Long cycles in graphs:
Extremal Combinatorics meets Parameterized Algorithms

Fedor V. Fomin
Longest Cycle: Given a graph $G$ and integer $k$, decide whether $G$ contains a cycle of length at least $k$?
Historical notes

Sir William Rowan Hamilton
(1805 - 1865)

Rudrata (9th century)

Al-Ádlí ar-Rúmí (9th century)
Historical notes

Extremal combinatorics

Extremal combinatorics is a field of combinatorics, which is itself a part of mathematics. Extremal combinatorics studies how large or how small a collection of finite objects (numbers, graphs, vectors, sets, etc.) can be, if it has to satisfy certain restrictions.

How dense should be a graph to be Hamiltonian?
To contain a long cycle?
Dirac’s Theorem

Theorem (Dirac, 1952)

Every \( n \)-vertex 2-connected graph \( G \) with minimum vertex degree \( \delta(G) \geq 2 \), contains a cycle with at least \( \min\{2\delta(G), n\} \) vertices.

**SOME THEOREMS ON ABSTRACT GRAPHS**

By G. A. Dirac

[Received 4 April 1951.—Read 19 April 1951]

A graph is a set \( \mathcal{N} \) whose members are called the nodes together with a set \( \mathcal{E} \) of unordered pairs of unequal members of \( \mathcal{N} \) called the edges. In this paper nodes will generally be denoted by small letters \( a, b, \) etc., possibly with suffixes, and edges by \( (a, b) \), etc., where \( a \neq b \) and \( (a, b) = (b, a) \). Each of the nodes \( a \) and \( b \) is called an end node of the edge \( (a, b) \) and these two nodes are said to be joined by the edge or to be adjacent to the edge. A graph is finite if the set of its nodes is finite, otherwise it is infinite. The order of a graph is the (cardinal) number of the set of its nodes. A subgraph is a graph whose sets of nodes and edges are subsets of the sets of nodes and edges of the graph. The number of edges adjacent to a node is called the degree of the node.

A path is a graph whose nodes are \( a_1, a_2, a_3, \ldots, a_n \), where \( n \geq 2 \) and different suffixes denote different nodes, and whose edges are \( (a_1, a_2), (a_2, a_3), \ldots, (a_{n-1}, a_n) \). A circuit is a graph whose nodes are \( a_1, a_2, a_3, \ldots, a_m, \) where \( m \geq 3 \) and different suffixes denote different nodes, and whose edges are \( (a_1, a_2), (a_2, a_3), \ldots, (a_{m-1}, a_m), (a_m, a_1) \). The length of a path (circuit) is the number of edges in the path (circuit).
Graph Theory Hall of Fame

Gabriel Andrew Dirac
Paul Erdős
Tibor Gallai
Øystein Ore
Lajos Pósa
Václav Chvátal
Crispin Nash-Williams
Adrian Bondy
Hassler Whitney
William Tutte
Parameterized Algorithms

- Monien [1982], $k^k \cdot n^{o(1)}$
  representative sets

- Bodlaender [1984]: $k^k \cdot n^{o(1)}$
  treewidth

- Papadimitriou and Yannakakis [1996]: Is in $P$ for $k=\log n$?
Color-Coding

NOGA ALON

Institute for Advanced Study, Princeton, New Jersey and Tel-Aviv University, Tel-Aviv, Israel

RAPHAEL YUSTER AND URI ZWICK

Tel-Aviv University, Tel-Aviv, Israel

Abstract. We describe a novel randomized method, the method of color-coding for finding simple paths and cycles of a specified length $k$, and other small subgraphs, within a given graph $G = (V, E)$. The randomized algorithms obtained using this method can be derandomized using families of perfect hash functions. Using the color-coding method we obtain, in particular, the following new results:

— For every fixed $k$, if a graph $G = (V, E)$ contains a simple cycle of size exactly $k$, then such a cycle can be found in either $O(V^\omega)$ expected time or $O(V^\omega \log V)$ worst-case time, where $\omega < 2.376$ is the exponent of matrix multiplication. (Here and in what follows we use $V$ and $E$ instead of $|V|$ and $|E|$ whenever no confusion may arise.)

— For every fixed $k$, if a planar graph $G = (V, E)$ contains a simple cycle of size exactly $k$, then such a cycle can be found in either $O(V)$ expected time or $O(V \log V)$ worst-case time. The same algorithm applies, in fact, not only to planar graphs, but to any minor closed family of graphs which is not the family of all graphs.

— If a graph $G = (V, E)$ contains a subgraph isomorphic to a bounded tree-width graph $H = (V_H, E_H)$ where $|V_H| = O(\log V)$, then such a copy of $H$ can be found in polynomial time. This
Test bed for new methods

Determinant-sum

Algebraic fingerprints

Treewidth algorithms

Cut & count

Representative sets

Divide-and-color

Narrow sieves

Polynomial differentiation
Could the density of a graph be helpful in finding long cycles?
Algorithmic question

Theorem (Dirac, 1952)

Every $n$-vertex 2-connected graph $G$ with minimum vertex degree $\delta(G) \geq 2$, contains a cycle with at least $\min\{2\delta(G), n\}$ vertices.

'Naive' question

Is there a polynomial time algorithm to decide whether a 2-connected graph $G$ contains a cycle of length at least $\min\{2\delta(G) + 1, n\}$?
Algorithmic question

Dirac bound

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

Above Dirac bound

Does a 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\} + k$?
Remark: Why 2-connectivity is important

Let $G$ be an $n$-vertex graph.

- $H$ has a cycle of length $2d = n$ iff $G$ is Hamiltonian.

- Cliques of size $n/2$.
Algorithm that in time $2^{O(k)} n^{O(1)}$ decides whether a 2-connected graph with minimum degree $d$ contains a cycle of length at least $\min\{2d, n\} + k$. 

Abstract

In 1952, Dirac proved the following theorem about long cycles in graphs with large minimum vertex degrees: Every $n$-vertex 2-connected graph $G$ with minimum vertex degree $\delta \geq 2$ contains a cycle with at least $\min\{2\delta, n\}$ vertices. In particular, if $\delta \geq n/2$, then $G$ is Hamiltonian. The proof of Dirac’s theorem is constructive, and it yields an algorithm computing the corresponding cycle in polynomial time. The combinatorial bound of Dirac’s theorem is tight in the following sense. There are 2-connected graphs that do not contain cycles of length more than $2\delta + 1$. Also, there are non-Hamiltonian graphs with all vertices but one of degree at least $n/2$. This prompts naturally to the following algorithmic questions. For $k \geq 1$,

(A) How difficult is to decide whether a 2-connected graph contains a cycle of length at least $\min\{2\delta + k, n\}$?

(B) How difficult is to decide whether a graph $G$ is Hamiltonian, when at least $n - k$ vertices of $G$ are of degrees at least $n/2 - k$?

The first question was asked by Fomin, Golovach, Lokshtanov, Panolan, Saurabh, and Zehavi. The second question is due to Jansen, Kozma, and Nederlof. Even for a very special case of $k = 1$, the existence of a polynomial-time algorithm deciding whether $G$ contains a cycle of length at least $\min\{2\delta + 1, n\}$ was open. We resolve both questions by proving the following
Theorem. Longest Cycle above Dirac’s bound is FPT.

**Dirac:**

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

**How to construct a cycle of length $2d$ in polynomial time?**
Dirac’s Theorem (1952)

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

Proof:

Interesting part when $2d < n$

Take a cycle $C$, $|C| \leq 2d$

H:
Dirac’s Theorem (1952)

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

Proof:

$|C| < 2d$

Min-degree $d$
Dirac’s Theorem (1952)

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

Proof:

$H = \begin{array}{c}
\text{Min-degree } d
\end{array}$
Dirac’s Theorem (1952)

Every $n$-vertex 2-connected graph with minimum vertex degree $d$ contains a cycle of length at least $\min\{2d, n\}$

**Proof:**

Erdős-Gallai Lemma (1959)

A 2-connected graph of minimum vertex degree $d$ contains a path of length at least $d$ between any given pair of vertices.
Dirac’s Theorem (1952)

Every \( n \)-vertex 2-connected graph with minimum vertex degree \( d \) contains a cycle of length at least \( \min\{2d, n\} \)

Proof:

\[ |C| < 2d \]

Apply EG-lemma + count carefully
Dirac’s Theorem - how to proceed algorithmically?

Simplified question: we have a cycle $C$ of length $2d$, decide whether it is possible to enlarge it.

$|C| = 2d$

$v$ adjacent to every second vertex of $C$
Dirac’s Theorem - how to proceed algorithmically?

Simplified question: we have a cycle $C$ of length $2d$, decide whether it is possible to enlarge it.

- $v$ adjacent to every second vertex of $C$
- every other $u$ outside $C$ is adjacent exactly the same vertices
Dirac’s Theorem – how to proceed algorithmically?

Simplified question: we have a cycle $C$ of length $2d$, decide whether it is possible to enlarge it.

- $|C| = 2d$

$H = \circ$

- $v$ adjacent to every second vertex of $C$
- every other $u$ outside $C$ is adjacent to exactly the same vertices
- $N(v)$ is a vertex cover of $G$ of size $d$

Intuition: We either could enlarge the cycle or construct a “small” vertex cover
More generally (informally)

We try to enlarge a cycle $C$ of length $2d+p$.
Figure 9: A schematic example of a Dirac decomposition, vertices belonging to $B$ are in light gray.

Removing the paths $P_1$ and $P_2$ leaves two (D1)-type components that correspond to the long arcs $P_0$ and $P_{00}$ of the starting cycle $C$, one (D2)-type component, and a component consisting only of vertices from $B$, denoted by $D_0$. The four Dirac components are in thick blue.

- Let $G_0$ be the graph obtained from $G$ by applying $B$-refinement to every connected component $H$ of $G$, except those components $H$ with $V(H) \not\subseteq B$. Note that no edges of the paths $P_1$ and $P_2$ are contracted. Then for every connected component $H_0$ of $G_0$ with $V(H_0) \not\subseteq B$, holds $|V(H_0)| \geq 3$ and one of the following.

1. $H_0$ is 2-connected and the maximum size of a matching in $G_0$ between $V(H_0)$ and $V(P_1)$ is one, and between $V(H_0)$ and $V(P_2)$ is also one;
2. $H_0$ is not 2-connected, exactly one vertex of $P_1$ has neighbors in $H_0$, that is, $|N_{G_0}(V(H_0)) \setminus V(P_1)| = 1$, and no inner vertex from a leaf-block of $H_0$ has a neighbor in $P_2$;
3. The same as (D2), but with $P_1$ and $P_2$ interchanged. That is, $H_0$ is not 2-connected, $|N_{G_0}(V(H_0)) \setminus V(P_2)| = 1$, and no inner vertex from a leaf-block of $H_0$ has a neighbor in $P_1$.

- There is exactly one connected component $H$ in $G$ with $V(H) \cap B = V(P_0) \cap (B \setminus \{s_0, t_0\})$, where $s_0$ and $t_0$ are the endpoints of $P_0$. Analogously, there is exactly one connected component $H$ in $G$ with $V(H) \cap B = V(P_{00}) \cap (B \setminus \{s_{00}, t_{00}\})$.

The set of Dirac components for a Dirac decomposition is defined as follows. First, for each component $H_0$ of type (D1), $H_0$ is a Dirac component of the Dirac decomposition. Second, for each leaf-block of each $H_0$ of type (D2), or of type (D3), this leaf-block is also a Dirac component of the Dirac decomposition. For an example of a Dirac decomposition, see Figure 9.

Note that Lemma 5 holds for an arbitrary cycle $C$ if we replace Erdős-Gallai components and Erdős-Gallai decompositions with Dirac components and Dirac decompositions. We give the analogue of this lemma below without proof, since it is identical to the proof of Lemma 5.
Let $G$ be a 3-connected graph and $k$ be an integer such that $0 < k \leq \frac{1}{24}d$. Then there is an algorithm that, given a cycle $C$ of length less than $2d + k < n$, in polynomial time either

- returns a longer cycle in $G$, or
- returns a vertex cover of $G$ of size at most $d + 2k$

Does it solve the problem?
Let $G$ be a 3-connected graph and $k$ be an integer such that $0 < k \leq \frac{1}{24}d$. Then there is an algorithm that, given a cycle $C$ of length less than $2d + k < n$, in polynomial time either

- returns a longer cycle in $G$, or
- returns a vertex cover of $G$ of size at most $d + 2k$

The algorithm that in a graph $G$ with a vertex cover of size $d + 2k$ decides whether $G$ has a cycle of length at least $2d + k$ in time $2^{O(k)}$

Well-known: If the vertex cover of $G$ is at most $p$ then a longest cycle in $G$ could be found in time $2^{O(p)} \cdot n^{O(1)}$

A cycle of length at least $2d + k$ in time $2^{O(d+k)} \cdot n^{O(1)}$

Not what we shooting for!
Let $G$ be a 3-connected graph and $k$ be an integer such that $0 < k \leq \frac{1}{24}d$. Then there is an algorithm that, given a cycle $C$ of length less than $2d + k < n$, in polynomial time either

- returns a longer cycle in $G$, or
- returns a vertex cover of $G$ of size at most $d + 2k$

The algorithm that in a graph $G$ with a vertex cover of size $d + p$, $p < d/2$, decides whether $G$ has a cycle of length at least $2d + k$ in time $2^{O(p)} \cdot n^{O(1)}$

**Idea**: Every cycle of length at least $2d + k$

could be rerouted to make a new cycle with very specific properties.

It allows reducing the problem of finding a cycle to the problem of covering vertices in a subgraph by paths of total length $O(p)$. 
Let $G$ be a 3-connected graph and $k$ be an integer such that $0 < k \leq \frac{1}{24}d$. Then there is an algorithm that, given a cycle $C$ of length less than $2d + k < n$, in polynomial time either

- returns a longer cycle in $G$, or
- returns a vertex cover of $G$ of size at most $d + 2k$

We need algorithmic EG-Lemma: an algorithm that for any $s,t$ decides whether there is an $(s,t)$-path of length at least $d+k$ in time $2^{O(k)} \cdot n^{O(1)}$

New extremal properties of cycles that "cannot be enlarged" in "Erdős-Gallai" and "Dirac" way:
Erdős-Gallai decomposition and Dirac decomposition

An interesting interplay between parameterized algorithms and graph structure
Theorem [Longest Cycle above Dirac’s bound]

There is an algorithm deciding whether a 2-connected graph $G$ with minimum degree $d$ has a cycle of length at least $2d+1$ in time $2^{O(k)} \cdot n^{O(1)}$. 

---

Figure: The main steps and connections in the proof of Theorem 3.

---

Theorem 5.

The proof of this result is constructive, and it implies a polynomial time algorithm that finds such a path. We define a pair of vertices that are shortest disjoint paths avoiding $(i)$ the total length of the $(ii)$ either $(iii)$ or the length of the $(ii)$ either $(iii)$.

For every classical theorem due to Erdős and Gallai from [EG59, Theorem 1.16], see also [Loc85]. For every of the last min the method of random separation to distinguish the following three sets: (i) the total length of the $(ii)$ either $(iii)$.

First, we use color coding to verify whether the considered instance has a solution composed by two $(ii)$ either $(iii)$.

Second, we use color coding to verify whether the considered instance has a solution composed by a 2-connected graph with vertex set $V$.

Almost Hamiltonian Dirac Cycle (Theorem 7)

Almost Hamiltonian Dirac Cycle (Theorem 7)

Long Dirac Cycle (Theorem 3)

Long Dirac Cycle (Theorem 3)

Long Erdős-Gallai (s, t)-Path (Theorem 5)

Long Erdős-Gallai (s, t)-Path (Theorem 5)

Long (s, t)-Cycle (Theorem 4)

Long (s, t)-Cycle (Theorem 4)
If every vertex of an $n$-vertex graph $G$ is of degree at least $n/2$, then $G$ is Hamiltonian, that is, contains a Hamiltonian cycle.

**Theorem [Jansen, Kozma, Nederlof]**

**Abstract**

Dirac’s theorem (1952) is a classical result of graph theory, stating that an $n$-vertex graph ($n \geq 3$) is Hamiltonian if every vertex has degree at least $n/2$. Both the value $n/2$ and the requirement for every vertex to have high degree are necessary for the theorem to hold.

In this work we give efficient algorithms for determining Hamiltonicity when either of the two conditions are relaxed. More precisely, we show that the Hamiltonian cycle problem can be solved in time $c^k \cdot n^{O(1)}$, for some fixed constant $c$, if at least $n - k$ vertices have degree at least $n/2$, or if all vertices have degree at least $n/2 - k$. The running time is, in both cases, asymptotically optimal under the Exponential-Time Hypothesis (ETH).

The results extend the range of tractability of the Hamiltonian cycle problem, showing that it is fixed-parameter tractable when parameterized below a natural bound. In addition, for the first parameterization we show that a kernel with $O(k)$ vertices can be found in polynomial time.
**Long Dirac Cycle** parameterized by $k + |B|$

*Input:* Graph $G$ with vertex set $B \subseteq V(G)$ and integer $k \geq 0$.

*Task:* Decide whether $G$ contains a cycle of length at least $\min\{2\delta(G - B), |V(G)| - |B|\} + k$.

---

**Theorem [FF, Golovach, Sagunov, Simonov]**

**Long Dirac Cycle** on 2-connected graphs is solvable in time $2^{O(k + |B|)} \cdot n^{O(1)}$. 
How useful is Dirac decomposition for other problems?
Theorem (Erdős-Gallai, 1959)

Every graph with $n$ vertices and more than $(n - 1)\ell/2$ edges ($\ell \geq 2$) contains a cycle of length at least $\ell + 1$.

ON MAXIMAL PATHS AND CIRCuits OF GRAPHS

By

P. ERDŐS (Budapest), corresponding member of the Academy,
and T. GALLAI (Budapest)

Introduction

In 1940 TURÁN raised the following question: if the number of nodes, $n$, of a graph is prescribed and if $\ell$ is an integer $\leq n$, what is the number of edges which the graph has to contain in order to ensure that it necessarily contains a complete $\ell$-graph? TURÁN gave a precise answer to this question by determining the smallest number depending on $n$ and $\ell$, with the property that a graph with $n$ nodes and with more edges than this number necessarily contains a complete $\ell$-graph ([9], [10]). More generally, the question can be posed, as was done by TURÁN; given a graph with a
**Theorem (Erdős-Gallai, 1959)**

Every graph with \( n \) vertices and more than \((n - 1)\ell/2\) edges (\( \ell \geq 2 \)) contains a cycle of length at least \( \ell + 1 \).

**In other words**

Every graph with \( n \) vertices and \( m \) edges contains a cycle of length at least \( \frac{2m}{n-1} \).

Every graph contains a cycle of length at least its average degree \( D = \frac{2m}{n} \).

\[
\frac{2m}{n-1} - 1 \leq D \leq \frac{2m}{n-1}
\]
<table>
<thead>
<tr>
<th>Erdős-Gallai bound</th>
<th>Above Erdős-Gallai bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Every 2-connected graph of average vertex degree $D &gt; 2$ contains a cycle of length at least $D$</td>
<td>Does a 2-connected graph of average vertex degree $D &gt; 2$ contain a cycle of length at least $D + k$?</td>
</tr>
</tbody>
</table>
Theorem. Longest Cycle above Erdős–Gallai's bound is FPT.

Algorithm that in time $2^{O(k)} n^{O(1)}$ decides whether a 2-connected graph with average vertex degree $D$ contains a cycle of length at least $D + k$. 

Longest Cycle above Erdős–Gallai Bound*

Fedor V. Fomin†  Petr A. Golovach†  Danil Sagunov†§  Kirill Simonov¶

fedor.fomin@uib.no  petr.golovach@uib.no  danilka.pro@gmail.com  kirillsimonov@gmail.com

Abstract

In 1959, Erdős and Gallai proved that every graph $G$ with average vertex degree $\text{ad}(G) \geq 2$ contains a cycle of length at least $\text{ad}(G)$. We provide an algorithm that for $k \geq 0$ in time $2^{O(k)} n^{O(1)}$ decides whether a 2-connected $n$-vertex graph $G$ contains a cycle of length at least $\text{ad}(G) + k$. This resolves an open problem explicitly mentioned in several papers. The main ingredients of our algorithm are new graph-theoretical results interesting on their own.

Keywords: Longest path, longest cycle, fixed-parameter tractability, above guarantee parameterization, average degree, dense graph, Erdős and Gallai theorem

1 Introduction
GOING FAR FROM DEGENERACY*

FEDOR V. FOMIN†, PETR A. GOLOVACH†, DANIEL LOKSHTANOV‡, FAHAD PANOLAN§, SAKET SAURABH‡, AND MEIRAV ZEHAVI‖

Abstract. An undirected graph $G$ is $d$-degenerate if every subgraph of $G$ has a vertex of degree at most $d$. By the classical theorem of Erdős and Gallai from 1959, every graph of degeneracy $d > 1$ contains a cycle of length at least $d + 1$. The proof of Erdős and Gallai is constructive and can be turned into a polynomial time algorithm constructing a cycle of length at least $d + 1$. But can we decide in polynomial time whether a graph contains a cycle of length at least $d + 2$? An easy reduction from HAMILTONIAN CYCLE provides a negative answer to this question: Deciding whether a graph has a cycle of length at least $d + 2$ is NP-complete. Surprisingly, the complexity of the problem changes drastically when the input graph is 2-connected. In this case we prove that deciding whether $G$ contains a cycle of length at least $d + k$ can be done in time $2^{O(k)} \cdot |V(G)|^{O(1)}$. In other words, deciding whether a 2-connected $n$-vertex $G$ contains a cycle of length at least $d + \log n$ can be done in polynomial time. Similar algorithmic results hold for long paths in graphs. We observe that deciding whether a graph has a path of length at least $d + 1$ is NP-complete. However, we prove that if graph $G$ is connected, then deciding whether $G$ contains a path of length at least $d + k$ can be done in time $2^{O(k)} \cdot n^{O(1)}$. We complement these results by showing that the choice of degeneracy as the “above guarantee parameterization” is optimal in the following sense: For any $\varepsilon > 0$ it is NP-complete to decide whether a connected (2-connected) graph of degeneracy $d$ has a path (cycle) of length at least $d + \varepsilon$.
Cycle of length at least $\text{maximum average-degree}(G)+k$

Cycle of length at least $\text{average-degree}(G)+k$

Cycle of length at least $\text{degeneracy}(G)+k$

[FF, Golovach, Lokshtanov, Panolan, Saurabh, Zehavi, 2020]
General idea

Identify very dense graph $H$

Color-coding for outside part
How to identify a dense component?

After some preprocessing, an old friend, Dirac's decomposition comes to help.

\[ O(k) \] vertices in the vertex cover and select a subset of the independent set to achieve the property that (i) each of remaining vertices in the vertex cover is adjacent to at least \( ad(H) \) vertices in the selected independent subset, and (ii) every vertex of the selected subset of the independent set sees nearly all vertices of the vertex cover. This means that the obtained subgraph is also "dense", albeit in a different sense. Depending on the case, we use different arguments to establish the routing properties of \( H \).

Routing in \( H \). The case (a), when \( |V(H)| < ad(H) + k \), is easier. In this case, the degrees of almost all vertices are close to \( |V(H)| \). Let \( S = \{ x_1y_1, \ldots, x_ky_k \} \) be an arbitrary set of \( O(k) \) pairs of distinct vertices of \( H \) forming a linear forest (that is, the union of \( x_iy_i \) is a union of disjoint paths). The intuition behind \( S \) is that \( x_i \) corresponds to the vertex from where the long cycle leaves \( H \) and \( y_i \) when it enters \( H \) again. We show first how to construct a cycle in \( H + S \) (that is, the graph obtained from \( H \) by turning the pairs of \( S \) into edges) containing every pair \( x_iy_i \) from \( S \) as an edge. This is done by performing constant-length jumps: any two vertices can be connected either by an edge, or through a common neighbor, or through a sequence of two neighbors. Then we extend the obtained cycle to a Hamiltonian cycle in \( H + S \) — every vertex of \( H \) that is not yet on a cycle can be inserted due to the high degrees of the vertices. The extension of \( S \) into a Hamiltonian cycle is shown in Figure 2 (a).

![Figure 2](image-url)

Figure 2: Constructing cycles. The set of pairs \( S \) that may be both edges and nonedges of \( H \) is shown by red lines and the extension of \( S \) into a long cycle is blue. The paths "revolving" around \( H \) are green. The vertex cover in c) is denoted by \( A \).

Therefore, if there is a collection of at most \( k \) internally vertex disjoint paths going outside from \( H \) and returning back, the high density of \( H \) allows collecting all of them in a cycle containing all the vertices of \( H \). Together with all the additional vertices these paths visit outside of \( H \) we construct a long cycle in \( G \) (see Figure 2 (b)). The only condition is that these paths have to form a linear forest. Thus, if we find a collection of such paths with enough internal vertices, we immediately obtain a long cycle "revolving" around \( H \). The crucial part of the proof is to show that if there is any cycle of length at least \( ad(H) + k \) in \( G \), then it can be assumed to have this form.

Let us remark that a similar "rerouting" property was used by Fomin et al. [7] in their above-degeneracy study. Actually, for case (a), we need only a minor adjustment of the arguments from [7]. However, in the "bipartite dense" case (b) the structure of the dense subgraph \( H \) is more elaborate and this case requires a new approach. Contrarily to case (a), the long cycle that we construct in \( H + S \) is not Hamiltonian but visits all the vertices of the vertex cover. (See Figure 2 (c).) In this case, the behavior of paths depends on which part of \( H \) they hit. Because of that, while establishing the routing properties, we have to take into account the difference between paths connecting vertices from the vertex cover, independent set, and both. Pushing the "rerouting" intuition through, in this case, turns out to be quite challenging.

Final steps. After finalizing the "rerouting" arguments above, it only remains to design an algorithm that checks whether there exists a collection of paths in \( G \) that start and end in \( H \).
Conclusion
Open questions

**Theorem (Thomassen 1981)**

Let $D$ be a 2-connected digraph with at least $2d + 1$ vertices such that $d^-(v) \geq d$ and $d^+(v) \geq d$ for every $d \in V(D)$. Then $D$ contains a cycle of length at least $2d$.

Is there a polynomial-time algorithm deciding whether there is a cycle of length at least $2d+1$?

Is there an XP algorithm deciding whether there is a cycle of length at least $2d+k$?

Is there an FPT algorithm deciding whether there is a cycle of length at least $2d+k$?
Vassily Kandinsky, Composition X, 1939