continuing edge (v, x_{i+1}) on the same cycle that results in a path of parity χ_i. Summarizing, the probability of selecting a path $\langle x_i, ?, x_{i+1}\rangle$ of parity χ_i is at least

$$\frac{k^\star_i}{d^\star_{x_i}} + \sum_{v\in Q} \frac{k^\star_{i,v}}{d^\star_{x_i}} \cdot \frac{1}{2} \cdot \frac{\tau}{1+\tau} \enspace .$$

To express the probability of following such a path in $\mathbb{G}(\mathbb{C})$, observe that edges of $\mathbb{C}^\star$ constitute at least an ϑ fraction of edges incident to vertices present in $\mathbb{G}(\mathbb{C}^\star)$. Therefore, a random step in $\mathbb{G}(\mathbb{C})$ at a vertex present in $\mathbb{G}(\mathbb{C}^\star)$ follows an edge present in $\mathbb{G}(\mathbb{C}^\star)$ with probability at least ϑ. Summarizing the probability of following the path $\langle x_i, ?, x_{i+1}\rangle$ of parity χ_i in $\mathbb{G}(\mathbb{C})$ is at least

$$\vartheta \cdot \frac{k^\star_i}{d^\star_{x_i}} + \vartheta^2 \cdot \sum_{v\in Q} \frac{k^\star_{i,v}}{d^\star_{x_i}} \cdot \frac{1}{2} \cdot \frac{\tau}{1+\tau} \ge \frac{\vartheta^2\tau}{2(1+\tau)} \cdot \frac{k^\star_i + \sum_{v\in Q} k^\star_{i,v}}{d^\star_{x_i}} = \frac{\vartheta^2\tau}{2(1+\tau)} \cdot \frac{k_i}{d'_{x_i}} \enspace ,$$

which is smaller than the probability of the corresponding parity-χ_i step (x_i, x_{i+1}) from x_i in $\mathbb{G}(\mathbb{C}')$ by a factor of at most $\frac{\vartheta^2\tau}{2(1+\tau)}$.

Finishing the analysis. We have showed that for any random walk $\langle x_0, x_1, \ldots, x_t\rangle$ in $\mathbb{G}(\mathbb{C}')$ with prescribed parities, which occurs with probability δ, the probability that we take a length-$3t$ random walk with prefix $\langle ?, x_0, ?, x_1, ?, x_2, \ldots, x_{t-1}, ?, x_t\rangle$ and corresponding parities is at least

$$\vartheta\sigma \cdot \frac{\alpha}{\beta} \cdot \left(\frac{\vartheta^2\tau}{2(1+\tau)}\right)^t \cdot \delta \enspace .$$

In the worst case, we may lose a factor of $\vartheta\sigma \cdot \frac{\alpha}{\beta}$ in the initialization of the walk and a factor of $\vartheta^2\tau/(2(1+\tau))$ trying to mimic each transition between vertices. This means that an odd parity cycle is detected in $\mathbb{G}(\mathbb{C})$ using a walk of length 3^k with probability at least $\xi \cdot \alpha/\beta$ time the probability of detecting such a cycle in $\mathbb{G}(\mathbb{C}')$ with a random walk of length $t = 3^{k-1}$, where $\xi = \vartheta\sigma \cdot (\vartheta^2\tau/(2(1+\tau)))^t$ depends only on k. This completes the proof of Lemma 10. ■

6 Extending the analysis to minor-free graphs

While throughout the paper we focus on testing bipartiteness of planar graphs, our techniques can easily be extended to any class of minor-free graphs. Recall that a graph H is called a *minor* of a graph G if H can be obtained from G via a sequence of vertex and edge deletions and edge contractions. For any graph H, a graph G is called *H-minor-free* if H is not a minor of G. (For example, by Kuratowski's Theorem, a graph is planar if and only if it is $K_{3,3}$-minor-free and K_5-minor-free.)

Let us fix a graph H and consider the input graph G to be an H-minor-free graph. We now argue now that entire analysis presented in the previous sections easily extends to testing bipartiteness of G. The key observation is that our analysis in Sections 2–5 relies only on the following properties of planar graphs:

(i) the number of edges in a planar graph is $O(n)$, where n is the number of vertices (Fact 3),

(ii) every minor of a planar graph is planar (Fact 2),

(iii) a direct implication of the Klein-Plotkin-Rao theorem for planar graphs (Lemma 12).

The first two properties hold for any class of H-minor-free-graphs (that is, the second property would be that every minor of an H-minor-free-graph is H-minor-free). Since the Klein-Plotkin-Rao theorem holds for any minor-free graph as well, so does a version of Lemma 12 with a slightly different constant hidden by the big O notation. Therefore, we can proceed with nearly identical analysis for H-minor-free graphs and arrive at the following version of Theorem 4.

Theorem 24 *Let H be a fixed graph. There are positive functions f and g such that for any H-minor-free-graph G:*

- *if G is bipartite, then* **Random-Bipartiteness-Exploration**(G, ε) *accepts G, and*
- *if G is ε-far from bipartite, then* **Random-Bipartiteness-Exploration**(G, ε) *rejects G with probability at least* 0.99*.*

7 Conclusions

In this paper we proved that bipartiteness is testable with constant query complexity for arbitrary planar graphs. Our result was proven via a new type of analysis of random walks in planar graphs. Our analysis easily carries over to classes of graphs defined by arbitrary fixed forbidden minors.

This is merely the first step that poses the following main question:

What graph properties can be tested with constant query complexity in minor-free graphs?

References

[1] N. Alon, E. Fischer, I. Newman, and A. Shapira. A combinatorial characterization of the testable graph properties: it's all about regularity. *SIAM Journal on Computing*, 39:143–167, 2009.

[2] N. Alon and M. Krivelevich. Testing k-colorability. *SIAM Journal on Discrete Mathematics*, 15(2):211-227, 2002.

[3] I. Benjamini, O. Schramm, and A. Shapira. Every minor-closed property of sparse graphs is testable. *Advances in Mathematics*, 223:2200–2218, 2010.

[4] A. Bogdanov, K. Obata, and L. Trevisan. A lower bound for testing 3-colorability in bounded-degree graphs. *In Proceedings of the 43rd IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 93–102, 2002.

[5] A. Czumaj, A. Shapira, and C. Sohler. Testing hereditary properties of nonexpanding bounded-degree graphs. *SIAM Journal on Computing*, 38(6): 2499–2510, April 2009.

[6] A. Czumaj and C. Sohler. Testing expansion in bounded-degree graphs. *In Proceedings of the 48th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 570–578, 2007.

[7] O. Goldreich, editor. *Property Testing: Current Research and Surveys*. Lecture Notes in Computer Science 6390, Springer Verlag, Berlin, Heidelberg, December 2010.

[8] O. Goldreich, S. Goldwasser, and D. Ron. Property testing and its connection to learning and approximation. *Journal of the ACM*, 45(4): 653–750, July 1998.

[9] O. Goldreich and D. Ron. Property testing in bounded degree graphs. *Algorithmica*, 32(2): 302–343, 2002.

[10] O. Goldreich and D. Ron. A sublinear bipartiteness tester for bounded degree graphs. *Combinatorica*, 19(3):335–373, 1999.

[11] O. Goldreich and D. Ron. On testing expansion in bounded-degree graphs. *Electronic Colloquium on Computational Complexity (ECCC)*, Report No. 7, 2000.

[12] A. Hassidim, J. A. Kelner, H. N. Nguyen, and K. Onak. Local graph partitions for approximation and testing. *In Proceedings of the 50th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 22–31, 2009.

[13] T. Kaufman, M. Krivelevich, and D. Ron. Tight bounds for testing bipartiteness in general graphs. *SIAM Journal on Computing*, 33(6): 1441–1483, September 2004.

[14] P. Klein, S. Plotkin, and S. Rao. Excluded minors, network decomposition, and multicommodity flow. *In Proceedings of the 25th Annual ACM Symposium on Theory of Computing (STOC)*, pp. 682–690, 1993.

[15] S. Marko and D. Ron. Approximating the distance to properties in bounded-degree and general sparse graphs. *ACM Transactions on Algorithms*, 5(2), Article No. 22, March 2009.

[16] I. Newman and C. Sohler. Every property of hyperfinite graphs is testable. *In Proceedings of the 43rd Annual ACM Symposium on Theory of Computing (STOC)*, pp. 675–684, 2011.

[17] H. N. Nguyen and K. Onak. Constant-time approximation algorithms via local improvements. *In Proceedings of the 49th IEEE Symposium on Foundations of Computer Science (FOCS)*, pp. 327–336, 2008.