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*?

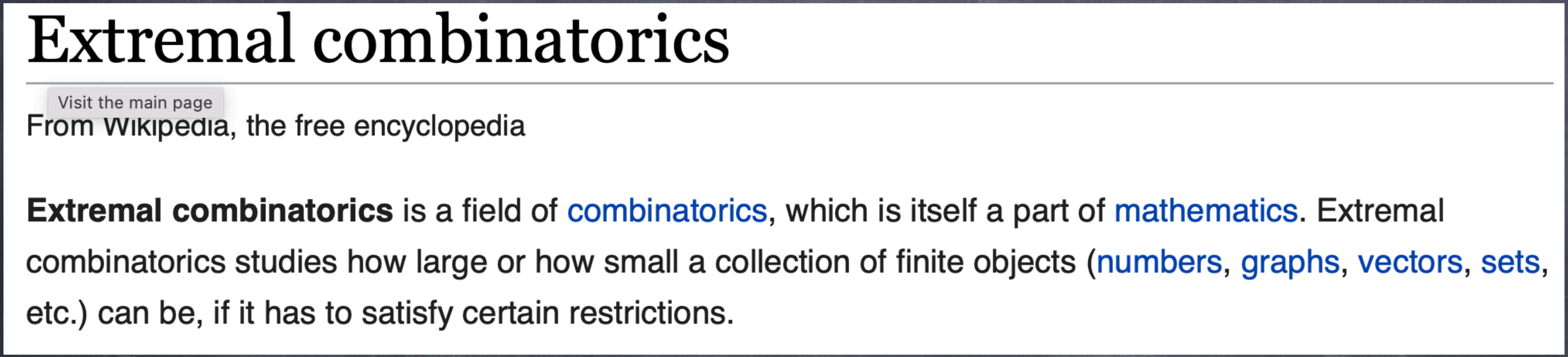
Historical notes

Sir William Rowan Hamilton
(1805 – 1865)

Rudrata (9th century)

Al-Ádlí ar-Rúmí (9th century)

Historical notes


Extremal combinatorics

Visit the main page

From Wikipedia, the free encyclopedia

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 $\mathscr{N}$ whose members are called the *nodes* together with a set $\mathscr{E}$ of unordered pairs of unequal members of $\mathscr{N}$ 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 *adjoint* 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 adjoint to a node is called the *degree* of the node.

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

Gabriel Andrew Dirac

Graph Theory Hall of Fame

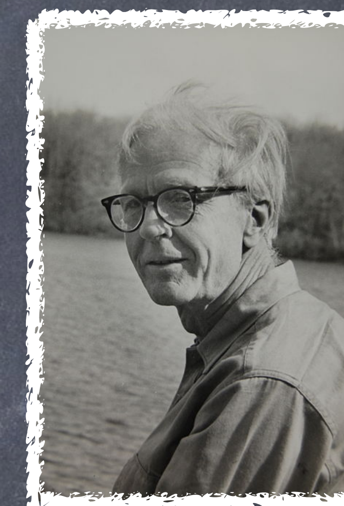
Gabriel Andrew Dirac

Paul Erdős

Tibor Gallai

Lajos Pósa

Øystein Ore

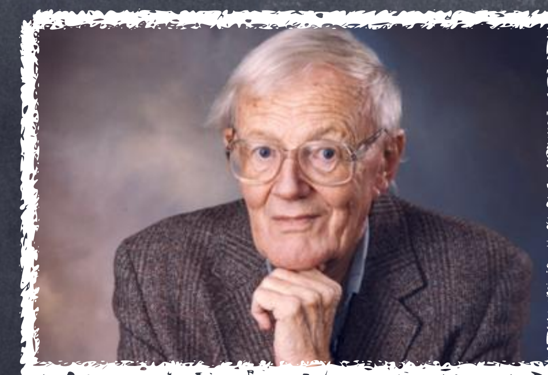


Hassler Whitney

Václav Chvátal

Crispin Nash-Williams

Adrian Bondy



William Tutte

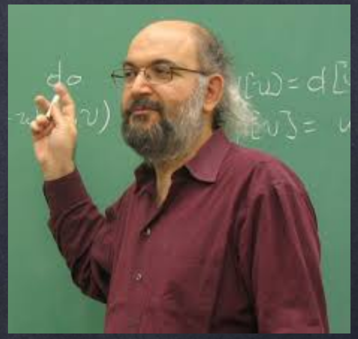
Parameterized Algorithms

- 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 $k = \log n$?

Burkhard Monien

Hans Bodlaender

Christos Papadimitriou



Mihalis Yannakakis

Color Coding [1995] $O(2^{O(k)} \cdot n)$

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


Algebraic fingerprints



Determinant-sum

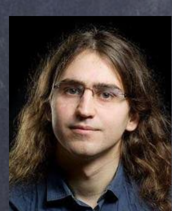


Treewidth algorithms


Representative sets

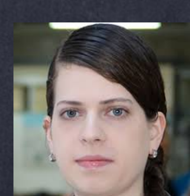

Divide-and-color





Cut & count


Narrow sieves




Polynomial differentiation

General question

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 $\min\{2\delta(G) + 1, n\}$?


Algebraic fingerprints
Determinant-sum
Treewidth algorithms
Divide-and-color
Representative sets
& count
Narrow sieves
Polynomial differentiation

Algorithmic question

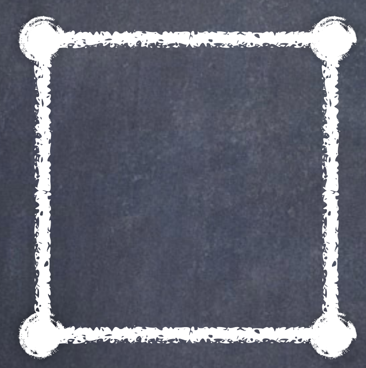
Dirac bound

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

Above Dirac bound

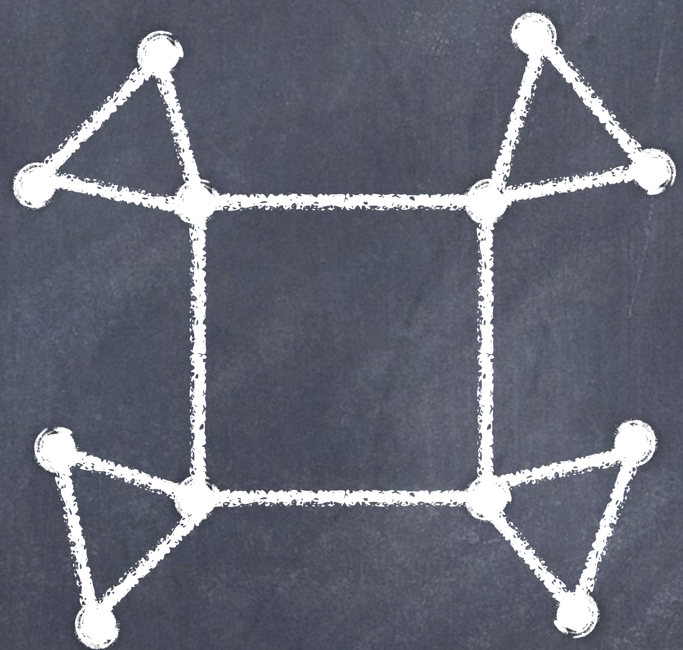
Does a 2-connected graph with minimum vertex degree d contains a cycle of length at least $\min\{2d, n\} + k$?

Remark: Why 2-connectivity is important



G: n-vertex graph

cliques of size $n/2$



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

Theorem. Longest Cycle above Dirac's bound is FPT.

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 $\min\{2d, n\} + k$.

Algorithmic Extensions of Dirac's Theorem

Fedor V. Fomin* fomin@ii.uib.no

Petr A. Golovach* petr.golovach@ii.uib.no

Danil Sagunov† danilka.pro@gmail.com

Kirill Simonov* kirillsimonov@gmail.com

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?

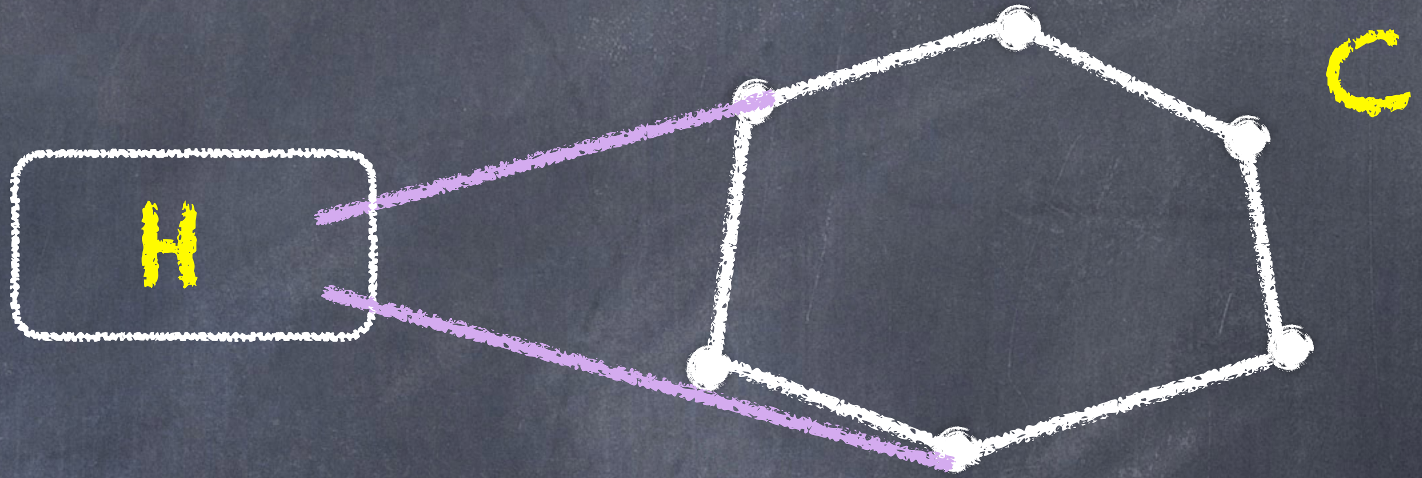
Dirac's Theorem (1952)

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

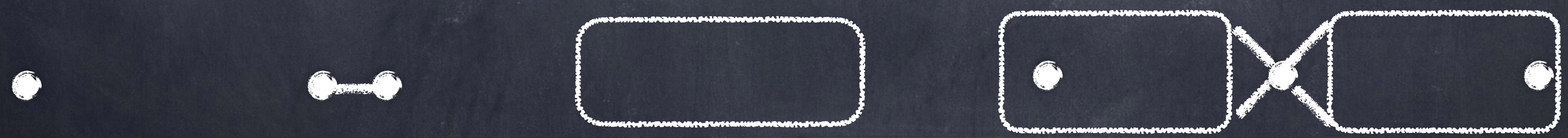
Proof:

Interesting part when $2d<n$

Take a cycle C. $|C|<2d$



H:



Dirac's Theorem (1952)

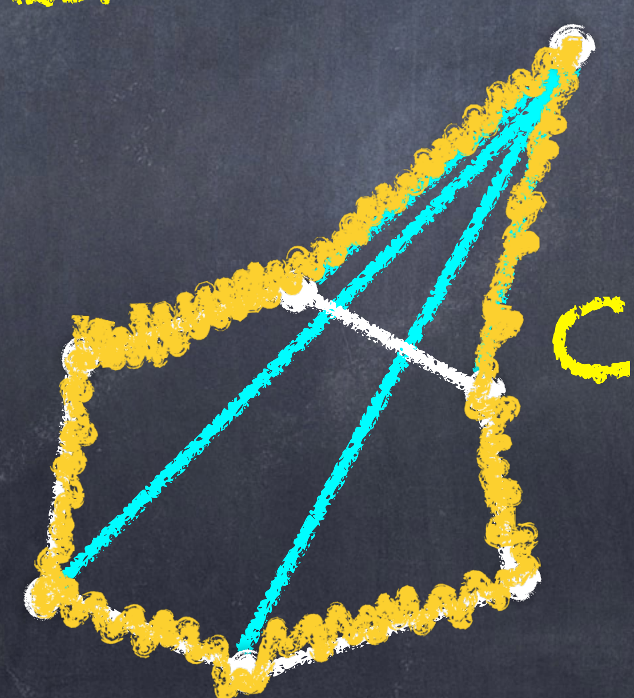
Every n-vertex 2-connected graph with minimum vertex degree d contains a cycle of length at least $\min\{2d, n\}$

Proof:

H= •

$|C|<2d$

Min-degree d



Dirac's Theorem (1952)

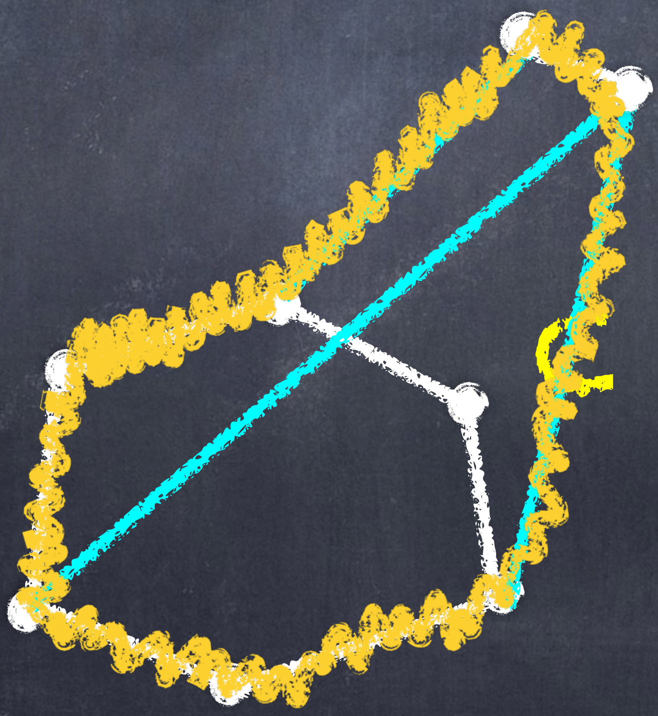
Every n-vertex 2-connected graph with minimum vertex degree d contains a cycle of length at least $\min\{2d, n\}$

Proof:

H=

|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=

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:

H= 

$|C|<2d$

Min-degree d

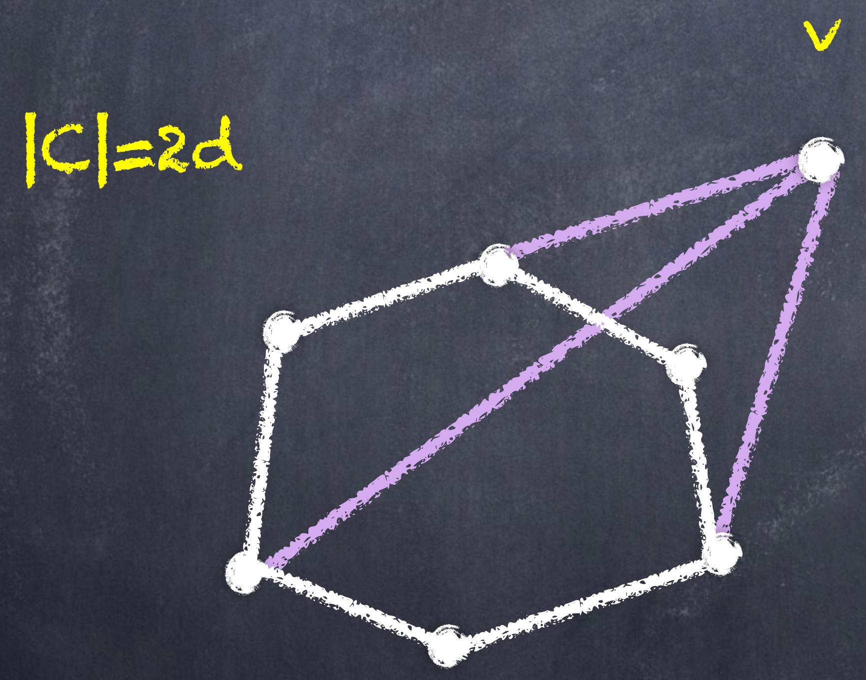
Apply EG-lemma + count carefully

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.

H= 

- v adjacent to every second vertex of C



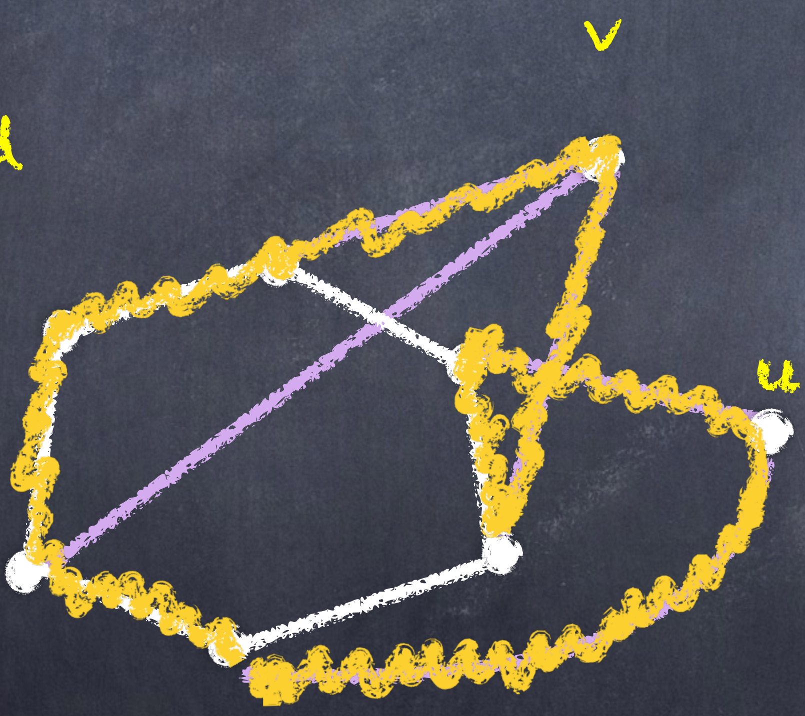
Dirac's Theorem - how to proceed algorithmically?

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

H=

- v adjacent to every second vertex of C
- every other u outside C is adjacent exactly the same vertices

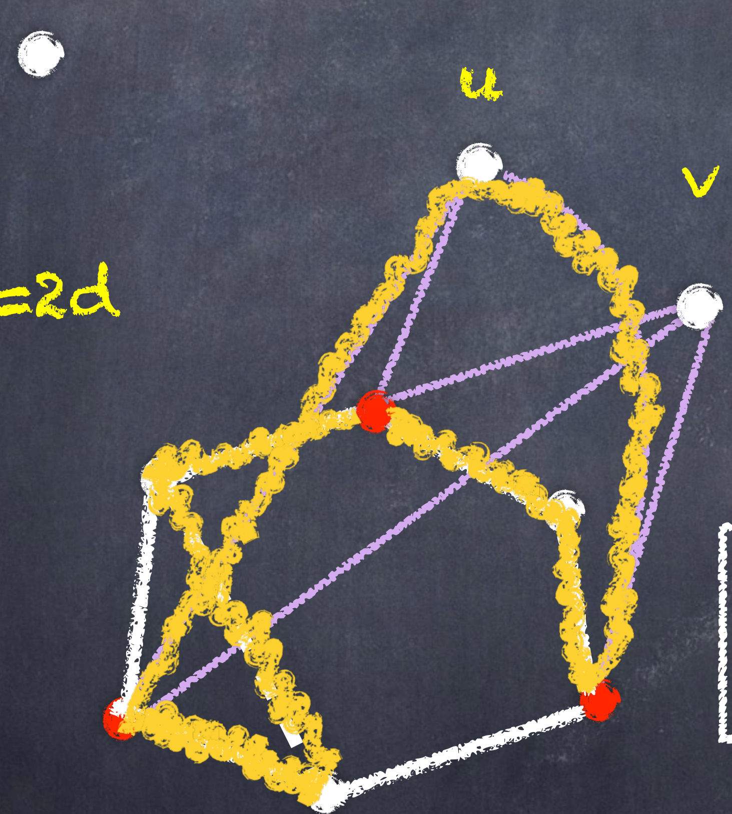
|C|=2d



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.

$H=$



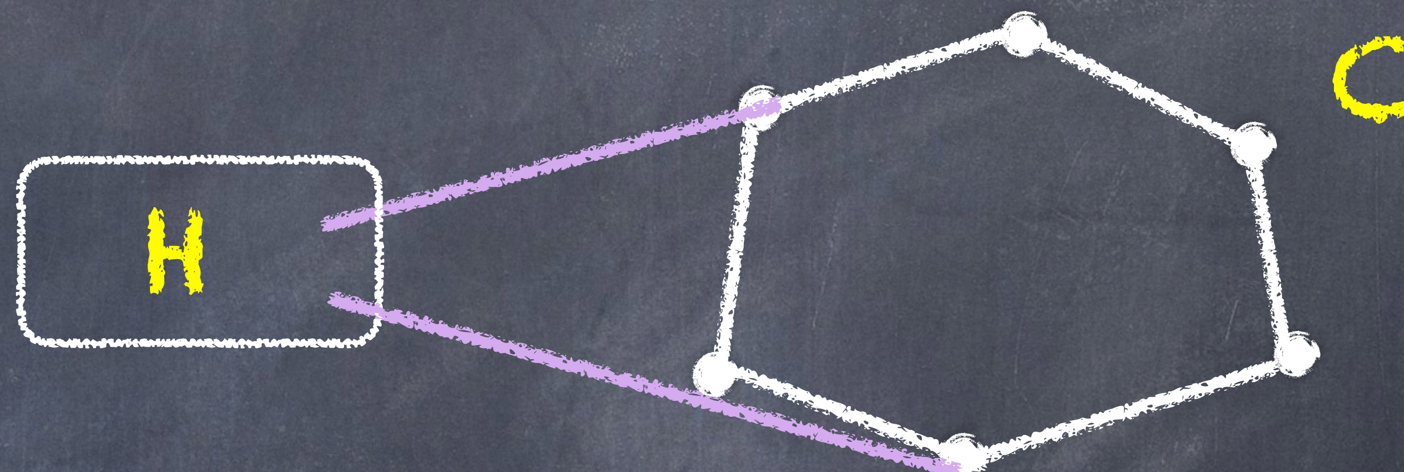
$|C|=2d$

- 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)

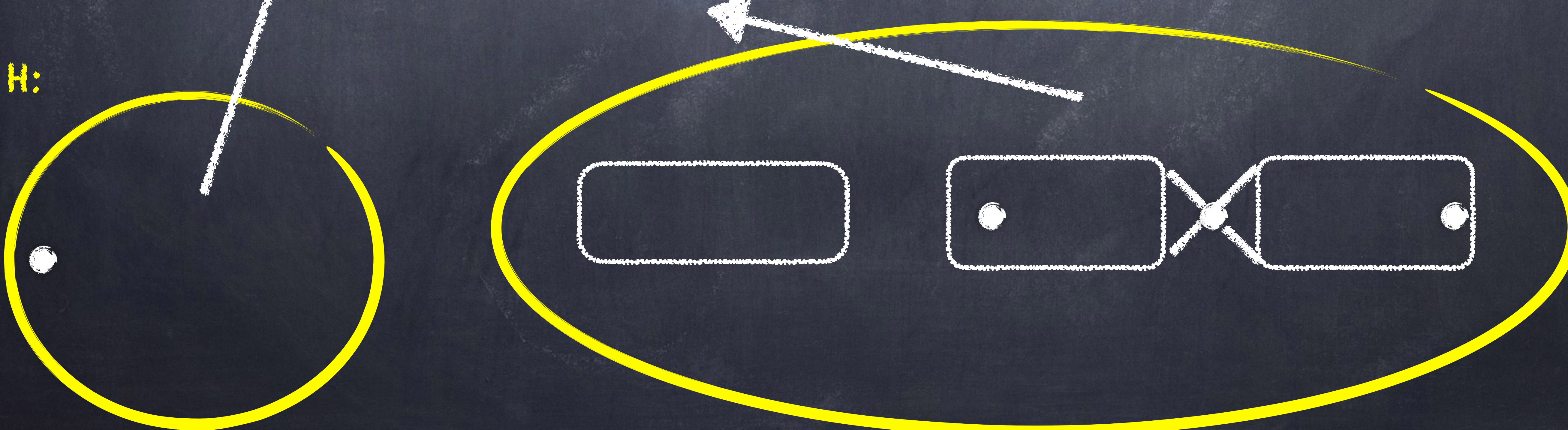
We try to enlarge a cycle C of length 2d+p



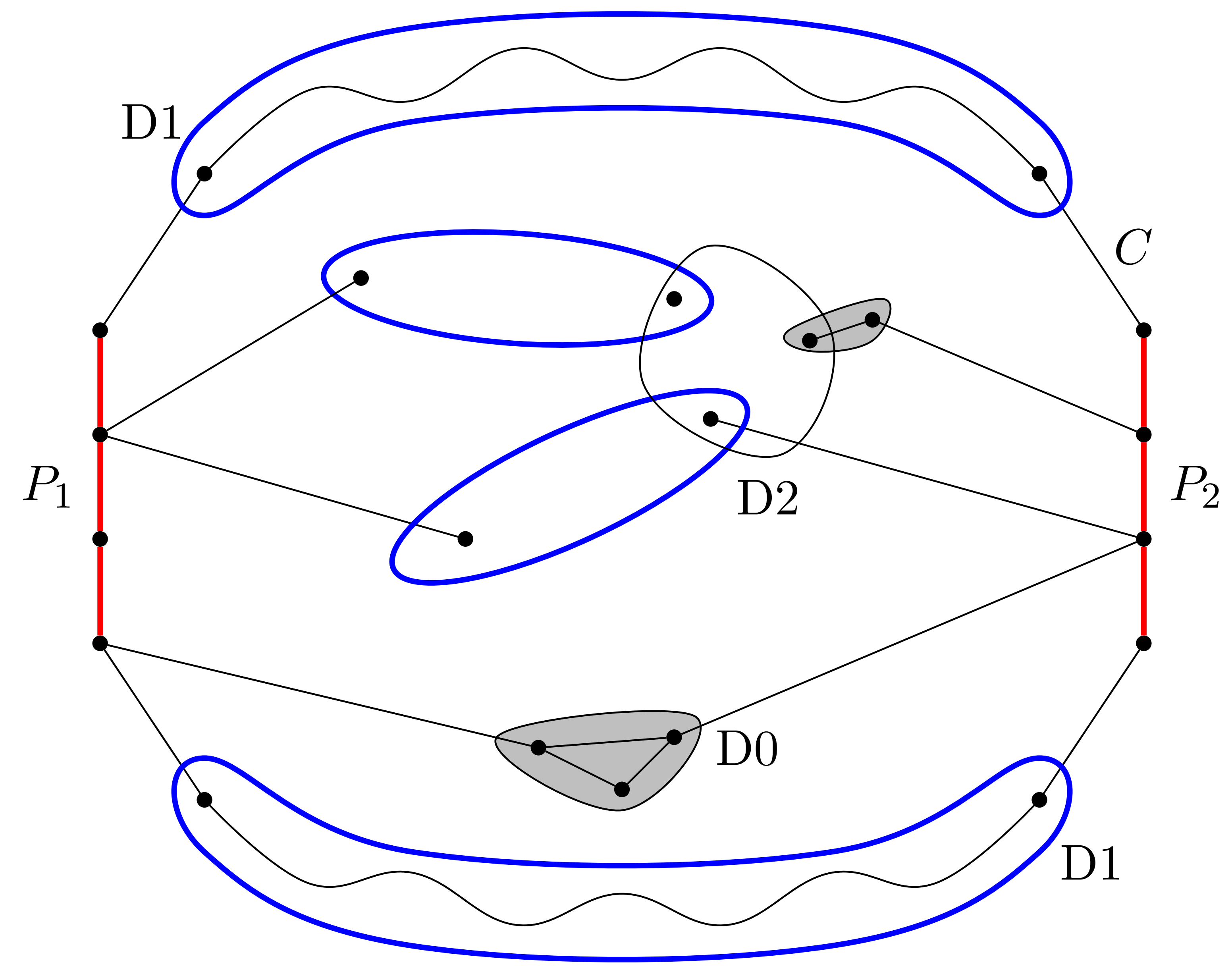
Small Vertex Cover

Dirac Decomposition

H:



Dirac decomposition



Structure of 3-connected graphs

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 $d + 2k$*

$\boxed{\text{Does it solve the problem?}}$

Let G be a 3-connected graph and k be an integer such that $0 < k \le \frac{1}{24}d$. 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 $2^{O(k)}$

Not what we shooting for!

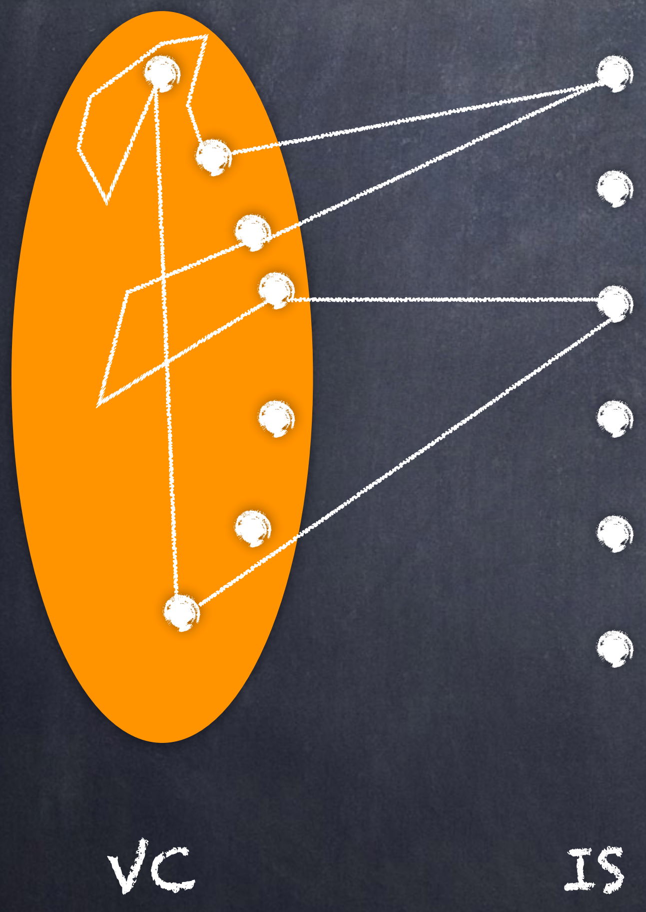
Well-known: If the vertex cover of G is at most p then a longest cycle in G could be found in time $2^{O(p)} \cdot n^{O(1)}$

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

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

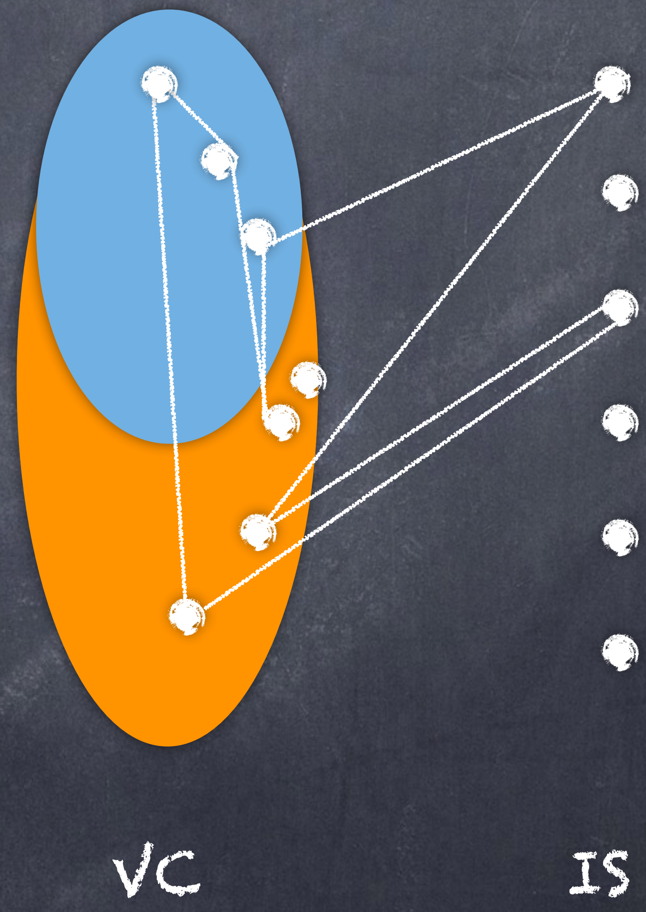
- *returns a longer cycle in G, or*
- *returns a vertex cover of G of size at most $d + 2k$*

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.

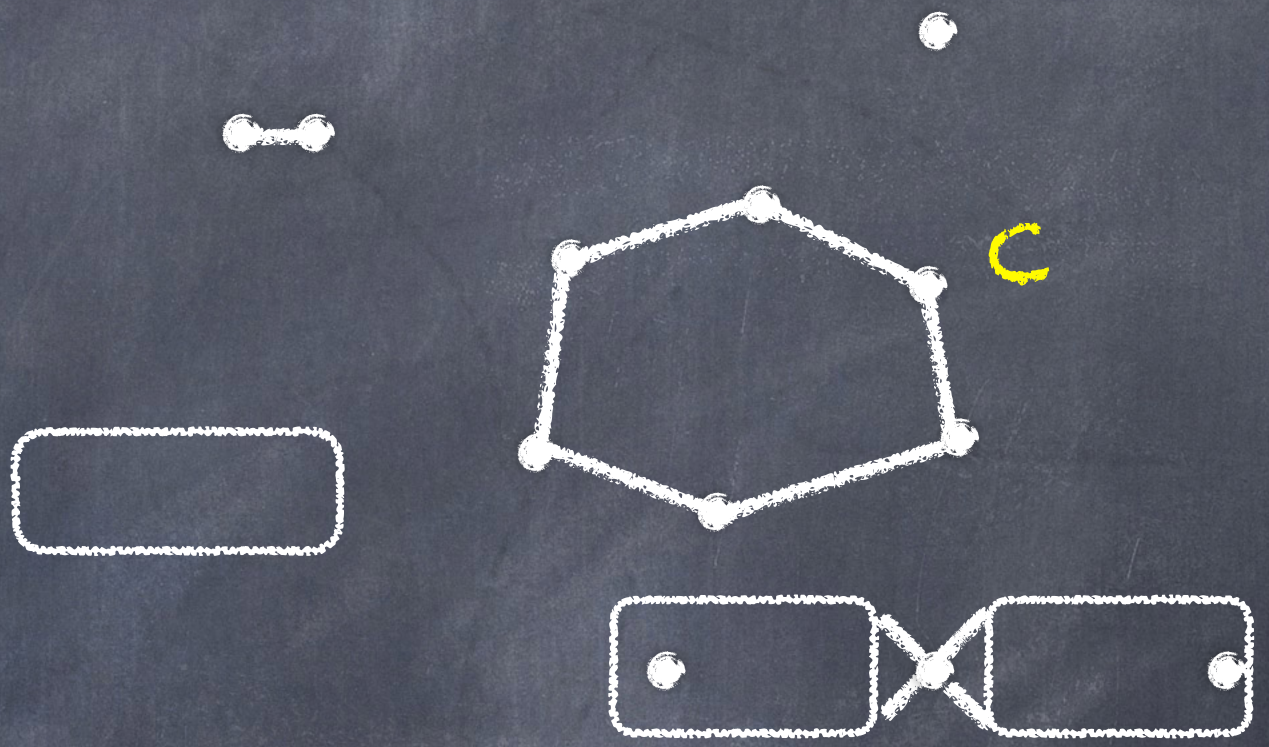
It allows reducing the problem of finding a cycle to the problem of covering vertices in a subgraph by paths of total length O(p).



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

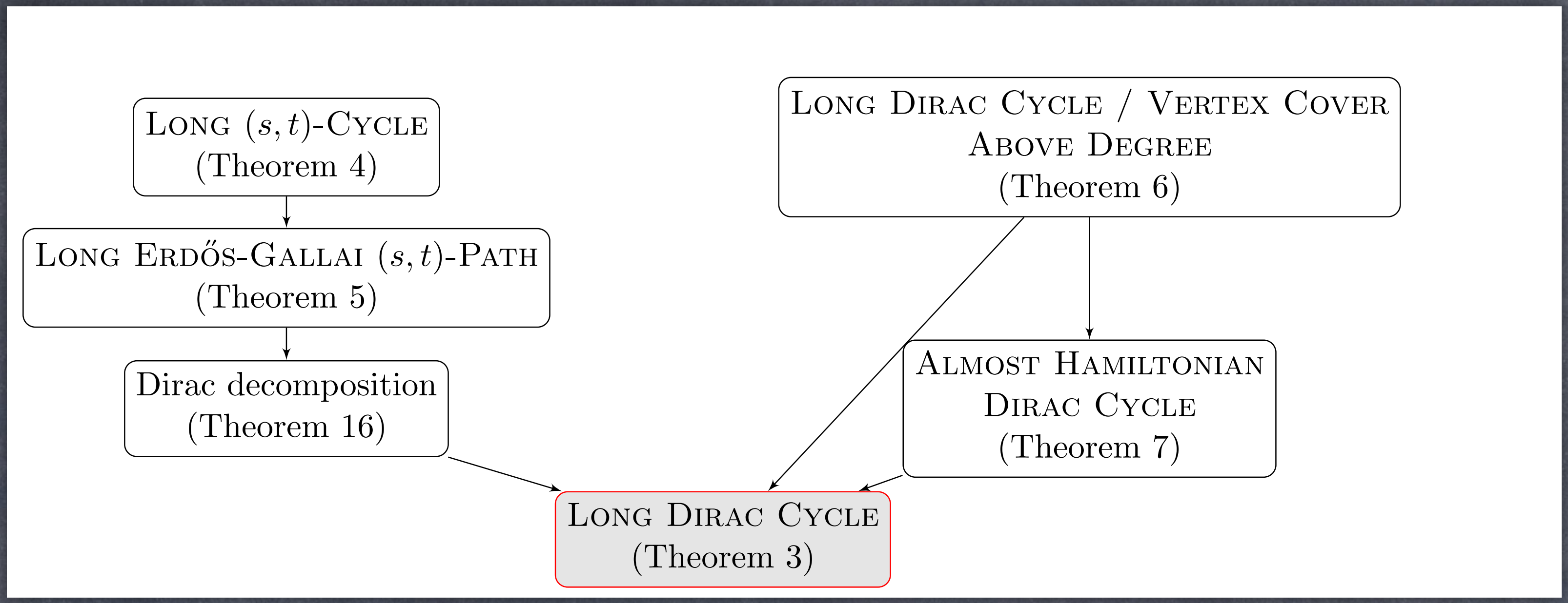
- *returns a longer cycle in G, or*
- *returns a vertex cover of G of size at most $d + 2k$*

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+k in time $2^{O(k)} \cdot n^{O(1)}$

Dirac:

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]

Hamiltonicity below Dirac's condition

Bart M.P. Jansen[1]
Eindhoven University of Technology
b.m.p.jansen@tue.nl
http://orcid.org/0000-0001-8204-1268

László Kozma[2]
Freie Universität Berlin
laszlo.kozma@fu-berlin.de

Jesper Nederlof[3]
Eindhoven University of Technology
j.nederlof@tue.nl

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 $\ell + 1$.

Paul Erdős

Tibor Gallai

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

Theorem (Erdős-Gallai, 1959)

Paul Erdős

Tibor Gallai

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 $D = \frac{2m}{n}$.

$$\frac{2m}{n-1} - 1 \leq D \leq \frac{2m}{n-1}$$

Algorithmic question

Erdős-Gallai bound

Every 2-connected graph of average vertex degree $D > 2$ contains a cycle of length at least D

Above Erdős-Gallai bound

Does a 2-connected graph of average vertex degree $D > 2$ contain a cycle of length at least $D + k$?

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† fedor.fomin@uib.no

Petr A. Golovach† petr.golovach@uib.no

Danil Sagunov‡§ danilka.pro@gmail.com

Kirill Simonov¶ kirillsimonov@gmail.com

Abstract

In 1959, Erdős and Gallai proved that every graph G with average vertex degree $\mathsf{ad}(G) \geq 2$ contains a cycle of length at least $\mathsf{ad}(G)$. We provide an algorithm that for $k \geq 0$ in time $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$ decides whether a 2-connected n-vertex graph G contains a cycle of length at least $\mathsf{ad}(G)+k$. 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

Relevant Work: Cycles above degeneracy

SIAM J. Discrete Math.
Vol. 34, No. 3, pp. 1587–1601

© 2020 Society for Industrial and Applied Mathematics

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 $2^{\mathcal{O}(k)} \cdot |V(G)|^{\mathcal{O}(1)}$. 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 $2^{\mathcal{O}(k)} \cdot n^{\mathcal{O}(1)}$. 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

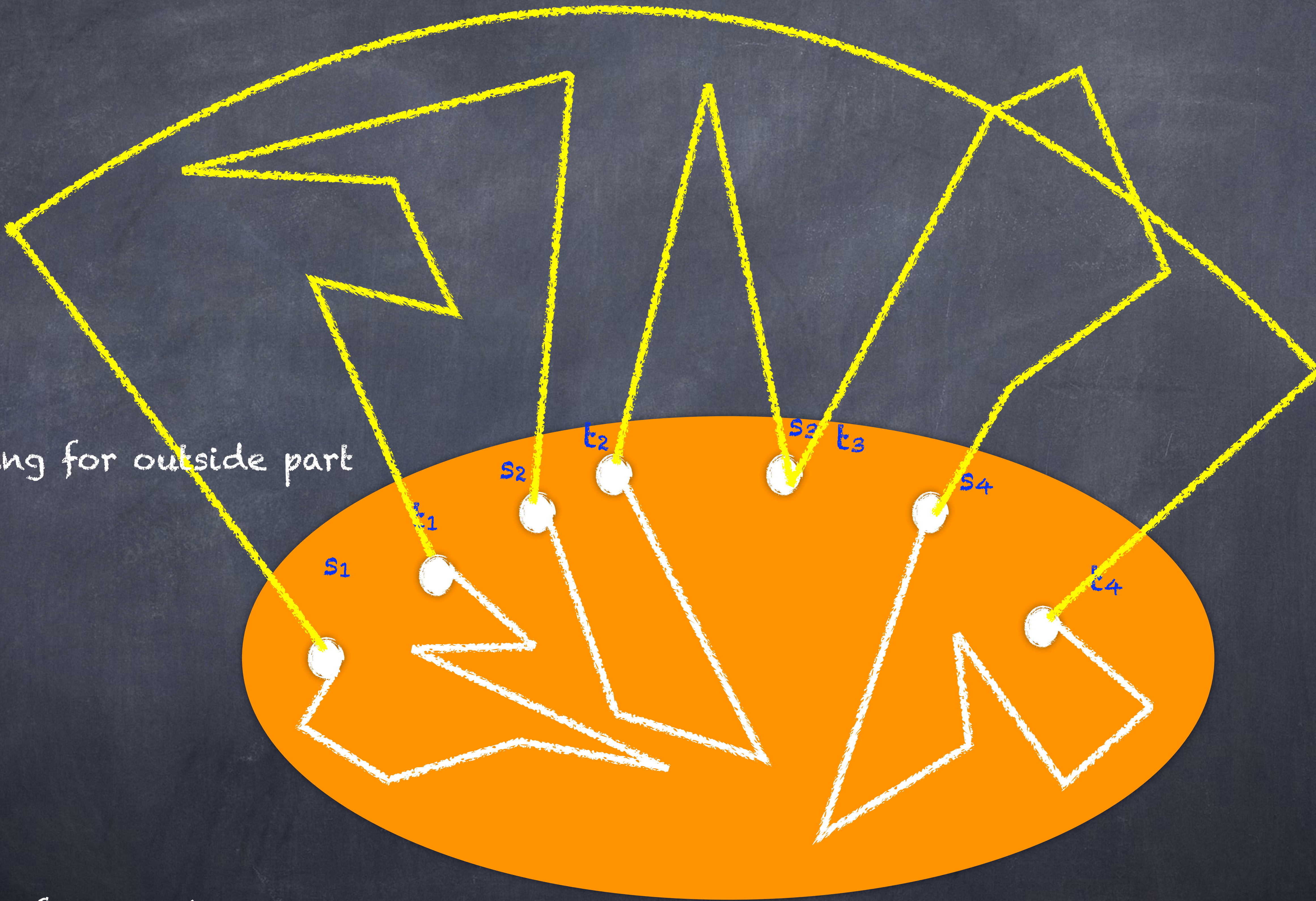
Cycle of length at least maximum average-degree(G)+k

Cycle of length at least average-degree(G)+k

Cycle of length at least degeneracy(G)+k

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

General idea

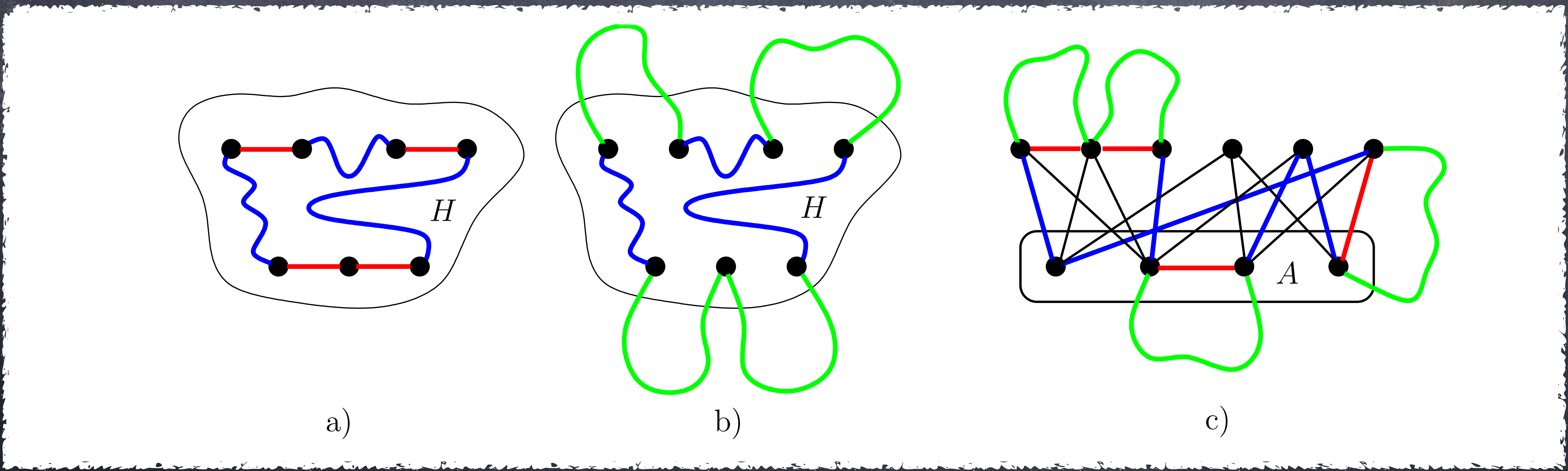


Color-coding for outside part

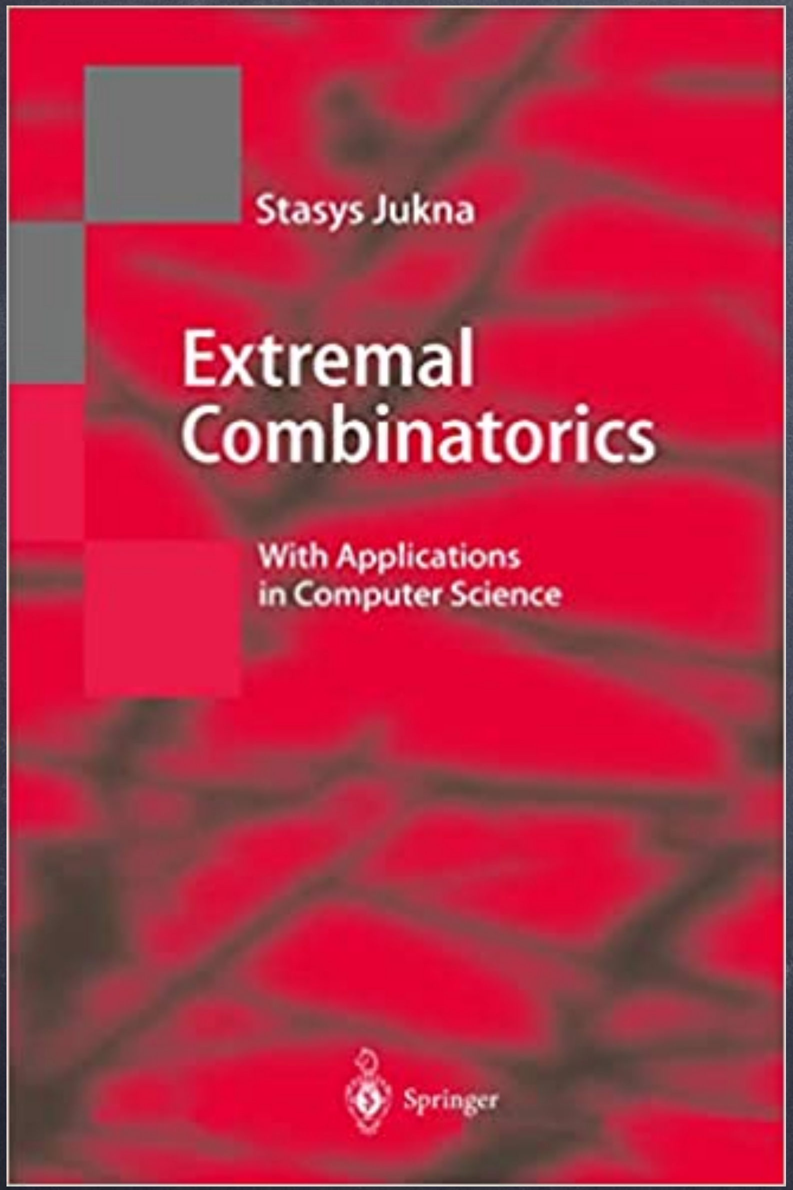
Identify very dense graph H

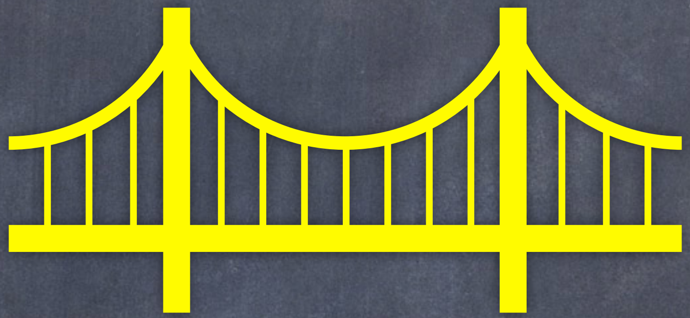
How to identify a dense component?

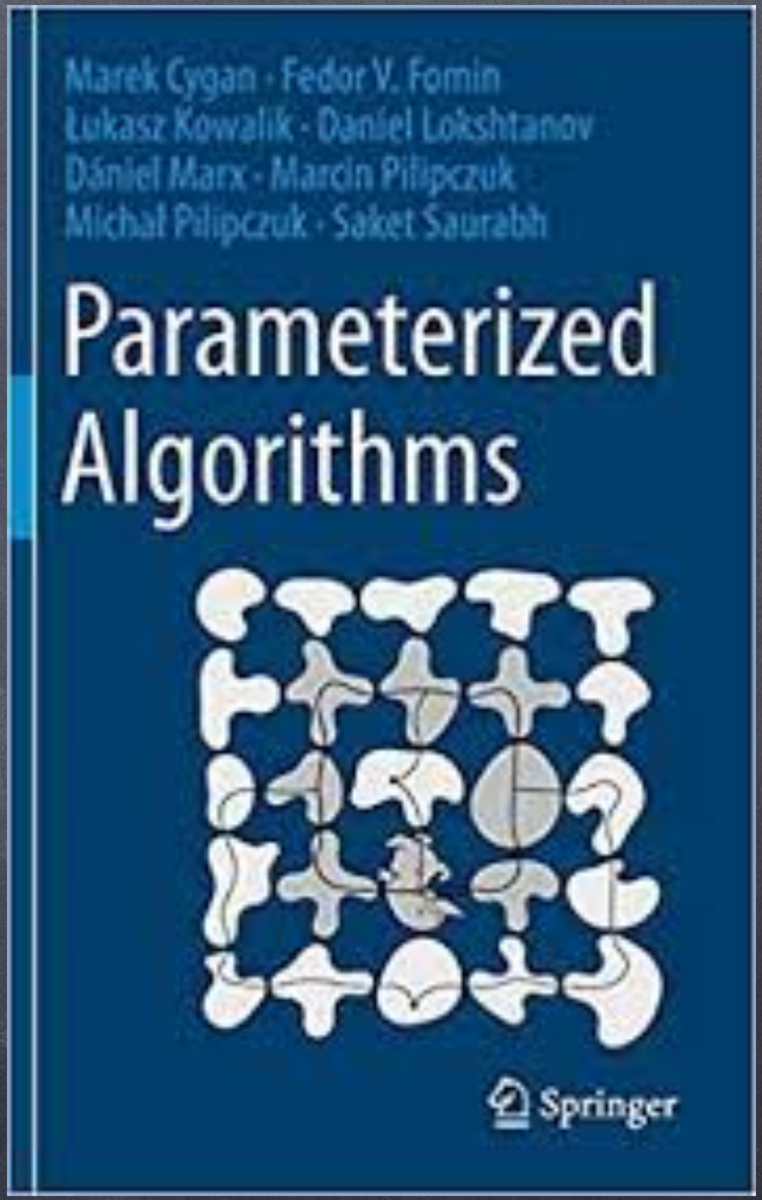
After some preprocessing, an old friend, Dirac's decomposition comes to help



Conclusion


Stasys Jukna
Extremal Combinatorics
With Applications in Computer Science
Springer




Marek Cygan · Fedor V. Fomin
Łukasz Kowalik · Daniel Lokshtanov
Dániel Marx · Marcin Pilipczuk
Michał Pilipczuk · Saket Saurabh
Parameterized Algorithms
Springer

Open questions

Theorem (Thomassen 1981)

Let D be a 2-connected digraph with at least $2d+1$ vertices such that $d^-_D(v) \geq d$ and $d^+_D(v) \geq d$ 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