Algorithmica
https://doi.org/10.1007/s00453-024-01238-z

Approximate and Randomized Algorithms for Computing a Second Hamiltonian Cycle

Argyrios Deligkas[1] · **George B. Mertzios**[2] · **Paul G. Spirakis**[3] · **Viktor Zamaraev**[3]

Received: 18 April 2022 / Accepted: 24 April 2024

Abstract

In this paper we consider the following problem: Given a Hamiltonian graph G, and a Hamiltonian cycle C of G, can we compute a second Hamiltonian cycle $C' \neq C$ of G, and if yes, how quickly? If the input graph G satisfies certain conditions (e.g. if every vertex of G is odd, or if the minimum degree is large enough), it is known that such a second Hamiltonian cycle always exists. Despite substantial efforts, no subexponential-time algorithm is known for this problem. In this paper we relax the problem of computing a second Hamiltonian cycle in two ways. First, we consider *approximating* the length of a second longest cycle on n-vertex graphs with minimum degree δ and maximum degree Δ. We provide a linear-time algorithm for computing a cycle $C' \neq C$ of length at least $n - 4\alpha(\sqrt{n} + 2\alpha) + 8$, where $\alpha = \frac{\Delta-2}{\delta-2}$. This results provides a constructive proof of a recent result by Girão, Kittipassorn, and Narayanan in the regime of $\frac{\Delta}{\delta} = o(\sqrt{n})$. Our second relaxation of the problem is probabilistic. We propose a randomized algorithm which computes a second Hamiltonian cycle *with high probability*, given that the input graph G has a large enough minimum degree.

Supported by the NeST initiative of the EEE/CS School of the University of Liverpool and by the EPSRC grants EP/P02002X/1 and EP/P020372/1. The results of Sect. 2 previously appeared in an extended abstract in the proceedings of the 45th International Symposium on Mathematical Foundations of Computer Science, MFCS 2020 [12].

✉ George B. Mertzios
george.mertzios@durham.ac.uk

Argyrios Deligkas
argyrios.deligkas@rhul.ac.uk

Paul G. Spirakis
p.spirakis@liverpool.ac.uk

Viktor Zamaraev
viktor.zamaraev@liverpool.ac.uk

[1] Department of Computer Science, Royal Holloway University of London, Egham, UK

[2] Department of Computer Science, Durham University, Durham, UK

[3] Department of Computer Science, University of Liverpool, Liverpool, UK

Published online: 12 June 2024

More specifically, we prove that for every $0 < p \leq 0.02$, if the minimum degree of G is at least $\frac{8}{p}\log\sqrt{8}n + 4$, then a second Hamiltonian cycle can be computed with probability at least $1-\frac{1}{n}\left(\frac{50}{p^4}+1\right)$ in $poly(n)\cdot 2^{4pn}$ time. This result implies that, when the minimum degree δ is sufficiently large, we can compute with high probability a second Hamiltonian cycle faster than any known deterministic algorithm. In particular, when $\delta = \omega(\log n)$, our probabilistic algorithm works in $2^{o(n)}$ time.

Keywords Hamiltonian cycle · Graph with minimum degree 3 · Approximation algorithm · Randomized algorithm

1 Introduction

A classical fact published by Tutte [26] and accredited to Smith states that, in a cubic (i.e. degree-3 regular) graph, every edge is contained in an *even* number of Hamiltonian cycles. The theorem of Smith was later generalized by Thomason [22] to all graphs in which every vertex has an odd degree. Therefore, every Hamiltonian graph with odd-degree vertices has at least one further Hamiltonian cycle; actually it is not hard to derive that also a third Hamiltonian cycle exists. However, if we allow vertices of even degree, the existence of a second Hamiltonian cycle is not guaranteed. More specifically, Entringer and Swart [13] constructed *uniquely Hamiltonian graphs* (i.e. graphs with a unique Hamiltonian cycle), in which all vertices have degree 3, except only two of them which have degree 4. Fleischner [15] also constructed two infinite families of uniquely Hamiltonian graphs, one with minimum degree 4 and maximum degree 14, and one with arbitrarily high maximum degree.

On the other hand, for *regular* graphs it is known that uniquely Hamiltonian graphs cannot have high degree. More specifically, Thomassen [25] proved that, for every $r \geq 300$, every Hamiltonian r-regular graph has a second Hamiltonian cycle. This was subsequently improved to every $r \geq 23$ [17]. It still remains unknown whether every r-regular Hamiltonian graph has a second Hamiltonian cycle, where $4 \leq r \leq 22$ is an even number. Sheehan conjectured in 1975 that this is true for $r = 4$ [21]. If Sheehan's conjecture is true, then this will imply that, for every $r \geq 3$, every r-regular Hamiltonian graph has a second Hamiltonian cycle. For *non-regular* graphs, it is known that a second Hamiltonian cycle is guaranteed to exist also in graphs with large minimum degree. Bondy and Jackson proved this for graphs with n vertices and minimum degree at least $2.41 \log 8n + 3$ [8].[1] This bound was later improved to $1.752 \log n$ [17] and to $1.71 \log n + 2$ [1]. All the above results have been proved with the probabilistic method using non-constructive arguments.

Our focus in this paper is to devise algorithms which, given a graph G and a Hamiltonian cycle C of G, actually *compute* a second Hamiltonian cycle $C' \neq C$ in G, instead of just proving the existence of C'. For cubic (or, more generally, for odd-degree) Hamiltonian graphs, several exponential-time algorithms are known for computing C' [3–7, 10–12, 14, 18, 19, 28]; however, it remains open whether C' can be computed in polynomial, or even subexponential time. One possible relaxation of

[1] All logarithms in this paper are base 2.

the problem is, given a Hamiltonian cycle C, to efficiently compute a second cycle $C' \neq C$ which is large enough (not necessarily Hamiltonian). For cubic Hamiltonian graphs, Bazgan et al. [2] showed that the knowledge of the first Hamiltonian cycle C algorithmically *strictly* helps to approximate the length of a second cycle. In fact, if C is not given along with the input, there is no polynomial-time constant-factor approximation algorithm for finding a long cycle in cubic graphs, unless P = NP. In contrast, if C is given, then for every $\varepsilon > 0$ a cycle $C' \neq C$ of length at least $(1-\varepsilon)n$ can be found in $2^{O(1/\varepsilon^2)} \cdot n$ time, i.e. there is a linear-time PTAS for approximating the second Hamiltonian cycle [2]. The main ingredient in the proof of the latter result is an $O(n^{\frac{3}{2}} \log n)$-time algorithm which, given G and C, computes a cycle $C' \neq C$ of length at least $n - 4\sqrt{n}$ [2]. In wide contrast to cubic graphs, for graphs of minimum degree at least three, only existential proofs are known for a second large cycle. In particular, Girão et al. [16] recently proved with a *non-constructive* argument that any n-vertex Hamiltonian graph with minimum degree at least 3 contains another cycle of length at least $n - o(n)$.

Our Contribution In this paper we consider two relaxations of the problem of computing a second Hamiltonian cycle. First, we consider *approximating* the length of a second cycle on n-vertex graphs with minimum degree δ and maximum degree Δ. We provide in Sect. 2 a linear-time algorithm for computing a cycle $C' \neq C$ of length at least $n - 4a(\sqrt{n} + 2\alpha) + 8$, where $\alpha = \frac{\Delta-2}{\delta-2}$. On the one hand, this improves the results of Bazgan et al. [2] in two ways: (1) it provides a generalization from cubic Hamiltonian graphs to arbitrary Hamiltonian graphs of degree at least 3; (2) our algorithm works in linear time in n for all constant-degree regular graphs. In particular it works in time $O(n)$ on cubic graphs (see Corollary 6). On the other hand, we complement the results of [16] as we provide a *constructive* proof for their result in case where δ and Δ are $o(\sqrt{n})$-factor away from each other. Formally, our algorithm constructs in linear time another cycle of length $n - o(n)$ whenever $\frac{\Delta}{\delta} = o(\sqrt{n})$ (see Corollary 7).

Our second relaxation of the problem is probabilistic. We propose in Sect. 3 a randomized algorithm which computes a second Hamiltonian cycle *with high probability*, given that the input graph G has a large enough minimum degree. More specifically, we prove that for every $0 < p \leq 0.02$, if the minimum degree of G is at least $\frac{8}{p} \log \sqrt{8}n + 4$, then a second Hamiltonian cycle can be computed with probability at least $1 - \frac{1}{n}\left(\frac{50}{p^4} + 1\right)$ in $poly(n) \cdot 2^{4pn}$ time. Thus, by appropriately choosing the value of the parameter p, we can achieve different trade-offs between the minimum degree and the resulting running time, as it is illustrated in Table 1 (see Corollary 15). These results imply that, when the minimum degree δ is sufficiently large, we can compute with high probability a second Hamiltonian cycle faster than any known deterministic algorithm. In particular, when $\delta = \omega(\log n)$, our probabilistic algorithm works in $2^{o(n)}$ time.

Table 1 Summary of our probabilistic results for computing a second Hamiltonian cycle in a Hamiltonian graph of large minimum degree δ, where c is a constant and $\varepsilon \in (0, 0.25)$

Minimum degree δ at least	Running time	Probability at least
$32c \cdot \log \sqrt{8}n + 4$	$poly(n) \cdot 2^{n/c}$	$1 - O\left(\frac{1}{n}\right)$
$64 \log^{c+1} n$	$poly(n) \cdot 2^{n/\log^{c} n}$	$1 - O\left(\frac{\log^{4c} n}{n}\right)$
$32n^{\varepsilon} \cdot \log \sqrt{8}n + 4$	$poly(n) \cdot 2^{n^{1-\varepsilon}}$	$1 - O\left(\frac{1}{n^{1-4\varepsilon}}\right)$

2 Computing Another Long Cycle in a Hamiltonian Graph in Linear Time

Since the complexity status of finding a second Hamiltonian cycle in a given cubic Hamiltonian graph is illusive at the moment, we consider a simpler problem of finding another *almost* Hamiltonian cycle in a given Hamiltonian graph of minimum degree 3. Our main result in this section provides a $O(m)$ time algorithm for finding a $n - o(n)$ cycle in a given Hamiltonian graph of minimum degree at least 3 when the maximum and the minimum degrees of the graph are in certain relation.

Bazgan et al. [2] studied an optimization problem that can be seen as a relaxation of the problem of finding another Hamiltonian cycle in a cubic Hamiltonian graph. Namely, they considered the following problem.

Second Largest Cycle
Input: A cubic Hamiltonian graph G and a Hamiltonian cycle C.
Solution: A cycle different from C.
Value: The length of the cycle.

The authors showed that Second Largest Cycle has an efficient polynomial-time approximation scheme (EPTAS) by proving the following theorem; recall an EPTAS finds a $(1 - \varepsilon)$-approximate solution in time $f(\frac{1}{\varepsilon}) \cdot poly(n)$, where f is a computable function.

Theorem 1 ([2]) *Let G be an n-vertex cubic Hamiltonian graph and let C be a Hamiltonian cycle in G. Given G and C, for every $\varepsilon > 0$, a cycle $C' \neq C$ of length at least $(1 - \varepsilon) \cdot n$ can be found in time $2^{O(1/\varepsilon^2)} \cdot n$.*

The main ingredient of the proof of Theorem 1 is a polynomial-time procedure that finds in G a cycle C' different from C of length at least $n - 4\sqrt{n}$. This result is formalized in the following:

Theorem 2 ([2]) *Let G be an n-vertex cubic Hamiltonian graph and let C be a Hamiltonian cycle in G. There is an algorithm that given G and C finds in G a cycle $C' \neq C$ of length at least $n - 4\sqrt{n}$ in time $O(n^{3/2} \log n)$.*

Our main result of this section improves Theorem 2 in two ways. First, it implies a generalization of Theorem 2 to arbitrary regular Hamiltonian graphs of degree at least 3. Second, our algorithm works in linear time in n for constant degree regular graphs. In particular, it works in time $O(n)$ on cubic graphs (see Corollary 6).

2.1 Notation and Auxiliary Results

Before we proceed to the main result of the section, we introduce preliminary notation and state auxiliary results. Let G be a graph with a designated Hamiltonian cycle $C = v_1 v_2 \dots v_n$. We assume that the the cycle has a fixed, clockwise, direction. Any edge of G that is not in C is called a *chord*. Two chords are *independent* if they do not share an endpoint. The *length* of a chord $v_i v_j$ with $i < j$ is defined as $\min\{j - i, n + i - j\}$. We say that two vertices $u, v \in V(G)$ are *chord-adjacent* if they are connected by a chord of G. Two independent chords e_1 and e_2 are called *crossing* if their endpoints appear in alternating order around C; otherwise e_1 and e_2 are *parallel*.

For $x, y \in V(G)$, we denote by $d_C(x, y)$ the length of the path from x to y around C, using the clockwise direction. Note that $d_C(x, y) \neq d_C(y, x)$ in general. We define the *distance* between two independent chords xy and ab, denoted $\text{dist}(xy, ab)$, as follows:

1. if xy and ab are crossing, then $\text{dist}(xy, ab) = \min\{d_C(x, a)+d_C(y, b), d_C(x, b)+ d_C(y, a)\}$;
2. if xy and ab are parallel such that neither y nor b lie on the path the path from x to a around C, then $\text{dist}(xy, ab) = d_C(x, a) + d_C(y, b)$.

In the proof of our main result we use the following two lemmas. The first is a basic fact from graph theory and the second is straightforward to check (see Fig. 1 for illustration).

Lemma 3 ([27], Exercise 3.1.29) *Let $G = (V, E)$ be a bipartite graph of maximum degree Δ. Then G has a matching of size at least $\frac{|E|}{\Delta}$.*

Lemma 4 *Let $G = (V, E)$ be an n-vertex graph with a Hamiltonian cycle $C = v_1 v_2 \dots v_n$.*

(*1*) *If G has a chord of length ℓ, then G contains a cycle $C' \neq C$ of length at least $n - \ell + 1$.*
(*2*) *If G has two crossing chords e_1, e_2 and* $\text{dist}(e_1, e_2) = d$, *then G contains a cycle $C' \neq C$ of length at least $n - d + 2$.*
(*3*) *If G has four pairwise independent chords e_1, e_2, f_1, and f_2 such that*

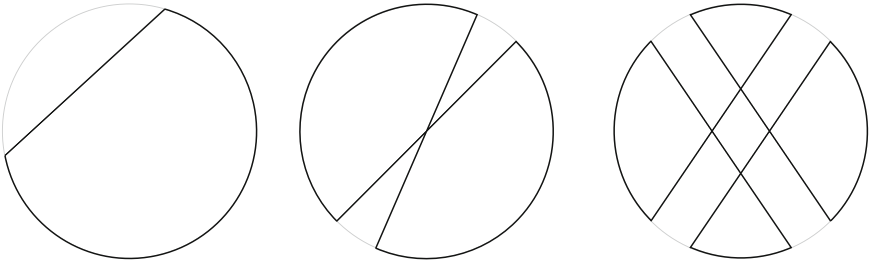
(1) A short chord (2) A pair of crossing chords (3) Crossing pairs of parallel chords

Fig. 1 Illustration of Lemma 4

(*a*) *e_1, e_2 are parallel and f_1, f_2 are parallel,*
(*b*) *e_i and f_j are crossing for every $i, j \in \{1, 2\}$,*
(*c*) $\mathrm{dist}(e_1, e_2) = d_1$ *and* $\mathrm{dist}(f_1, f_2) = d_2$,

then G contains a cycle $C' \neq C$ of length at least $n - d_1 - d_2 + 4$.

2.2 Long Cycles in Hamiltonian Graphs

Theorem 5 *Let $G = (V, E)$ be an n-vertex Hamiltonian graph of minimum degree $\delta = \delta(G) \geq 3$. Let $C = v_1 v_2 \dots v_n$ be a Hamiltonian cycle in G and let Δ denote the maximum degree of G. Then G has a cycle $C' \neq C$ of length at least $n - 4\alpha(\sqrt{n} + 2\alpha) + 8$, where $\alpha = \frac{\Delta-2}{\delta-2}$. Moreover, given C a cycle C' can be computed in $O(m)$ time, where $m = |E|$.*

Proof We start by showing the existence of the desired cycle C'.

Without loss of generality we assume that $\alpha < \frac{\sqrt{n}}{2}$, as otherwise any cycle $C' \neq C$ in G satisfies the theorem.

Furthermore, we assume that the length of every chord in G is at least $4\alpha(\sqrt{n} + 2\alpha) - 6$, as otherwise the existence of C' follows from Lemma 4(1).

Let $q = \alpha\sqrt{n}$.

We partition arbitrarily the vertices of the Hamiltonian cycle C into r consecutive intervals $B_0, B_1, \dots, B_{r-1}$, such that $r \in \left\{\left\lfloor \frac{\sqrt{n}}{\alpha} \right\rfloor, \left\lfloor \frac{\sqrt{n}}{\alpha} \right\rfloor + 1\right\}$ and $\lfloor q \rfloor \leq |B_i| \leq \lfloor q \rfloor + 2\alpha^2$ for every $i \in \{0, 1, \dots, r-1\}$. It is a routine task to check that such a partition exists.

For every $i \in \{0, 1, \dots, r-1\}$ we denote by W_i the set of vertices that are chord-adjacent to a vertex in B_i and by E_i we denote the set of chords that are incident with a vertex in B_i. Furthermore, we denote by H_i the graph with vertex set $B_i \cup W_i$ and the edges set E_i. Since the length of every chord in G is at least $4\alpha(\sqrt{n} + 2\alpha) - 6$, we observe that for every $i \in \{0, 1, \dots, r-1\}$ set W_i is disjoint from $B_{i-1} \cup B_i \cup B_{i+1}$ (where the arithmetic operations with indices are modulo r). The latter, in particular, implies that H_i is a bipartite graph with color classes B_i and W_i.

We say that the intervals B_i and B_j, where $i \neq j$, are *matched* if there exist two independent chords such that each of them has one endpoint in B_i and the other endpoint in B_j. We claim that every interval B_i is matched to another interval B_j for some $j \in \{0, 1, \dots, r-1\} \setminus \{i-1, i, i+1\}$. Indeed, by Lemma 3, graph H_i has a matching M_i of size at least

$$\frac{\lfloor q \rfloor(\delta - 2)}{\Delta - 2} = \frac{\lfloor \alpha\sqrt{n} \rfloor}{\alpha} > \frac{\alpha\sqrt{n} - 1}{\alpha} \geq \sqrt{n} - 1 > \left\lfloor \frac{\sqrt{n}}{\alpha} \right\rfloor - 2 \geq r - 3,$$

and therefore, by the pigeonhole principle, there exists $j \in \{0, 1, \dots, r-1\} \setminus \{i-1, i, i+1\}$ such that at least two edges in M_i have their endpoints in B_j, meaning that B_i is matched to B_j.

Let $\sigma : \{0, 1, \dots, r-1\} \to \{0, 1, \dots, r-1\}$ be a function such that B_i is matched to $B_{\sigma(i)}$, and denote by $f_{i,1}$ and $f_{i,2}$ some fixed pair of independent chords between B_i and $B_{\sigma(i)}$. We observe that $\mathrm{dist}(f_{i,1}, f_{i,2}) \leq 2(\lfloor q \rfloor + 2\alpha^2 - 1) \leq 2\alpha(\sqrt{n} + 2\alpha) - 2$, as the endpoints of $f_{i,1}$ and $f_{i,2}$ lie in the intervals B_i and $B_{\sigma(i)}$ each of length at most $\lfloor q \rfloor + 2\alpha^2$.

Let now R be an auxiliary graph with a Hamiltonian cycle $x_0 x_1 \dots x_{r-1}$ and the chord set being $\{x_i x_{\sigma(i)} : i = 0, 1, \dots, r-1\}$.

Let $x_i x_j$ be a chord in R of the minimum length, where $j = \sigma(i)$. Without loss of generality, we assume that $i < j$ and $j - i \leq r + i - j$. Let x_k be a vertex of R such that $i < k < j$ and let $s = \sigma(k)$.

Since $x_i x_j$ is of minimum length, the chords $x_i x_j$ and $x_k x_s$ are crossing, and hence each of $f_{i,1}$ and $f_{i,2}$ crosses both $f_{k,1}$ and $f_{k,2}$.

Finally, if $f_{i,1}$, $f_{i,2}$, or $f_{k,1}$, $f_{k,2}$, are crossing, then by Lemma 4(2) there exists a cycle $C' \neq C$ of length at least $n - 2\alpha(\sqrt{n} + 2\alpha) + 4$. Otherwise, $f_{i,1}$, $f_{i,2}$ are parallel and $f_{k,1}$, $f_{k,2}$ are parallel, and hence by Lemma 4(3) there exists a cycle $C' \neq C$ of length at least $n - 4\alpha(\sqrt{n} + 2\alpha) + 8$, which proves the first part of the theorem.

The above proof is constructive. We now explain at a high level how the proof can be turned into the desired algorithm. First, if $\alpha \geq \frac{\sqrt{n}}{2}$, then we output any cycle formed by a chord and the longer path of C connecting the endpoints of the chord. Otherwise, we partition the vertices of C into intervals $B_0, B_1, \dots, B_{r-1}$ and assign to each vertex the index of its interval. Clearly, this can be done in $O(n)$ time. Next, we traverse the vertices of G along the cycle C and for every vertex v of an interval B_i we check the chords incident to v. If we encounter a chord f of length less than $4\alpha(\sqrt{n} + 2\alpha) - 6$, then we output the cycle formed by f and the longer path of C connecting the endpoints of f. Otherwise, for the interval B_i we keep the information of how many and which vertices of W_i belong to other intervals B_j for $j \in \{0, 1, \dots, r-1\} \setminus \{i-1, i, i+1\}$. When we find an interval B_j that has at least two elements from W_i, we set $\sigma(i)$ to j and proceed to the first vertex of the next interval B_{i+1}. By doing this, we also keep the information of the current shortest chord in the graph R (defined in the proof above). After finishing this procedure: (1) we have a function $\sigma(\cdot)$; (2) for every $i \in \{0, 1, \dots, r-1\}$ we know a pair $f_{i,1}$, $f_{i,2}$ of independent edges between B_i and $B_{\sigma(i)}$; and (3) we know k such that $x_k x_{\sigma(k)}$ is a minimum length chord in R. Clearly, this information is enough to identify the desired cycle in constant time. In total, we spent $O(n)$ time to compute the partition of the vertices into the intervals and we visited every chord at most twice, which implies the claimed $O(m)$ running time. □

By noticing that in the above theorem parameter α is equal to 1 for any regular graph, we obtain the following corollary.

Corollary 6 *Let $G = (V, E)$ be an n-vertex Hamiltonian δ-regular graph with $\delta \geq 3$, and let C be a Hamiltonian cycle in G. Then G has a cycle $C' \neq C$ of length at least $n - 4\sqrt{n}$, which can be computed in $O(\delta n)$ time.*

Corollary 7 *Let $G = (V, E)$ be an n-vertex Hamiltonian graph of minimum degree $\delta = \delta(G) \geq 3$. Let C be a Hamiltonian cycle in G and let Δ denote the maximum degree of G. If $\frac{\Delta}{\delta} = o(\sqrt{n})$, then G has a cycle $C' \neq C$ of length at least $n - o(n)$, which can be computed in $O(m)$ time.*

3 Second Hamiltonian Cycle in Graphs of Large Minimum Degree

In this section we provide an algorithm (see Theorem 14) which, given a Hamiltonian cycle C in an arbitrary Hamiltonian graph G with n vertices and with large minimum degree $\delta = \Omega(\log n)$, computes a second Hamiltonian cycle $C' \neq C$ faster than the best known algorithms (cf. the discussion in the Introduction).

3.1 Second Hamiltonian Cycle via Red-Independent and Green-Dominating Sets

Let G be a graph with a Hamiltonian cycle C. Colour the edges of C red and the chords green. A neighbor u of a vertex v is a *red* (resp. *green*) *neighbor* of v if uv is a red (resp. green) edge. A set X of vertices of G is *red-independent* if the subgraph $G[X]$ of G induced by X contains no red edges, and X is *green-dominating* if every vertex outside X is adjacent to a vertex in X via green edge. The following lemma is a classical result of Thomassen [24] proved via Thomason's lollipop argument [23].

Lemma 8 (Thomassen's lemma, [24]) *If G has a red-independent and green-dominating set of vertices, then G contains a Hamiltonian cycle $C' \neq C$.*

Motivated by Thomassen's lemma, it is natural to ask whether the knowledge of a red-independent and green-dominating set X in G can improve the time complexity of finding a second Hamiltonian cycle. The proof of Lemma 8 is reduced to the application of Thomason's lollipop argument and therefore does not immediately imply an efficient algorithm for finding a second Hamiltonian cycle; it is known that there exist cubic graphs where Thomason's algorithm requires exponential time [9]. However, if X is not too large, it can in fact be used to speed up the computation of a second Hamiltonian cycle. To explain this, we need to state another lemma, which can be easily extracted from Thomassen's proof. Following the terminology of Bondy and Jackson [8], a set of vertices X of G is *weakly green-dominating* if every red neighbor of a vertex in X is also a green neighbor of some vertex in X.

Lemma 9 ([24]) *If G has a red-independent and* weakly *green-dominating set of vertices, then G contains a Hamiltonian cycle $C' \neq C$.*

Clearly, any green-dominating set X in G is also weakly green-dominating. Let X be a red-independent and green-dominating set of size k, and let $P_1, P_2, \ldots, P_k$ be the k paths obtained from C by removing the vertices of X. Let G' be the graph obtained from G by removing all chords connecting two vertices not in X and all chords connecting a vertex in X and an internal vertex of any of the paths $P_1, P_2 \ldots, P_k$. Clearly, X is red-independent and weakly green-dominating in G', and therefore, by Lemma 9, G' contains a Hamiltonian cycle $C' \neq C$, which is also a Hamiltonian cycle in G.

A simple, but important observation is that all *internal* vertices of the paths $P_1, P_2 \ldots, P_k$ have degree two in G', and hence all edges incident with these vertices belong to *every* Hamiltonian cycle of G'. Note here that the extreme vertices of these paths $P_1, P_2 \ldots, P_k$ might have a higher degree. Let H be the graph obtained from G' by removing, for every path $P \in \{P_1, P_2 \ldots, P_k\}$ with at least four vertices, all its internal vertices and adding one new vertex and two new edges connecting the new vertex with the endpoints of P. Then, there is an obvious bijection between the Hamiltonian cycles of G' and the Hamiltonian cycles of H. Denote by C_H the Hamiltonian cycle of H corresponding to the Hamiltonian cycle C of G'. We observe that X is a red-independent and weakly green-dominating set (with respect to C_H) in H. Therefore, by Lemma 9, H contains a Hamiltonian cycle $C'_H \neq C_H$, which in turn corresponds to a cycle $C' \neq C$ of G' and can be efficiently computed from C'_H. We call the graph H *shrunken* and observe that the number of vertices in H is at most $k + 3k = 4k$. If k is small enough, then applying a best known algorithm A for finding a Hamiltonian cycle to every graph $H - e$, where e is a red edge in H that is incident to a vertex in X, one can find a Hamiltonian cycle $C'_H \neq C_H$ in H, and therefore a Hamiltonian cycle $C' \neq C$ in G, faster than applying n times the best known algorithm A directly to G. The above implies the correctness and running time of Algorithm 1, which is formally given in Theorem 10.

Algorithm 1 Second Hamiltonian Cycle via red-independent and green-dominating set

Input: A graph $G = (V, E)$; a Hamiltonian cycle C of G; a red-independent and green-dominating set $X \subseteq V$ in G; an algorithm A with running time $f_A(n)$ that, given an input n-vertex graph, either computes a Hamiltonian cycle or concludes that no Hamiltonian cycle exists.
Output: A Hamiltonian cycle C' in G such that $C' \neq C$.

```
Using set $X$, construct from $G$ the shrunken graph $H$
for every red edge $e$ in $H$ that is incident to a vertex in $X$ do
    Check using algorithm $A$ whether $H - e$ admits a Hamiltonian cycle
    If such a Hamiltonian cycle exists then denote it by $C'_H$ {$C'_H$ exists for at least one edge $e$}
Reconstruct from $C'_H$ a Hamiltonian cycle $C'$ in $G$ such that $C' \neq C$

return $C'$
```

Theorem 10 *Let $G = (V, E)$ be an n-vertex graph, let C be a Hamiltonian cycle in G, and let X be a red-independent and green-dominating set in G, where $|X| = k$. Let A be an algorithm that, given an input n-vertex graph, computes a Hamiltonian cycle in time $f_A(n)$. Then* Algorithm 1 *computes a second Hamiltonian cycle in time at most $poly(n) \cdot f_A(4k)$.*

Using the above approach, we develop in the subsequent sections a randomized algorithm that computes a second Hamiltonian cycle in Hamiltonian graphs of large minimum degree.

3.2 Finding Sparse Red-Independent Green-Dominating Sets in Graphs of Large Minimum Degree

In this section we give a procedure that with high probability produces a red-independent and green-dominating set in graphs of large minimum degree. The procedure is a generalization of a random process used by Bondy and Jackson [8] to show that every n-vertex Hamiltonian graph of minimum degree at least $c \log n$, for some constant c and large enough n, has a red-independent and green-dominating set, and hence a second Hamiltonian cycle.

For convenience, we reuse most of the notation and terminology from [8]. Let $G = (V, E)$, where $V = \{0, 1, \dots, n-1\}$, be a graph on n vertices and let $C = (0, 1, 2, \dots, n-1, 0)$ be a Hamiltonian cycle in G. Let $p \in (0, 1)$ be a fixed number. We define a random subset X of V by the following procedure:
For $p = \frac{1}{2}$ this is exactly the procedure used by Bondy and Jackson [8] to show that if every vertex in G has degree at least $c \log 8n + 4$, where $c = (2 - \log 3)^{-1}$, then with non-zero probability the set X produced by the procedure is red-independent and green-dominating. This implies that any such graph contains at least one red-independent and green-dominating set and therefore contains a second Hamiltonian cycle.

We will generalize and complement the arguments of Bondy and Jackson to show that for any small enough p, if every vertex in G has degree at least $\frac{8}{p} \log \sqrt{8}n + 4$, then, with high probability, X is red-independent *and* green-dominating *and* also has size at most pn. In Sect. 3.3, we will combine this result with the general approach from Sect. 3.1 to obtain faster exponential and subexponential algorithms that find a second Hamiltonian cycle in graphs of large minimum degree with high probability. We proceed with some notation and an auxiliary lemma.

Let $d \in \mathbb{N}$. A *k-part composition* of d is an ordered k-tuple $(d_1, d_2, \dots, d_k)$ of positive integers summing up to d. We call each d_i a *part* of this composition of the number d. Denote by $C(k, d)$ the set of all k-part compositions of d, in which $d_i \geq 2$ for every $i \in \{2, \dots, k-1\}$. We write

$$c(k, d) = |C(k, d)| \quad \text{and} \quad C(d) = \bigcup_{k \geq 1} C(k, d).$$

Similarly, we denote by $C^*(k, d)$ the set of all k-part compositions of d, in which $d_i \geq 2$ for every i, except possibly *at most one* of the two parts d_1 or d_k. We write

$$c^*(k, d) = |C^*(k, d)| \quad \text{and} \quad C^*(d) = \bigcup_{k \geq 1} C^*(k, d).$$

Lemma 11 *Let $k, d \in \mathbb{N}$ and $\mu \in (0, 2]$, and let $\alpha = \frac{\sqrt{4\mu+1}-1}{2}$ and $\beta = \frac{\sqrt{4\mu+1}+1}{2}$. Then*

(a) $\sum_{k \geq 1} \mu^{k-1} \cdot c(k, d) = \frac{1}{\alpha+\beta}\left(\beta^{d+1} + (-1)^d \alpha^{d+1}\right) \leq \left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right) \beta^d$;

(b) $\sum_{k \geq 1} \mu^{k-1} \cdot c^*(k, d) = \beta^{d-1} + (-1)^{d-1} \alpha^{d-1} \leq 2\beta^{d-1}$,

Proof In both cases we will count weighted compositions of d in which each composition into k parts is given weight μ^{k-1}. This is done by assigning weight one to the first part d_1 and weight μ to each of the remaining parts. For a polynomial $f(x)$ we denote by $[x^d]f(x)$ the coefficient of x^d in $f(x)$. We will also make use of the following equation, which is a routine task to verify:

$$\begin{aligned}
&[x^d](x+x^2+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)(\mu x+\mu x^2+\cdots)\\
&=[x^d]\left(\frac{x}{1-x}\right)\left(\frac{1}{1-\frac{\mu x^2}{1-x}}\right)\left(\frac{\mu x}{1-x}\right)\\
&=[x^d]\frac{1}{\alpha+\beta}\left(\frac{\beta}{1-\beta x}-\frac{\alpha+\beta}{1-x}+\frac{\alpha}{1+\alpha x}\right)\\
&=\frac{1}{\alpha+\beta}\left(\beta^{d+1}-(\alpha+\beta)+(-1)^d\alpha^{d+1}\right)\\
&=\frac{1}{\alpha+\beta}\left(\beta^{d+1}+(-1)^d\alpha^{d+1}\right)-1.
\end{aligned}\tag{1}$$

We proceed now by proving item (a).

$$\begin{aligned}
\sum_{k\geq 1}\mu^{k-1}\cdot c(k,d) &= 1+\sum_{k\geq 2}\mu^{k-1}c(k,d)\\
&= 1+[x^d](x+x^2+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)\\
&\quad(\mu x+\mu x^2+\cdots)\overset{(1)}{=}\frac{1}{\alpha+\beta}\left(\beta^{d+1}+(-1)^d\alpha^{d+1}\right).
\end{aligned}\tag{2}$$

Now, since $\mu\leq 2$, we have $\frac{\alpha}{\beta}\leq\frac{1}{2}$, and therefore

$$\begin{aligned}
\sum_{k\geq 1}\mu^{k-1}\cdot c(k,d) &\overset{(2)}{=}\frac{1}{\alpha+\beta}\left(\beta^{d+1}+(-1)^d\alpha^{d+1}\right)\\
&=\frac{1}{\alpha+\beta}\left(\beta+(-1)^d\alpha\left(\frac{\alpha}{\beta}\right)^d\right)\beta^d\\
&\leq\frac{1}{\alpha+\beta}\left(\beta+\frac{\alpha}{4}\right)\beta^d=\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)\beta^d.
\end{aligned}$$

To prove item (b) we first notice that by the inclusion–exclusion principle

$$\begin{aligned}
&\sum_{k\geq 1}\mu^{k-1}\cdot c^*(k,d)=1+\sum_{k\geq 2}\mu^{k-1}\cdot\left(2c'(k,d)-c''(k,d)\right)=1\\
&+2\sum_{k\geq 2}\mu^{k-1}\cdot c'(k,d)-\sum_{k\geq 2}\mu^{k-1}\cdot c''(k,d),
\end{aligned}\tag{3}$$

where $c'(k,d)$ is the number of k-part compositions of d in which all parts are at least two, except possibly the first one; and $c''(k,d)$ is the number of k-part compositions of d in which all parts are at least two. Now,

$$\begin{aligned}
&\sum_{k\geq 2}\mu^{k-1}\cdot c'(k,d)\\
&\qquad = [x^d](x+x^2+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)(\mu x^2+\mu x^3+\cdots)\\
&\qquad = [x^{d-1}](x+x^2+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)(\mu x+\mu x^2+\cdots)\\
&\qquad \overset{(1)}{=} \frac{1}{\alpha+\beta}\left(\beta^d+(-1)^{d-1}\alpha^d\right)-1. \qquad (4)
\end{aligned}$$

Similarly,

$$\begin{aligned}
&\sum_{k\geq 2}\mu^{k-1}\cdot c''(k,d)\\
&\qquad = [x^d](x^2+x^3+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)(\mu x^2+\mu x^3+\cdots)\\
&\qquad = [x^{d-2}](x+x^2+\cdots)\left(\sum_{i\geq 0}(\mu x^2+\mu x^3+\cdots)^i\right)(\mu x+\mu x^2+\cdots)\\
&\qquad \overset{(1)}{=} \frac{1}{\alpha+\beta}\left(\beta^{d-1}+(-1)^{d-2}\alpha^{d-1}\right)-1. \qquad (5)
\end{aligned}$$

Plugging in (4) and (5)–(3) and using the fact that $-2\beta+1=-(2\alpha+1)=-(\alpha+\beta)$ we obtain

$$\begin{aligned}
&\sum_{k\geq 1}\mu^{k-1}\cdot c^*(k,d)\\
&\quad = 1+\frac{2}{\alpha+\beta}\left(\beta^d+(-1)^{d-1}\alpha^d\right)-2-\frac{1}{\alpha+\beta}\left(\beta^{d-1}+(-1)^{d-2}\alpha^{d-1}\right)+1\\
&\quad = \frac{1}{\alpha+\beta}\left(2\beta^d+2(-1)^{d-1}\alpha^d-\beta^{d-1}+(-1)^{d-1}\alpha^{d-1}\right)\\
&\quad = \frac{1}{\alpha+\beta}\left((-1)^{d-1}\alpha^{d-1}(2\alpha+1)-\beta^{d-1}(-2\beta+1)\right)\\
&\quad = \beta^{d-1}+(-1)^{d-1}\alpha^{d-1}\\
&\quad \leq \beta^{d-1}+\alpha^{d-1}\leq 2\beta^{d-1},
\end{aligned}$$

where in the latter inequality we used the fact that $\alpha\leq\beta$. □

Green-Dominating Clearly, the set X produced by Algorithm 2 is always red-independent. We show next that, if the minimum degree of G is large enough, then X is also green-dominating with high probability.

Lemma 12 *Let* $p \in (0, 0.2]$. *If every vertex in* G *has degree at least* $\frac{8}{p} \log \sqrt{8}n + 4$, *then the set* X *produced by* Algorithm 2 *is green-dominating with probability at least* $1 - \frac{1}{n}$.

Proof Let $m \in \{0, 1, \ldots, n-1\}$ be a fixed vertex of G. The *green neighborhood* $\widehat{N}(m)$ of m is the set

$$\widehat{N}(m) = N(m) \setminus \{m-1, m+1\} = \{m_1, m_2, \ldots, m_t\},$$

where $0 \le m_1 < m_2 < \ldots < m_t$ and the arithmetic operations are modulo n.

If $m_1 = 0$, then $0 \in \widehat{N}(m) \cap X$. We shall show that in all other cases $\widehat{N}(m) \cap X = \emptyset$ with probability less than n^{-2}, which implies by the union bound that with probability at least $1 - \frac{1}{n}$ the set X is green-dominating.

Algorithm 2 Bondy and Jackson's procedure

Input: A graph $G = (V, E)$; a Hamiltonian cycle $C = (0, 1, 2, \ldots, n-1, 0)$ of G; a number $p \in (0, 1)$.
Output: A red-independent set $X \subseteq V$.

```
X = {0}
for i = 1 to n − 2 do
    if i − 1 does not belong to X then
        Add i to X with probability p and skip with probability 1 − p
return X
```

Denote by $\mathcal{A}$ the set of all sequences $A = (a_0, a_1, \ldots, a_r)$, where $0 = a_0 < a_1 < \ldots < a_r < n-1$, and $a_{i+1} - a_i \ge 2$, $0 \le i \le r-1$. For $A = (a_0, a_1, \ldots, a_r) \in \mathcal{A}$,

$$\begin{aligned} P(X = A) &= \prod_{i=1}^{r} p(1-p)^{a_i - a_{i-1} - 2} \left(\sum_{i=0}^{\infty} p(1-p)^{-a_r + (n-1+i) - 2} \right) \\ &= p^{r+1}(1-p)^{n-2r-3} \sum_{i=0}^{\infty} (1-p)^i \\ &= p^r (1-p)^{n-2r-3} \\ &= (1-p)^{n-3} \left(\frac{p}{(1-p)^2} \right)^r \\ &= (1-p)^{n-3} \mu^r, \end{aligned}$$

where we denote $\mu = \frac{p}{(1-p)^2}$.

Let now $d_1 = m_1$, $d_i = m_i - m_{i-1}$, $2 \le i \le t$, and $d_{t+1} = n - m_t$, and

$$\mathcal{C} = C^*(d_1) \times C(d_2) \times \cdots \times C(d_t) \times C^*(d_{t+1}).$$

There is a natural bijection between $\mathcal{C}$ and the subset $\widehat{\mathcal{A}}$ of $\mathcal{A}$ consisting of those sequences which are disjoint from $\widehat{N}(m)$. Moreover, for $A = (a_0, a_1, \ldots, a_r) \in \widehat{\mathcal{A}}$,

$$P(X = A) = (1-p)^{n-3}\mu^r = (1-p)^{n-3}\prod_{i=1}^{t+1}\mu^{k_i-1}, \tag{6}$$

where k_i is the number of parts in the corresponding composition of d_i, $1 \le i \le t+1$.

Note that

$$\begin{aligned}\mathcal{C} = &\left(\bigcup_{k_1\ge 1} C^*(k_1, d_1)\right) \times \left(\bigcup_{k_2\ge 1} C(k_2, d_2)\right) \times \cdots \times \left(\bigcup_{k_t\ge 1} C(k_t, d_t)\right)\\ &\times \left(\bigcup_{k_{t+1}\ge 1} C^*(k_{t+1}, d_{t+1})\right) = \bigcup \Big(C^*(k_1, d_1) \times C(k_2, d_2) \times \cdots \times C(k_t, d_t)\\ &\times C^*(k_{t+1}, d_{t+1})\Big),\end{aligned} \tag{7}$$

where the union is taken over all $(t+1)$-tuples $(k_1, k_2, \ldots, k_{t+1})$ of positive integers. Therefore (6) implies that

$$\begin{aligned}P(X \in \widehat{\mathcal{A}}) = (1-p)^{n-3}\sum\Bigg[&\mu^{k_1-1}\cdot c^*(k_1, d_1)\left(\prod_{i=2}^{t}\mu^{k_i-1}\cdot c(k_i, d_i)\right)\\ &\mu^{k_{t+1}-1}\cdot c^*(k_{t+1}, d_{t+1})\Bigg],\end{aligned}$$

where the summation is taken over all $(t+1)$-tuples $(k_1, k_2, \ldots, k_{t+1})$ of positive integers. Thus, by Lemma 11,

$$\begin{aligned}&P(X \in \widehat{\mathcal{A}})\\ &= (1-p)^{n-3}\left(\sum_{k_1\ge 1}\mu^{k_1-1}c^*(k_1, d_1)\right)\left(\prod_{i=2}^{t}\sum_{k_i\ge 1}\mu^{k_i-1}c(k_i, d_i)\right)\\ &\left(\sum_{k_{t+1}\ge 1}\mu^{k_{t+1}-1}c^*(k_{t+1}, d_{t+1}))\right)\\ &\le (1-p)^{n-3}\left(\frac{2}{\beta}\right)\beta^{d_1}\left(\prod_{i=2}^{t}\left[\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)\beta^{d_i}\right]\right)\left(\frac{2}{\beta}\right)\beta^{d_{t+1}}\\ &= (1-p)^{n-3}\beta^{\left(\sum_{i=1}^{t+1}d_i\right)}\left(\frac{2}{\beta}\right)^2\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)^{t-1}\\ &= (1-p)^{n-3}\beta^n\left(\frac{2}{\beta}\right)^2\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)^{t-1}.\end{aligned}$$

Now, since $\mu = \frac{p}{(1-p)^2}$ and $\beta = \frac{\sqrt{4\mu+1}+1}{2}$ (cf. Lemma 11), it follows that $1 + p \leq \beta \leq \frac{1}{1-p}$ for every $p \in (0, 1)$. Therefore, for $p \leq 0.2$, we have

$$\begin{aligned} &P(X \in \widehat{\mathcal{A}}) \\ &\leq (1-p)^{n-3}\left(\frac{1}{1-p}\right)^n\left(\frac{2}{1+p}\right)^2\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)^{t-1} \\ &= \frac{4}{(1-p)^3(1+p)^2}\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)^{t-1} \\ &\leq 8\left(\frac{\alpha+4\beta}{4(\alpha+\beta)}\right)^{t-1}. \end{aligned}$$

Therefore, if the degree of the fixed vertex m is equal to $t+2 \geq c \log \sqrt{8}n + 4$, where $c \geq 2 \log^{-1} \frac{4(\alpha+\beta)}{\alpha+4\beta}$, then $P(X \in \widehat{\mathcal{A}}) < \frac{1}{n^2}$. To conclude the proof, it is enough to show that $\frac{8}{p} \geq 2 \log^{-1} \frac{4(\alpha+\beta)}{\alpha+4\beta}$ or, equivalently, $\log \frac{4(\alpha+\beta)}{\alpha+4\beta} \geq \frac{p}{4}$ for any $p \in (0, 0.2]$.

First, using series expansions one can derive $p(1+p) \leq \alpha \leq \frac{p}{(1-p)^2}$ and $1 + p \leq \beta \leq \frac{1-p+p^2}{(1-p)^2}$ in the target interval $p \in (0, 0.2]$. Therefore,

$$\log \frac{4(\alpha+\beta)}{\alpha+4\beta} \geq \log \frac{4(p(1+p)+(1+p))}{\frac{p}{(1-p)^2}+\frac{4(1-p+p^2)}{(1-p)^2}} = \log \frac{4(p+1)^2(1-p)^2}{p+4(1-p+p^2)}.$$

Let us denote the latter function by $f(p)$. One can check that $f(p)$ is concave in the interval $[0, 0.2]$ and $f(p) \geq \frac{p}{4}$ in the extreme points of the interval. Consequently, $f(p) \geq \frac{p}{4}$ holds in all points of the interval, which gives the desired conclusion and completes the proof. □

In the remainder of this section we are going to show that the size of X is at most pn with high probability. To this end, we formalize Algorithm 2 as a Markov chain with two states: State 0 and State 1. At time point $t = 0$ the Markov chain is in State 1, and for $t \geq 1$

1. if at time point $t-1$ the Markov chain is in State 0, then at time point t it is in State 1 with probability p and in State 0 with probability $1-p$;
2. if at time point $t-1$ the Markov chain is in State 1, then at time point t it is in State 0.

Let $N = n-1$. For $i \in \{0, 1, \ldots, N-1\}$ we denote by ξ_i the random variable which is equal to $\alpha \in \{0, 1\}$ if the Markov chain is in State α at time point i. We interpret the Markov chain as follows: a vertex i is included in X if and only if the Markov chain is in State 1 at time point i, i.e. $\xi_i = 1$. In particular, $X = \{i \mid \xi_i = 1, i = 0, 1, \ldots, N-1\}$, $|X| = \sum_{i=0}^{N-1} \xi_i$.

Denote by $\Pi = (0, 1)$ the vector of initial distribution of the Markov chain, which means that at time point 0 the Markov chain is in State 1 with probability 1 (this corresponds to inclusion of 0 to X). Let also T denote the transition matrix

$$T = \begin{pmatrix} 1-p & p \\ 1 & 0 \end{pmatrix}$$

of the Markov chain. Then the probability distribution of ξ_i is defined by the vector $\Pi^{(i)} = \Pi \cdot T^i$.

It can be shown by induction that

$$T^i = \frac{1}{1+p}\begin{pmatrix} 1 & p \\ 1 & p \end{pmatrix} + \frac{(-p)^i}{1+p}\begin{pmatrix} p & -p \\ -1 & 1 \end{pmatrix},$$

and hence

$$\Pi^{(i)} = \Pi \cdot T^i = \frac{1}{1+p}\left(1-(-p)^i, \quad p+(-p)^i\right),$$

that is, $\mathbb{P}[\xi_i = 0] = \frac{1-(-p)^i}{1+p}$ and $\mathbb{P}[\xi_i = 1] = \frac{p+(-p)^i}{1+p}$.

Lemma 13 *For n large enough and $p \in (0, 0.2]$, the size of the set X produced by* Algorithm 2 *is at most pn with probability at least $1 - \frac{50}{np^4}$.*

Proof As before, let $N = n - 1$. To prove the statement, we first show that for $\varepsilon > 0$

$$\mathbb{P}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right| > \varepsilon\right] \leq \frac{34}{N\varepsilon^2}. \tag{8}$$

By Chebyshev's inequality we have

$$\mathbb{P}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right| > \varepsilon\right] \leq \frac{\mathbb{E}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right|^2\right]}{\varepsilon^2}.$$

Therefore, in order to prove (8), we only need to show that $\mathbb{E}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right|^2\right] \leq \frac{34}{N}$.

Let $\pi = \frac{p}{1+p}$ and $p_1^{(i)} = \mathbb{P}[\xi_i = 1] = \frac{p+(-p)^i}{1+p}$. Then

$$\begin{aligned}\mathbb{E}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right|^2\right] &= \mathbb{E}\left[\left|\frac{\sum_{i=0}^{N-1}\xi_i}{N} - \pi\right|^2\right] = \frac{1}{N^2}\mathbb{E}\left[\left(\sum_{i=0}^{N-1}(\xi_i - \pi)\right)^2\right] \\ &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}\delta_{ik},\end{aligned} \tag{9}$$

where $\delta_{ik} = \mathbb{E}[\xi_i\xi_k] - \pi\mathbb{E}[\xi_i] - \pi\mathbb{E}[\xi_k] + \pi^2$.

Observe that $|p_1^{(i)} - \pi| \le \frac{p^i}{1+p}$, and hence for $i, k \in \{0, \dots, N\}$

$$\begin{aligned}\delta_{ik} &= \mathbb{E}[\xi_i \xi_k] - \pi\mathbb{E}[\xi_i] - \pi\mathbb{E}[\xi_k] + \pi^2 \\ &= p_1^{(s)} p_1^{(t)} - \pi p_1^{(i)} - \pi p_1^{(k)} + \pi^2 \\ &\le \frac{2}{1+p}(p^s + p^t + p^i + p^k),\end{aligned} \tag{10}$$

where $s = \min\{i, k\}$ and $t = |i - k|$. Therefore, by plunging in (10)–(9) we obtain

$$\begin{aligned}\mathbb{E}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right|^2\right] &= \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}\delta_{ik} \\ &\le \frac{2}{N^2(1+p)}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1}[p^s + p^t + p^i + p^k] \\ &\le \frac{8}{N^2(1+p)}\sum_{i=0}^{N-1}\sum_{k=0}^{N-1} p^{\min\{s,t\}} \\ &\le \frac{8}{N^2(1+p)}\sum_{i=0}^{\lceil N/2\rceil - 1} p^i\big[2(N - 2i - 1) + 2(N - 2i)\big] \\ &\le \frac{8}{N^2(1+p)} \cdot \frac{4N}{1-p} \le \frac{34}{N},\end{aligned}$$

where in the latter inequality we used the assumption $p \le 0.2$. This proves (8).

Now, by setting $\varepsilon = \frac{p^2}{1+p}$, we deduce from (8) that

$$\mathbb{P}\left[\left|\frac{|X|}{N} - \frac{p}{1+p}\right| > \frac{p^2}{1+p}\right] \le \frac{34(1+p)^2}{Np^4} \le \frac{49}{Np^4},$$

which implies $\mathbb{P}[|X| \le pN] \ge 1 - \frac{49}{Np^4}$ and therefore $\mathbb{P}[|X| \le pn] \ge 1 - \frac{50}{np^4}$ for all $n \ge 50$. □

3.3 Faster Algorithms for Second Hamiltonian Cycle

In this section we combine Algorithm 2 with the general procedure from Sect. 3.1 to obtain randomized algorithms for finding a second Hamiltonian cycle in graphs of large minimum degree.

Theorem 14 *Let $G = (V, E)$ be an n-vertex graph, C be a Hamiltonian cycle in G, and $p \in (0, 0.2]$. If every vertex of G has degree at least $\frac{8}{p}\log\sqrt{8}n + 4$, then given G, C, and p,* Algorithm 3 *succeeds with probability at least $1 - \frac{1}{n}\left(\frac{50}{p^4} + 1\right)$. Furthermore, the algorithm works in $poly(n) \cdot 2^{4pn}$ time.*

Algorithm 3 Randomized algorithm for Second Hamiltonian Cycle

Input: A graph $G = (V, E)$; a Hamiltonian cycle C of G; a $p \in (0, 1)$.
Output: A Hamiltonian cycle C' in G such that $C' \neq C$, or a "fail" announcement.

```
Given $G$, $C$, and $p$, construct a set $X$ using Algorithm 2
if $X$ is a red-independent and green-dominating set of size at most $pn$ then
    Find a second Hamiltonian cycle $C'$ in $G$ using Algorithm 1 with input parameters: $G$, $C$, $X$ and the
    Bellman–Held–Karp algorithm
    return $C'$
else
    return Fail
```

Proof Algorithm 3 fails if and only if Algorithm 2 outputs a non-green-dominating set or the size of the set exceeds pn. The former happens with probability at most $\frac{1}{n}$, by Lemma 12, and the latter happens with probability at most $\frac{50}{np^4}$, by Lemma 13. Hence, the algorithm succeeds with probability at least $1 - \frac{1}{n}\left(\frac{50}{p^4} + 1\right)$.

Now, if Algorithm 3 does not fail, by Theorem 10, it constructs a second Hamiltonian cycle in G in time not exceeding $poly(n) \cdot f(4pn)$, where $f(k) = O(k^2 2^k)$ is the time complexity of the Bellman–Held–Karp algorithm [3, 18]. Hence the theorem. □

The next corollary follows by Theorem 14, when replacing the factor $\frac{8}{p}$ by either (1) a constant $32c$, or (2) a polylogarithmic factor $32\log^c n$, or (3) a polynomial factor $32n^\varepsilon$ (where $\varepsilon < 0.25$). Note that the second case of the corollary is implied by the fact that $64\log^{c+1} n > 32\log^c n \cdot \log\sqrt{8}n + 4$, for large enough n.

Corollary 15 *Let c, ε be positive constants, where $\varepsilon < 0.25$. Let $G = (V, E)$ be an n-vertex graph and C be a Hamiltonian cycle in G. Then:*

1. *If every vertex of G has degree at least $32c \cdot \log\sqrt{8}n + 4$, then* Algorithm 3 *finds a second Hamiltonian cycle in G in time $poly(n) \cdot 2^{n/c}$ with probability at least $1 - O\left(\frac{1}{n}\right)$.*
2. *If every vertex of G has degree at least $64\log^{c+1} n$, then* Algorithm 3 *finds a second Hamiltonian cycle in G in time $poly(n) \cdot 2^{n/\log^c n}$ with probability at least $1 - O\left(\frac{\log^{4c} n}{n}\right)$.*
3. *If every vertex of G has degree at least $32n^\varepsilon \cdot \log\sqrt{8}n + 4$, then* Algorithm 3 *finds a second Hamiltonian cycle in G in time $poly(n) \cdot 2^{n^{1-\varepsilon}}$ with probability at least $1 - O\left(\frac{1}{n^{1-4\varepsilon}}\right)$.*

4 Conclusions

We conclude by stating some questions for future research. The most prominent problem is to classify the complexity of finding a second Hamiltonian cycle, when its existence is guaranteed and we are given a first Hamiltonian cycle.

For cubic graphs, the only known upper bound for the problem is the complexity class PPA [20]. It is unknown whether the problem is hard for PPA, or some other complexity class, or if it can be solved in polynomial time. In fact, it is not even known, if the problem can be solved in $2^{o(n)}$ time by a randomized algorithm.

Our randomized algorithm can achieve such time complexity at the expense of a larger minimum degree. For example, it works in time $2^{o(n)}$ when the minimum degree is $\omega(\log n)$. This suggests the following problem:

Problem 16 *Let c be any sufficiently large constant. Does there exist a $2^{o(n)}$-time* deterministic *algorithm that, given an n-vertex Hamiltonian graph G of minimum degree at least* $\log^c n$ *and a Hamiltonian cycle C in G, computes a Hamiltonian cycle in G that is different from C?*

References

1. Abbasi, S., Jamshed, A.: A degree constraint for uniquely Hamiltonian graphs. Graphs Comb. **22**, 433–442 (2006)
2. Bazgan, C., Santha, M., Tuza, Z.: On the approximation of finding a(nother) Hamiltonian cycle in cubic Hamiltonian graphs. J. Algorithms **31**(1), 249–268 (1999)
3. Bellman, R.: Dynamic programming treatment of the travelling salesman problem. J. ACM **9**(1), 61–63 (1962)
4. Björklund, A.: Determinant sums for undirected hamiltonicity. SIAM J. Comput. **43**(1), 280–299 (2014)
5. Björklund, A., Husfeldt, T., Kaski, P., Koivisto, M.: The traveling salesman problem in bounded degree graphs. ACM Trans. Algorithms **8**(2), 18:1-18:13 (2012)
6. Bodlaender, H.L., Cygan, M., Kratsch, S., Nederlof, J.: Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. Inf. Comput. **243**, 86–111 (2015)
7. Bodlaender, H.L., Cygan, M., Kratsch, S., Nederlof, J.: Deterministic single exponential time algorithms for connectivity problems parameterized by treewidth. Inf. Comput. **243**, 86–111 (2015)
8. Bondy, J.A., Jackson, B.: Vertices of small degree in uniquely Hamiltonian graphs. J. Comb. Theory Ser. B **74**(2), 265–275 (1998)
9. Cameron, K.: Thomason's algorithm for finding a second Hamiltonian circuit through a given edge in a cubic graph is exponential on Krawczyk's graphs. Discrete Math. **235**, 69–77 (2001)
10. Cygan, M., Kratsch, S., Nederlof, J.: Fast hamiltonicity checking via bases of perfect matchings. In: Proceedings of the 45th ACM Symposium on Theory of Computing Conference (STOC), pp. 301–310 (2013)
11. Cygan, M., Kratsch, S., Nederlof, J.: Fast hamiltonicity checking via bases of perfect matchings. J. ACM (JACM) **65**(3), 1–46 (2018)
12. Deligkas, A., Mertzios, G.B., Spirakis, P.G., Zamaraev, V.: Exact and approximate algorithms for computing a second Hamiltonian cycle. In: Proceedings of the 45th International Symposium on Mathematical Foundations of Computer Science (MFCS), vol. 170, pp. 27:1–27:13 (2020)
13. Entringer, R.C., Swart, H.: Spanning cycles of nearly cubic graphs. J. Comb. Theory Ser. B **29**(3), 303–309 (1980)
14. Eppstein, D.: The traveling salesman problem for cubic graphs. J. Graph Algorithms Appl. **11**(1), 61–81 (2007)

15. Fleischner, H.: Uniquely Hamiltonian graphs of minimum degree 4. J. Graph Theory **75**(2), 167–177 (2014)
16. Girão, A., Kittipassorn, T., Narayanan, B.: Long cycles in Hamiltonian graphs. Isr. J. Math. **229**(1), 269–285 (2019)
17. Haxell, P., Seamone, B., Verstraete, J.: Independent dominating sets and Hamiltonian cycles. J. Graph Theory **54**(3), 233–244 (2007)
18. Held, M., Karp, R.M.: A dynamic programming approach to sequencing problems. J. Soc. Ind. Appl. Math. **10**(1), 196–210 (1962)
19. Liśkiewicz, M., Schuster, M.R.: A new upper bound for the traveling salesman problem in cubic graphs. J. Discrete Algorithms **27**, 1–20 (2014)
20. Papadimitriou, C.H.: On the complexity of the parity argument and other inefficient proofs of existence. J. Comput. Syst. Sci. **48**(3), 498–532 (1994)
21. Sheehan, J.: The multiplicity of Hamiltonian circuits in a graph. In: Recent Advances in Graph Theory, pp. 477–480 (1975)
22. Thomason, A.G.: Hamiltonian cycles and uniquely edge colourable graphs. Ann. Discrete Math. **3**, 259–268 (1978)
23. Thomason, A.G.: Hamiltonian cycles and uniquely edge colourable graphs. Adv. Graph Theory **3**, 259–268 (1978)
24. Thomassen, C.: Chords of longest cycles in cubic graphs. J. Comb. Theory Ser. B **71**(2), 211–214 (1997)
25. Thomassen, C.: Independent dominating sets and a second Hamiltonian cycle in regular graphs. J. Comb. Theory Ser. B **72**(1), 104–109 (1998)
26. Tutte, W.T.: On Hamiltonian circuits. J. Lond. Math. Soc. **1**(2), 98–101 (1946)
27. West, D.B.: Introduction to Graph Theory, 2nd edn. Prentice Hall, Upper Saddle River (2001)
28. Xiao, M., Nagamochi, H.: An exact algorithm for TSP in degree-3 graphs via circuit procedure and amortization on connectivity structure. Algorithmica **74**(2), 713–741 (2016)