Constructive algorithm for path-width of matroids

Jisu Jeong(KAIST)

joint work with Eun Jung Kim(CNRS-LAMESADE), Sang-il Oum(KAIST)

2015 Combinatorics Workshop

2015.7.16 NIMS

매트로이드의 패스위드에 대한 건설적인 알고리즘

정지수(카이스트)

김은정(프랑스 국립과학연구센터), 엄상일(카이스트)과의 공동 연구

2015 조합론 학술대회

2015.7.16 국가수리과학연구소

Constructive algorithm for path-width of matroids

algorithm

for

of

algorithm

알고리즘이란 어떠한 문제를 해결하기 위한 여러 동작들의 모임이다.

An **algorithm** is a specific set of instructions for carrying out a procedure or solving a problem, usually with the requirement that the procedure terminate at some point.

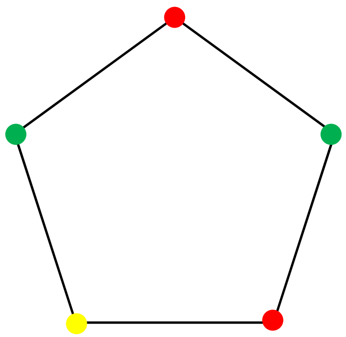
Constructive algorithm

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
planar	embedding on the plane

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
planar	embedding on the plane
chromatic number ≤ k	proper k-coloring

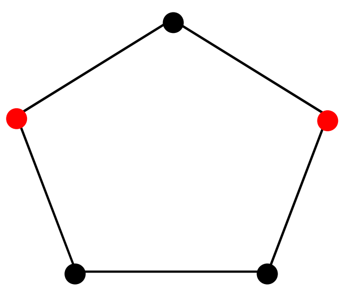


proper 3-coloring

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
planar	embedding on the plane
chromatic number ≤ k	proper k-coloring
dominating number ≤ k	dominating set of size ≤ k

For a graph G=(V,E), a *dominating set* is a subset D of V such that every vertex of G is either in D or a neighbor of D.
The *dominating number* is the *minimum* size of a dominating set.
A set of red vertices is a dominating set of size 2.



Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
planar	embedding on the plane
chromatic number ≤ k	proper k-coloring
dominating number ≤ k	dominating set of size ≤ k
path-width (of matroids) ≤ k	path-decomposition of width ≤ k

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
path-width (of matroids) ≤ k	path-decomposition of width ≤ k

Note that since we only consider `F-representable matroids' with a fixed finite field F, we can say that a **matroid** is a **set of vectors** in F^r.

Definition
A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \le k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example
V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) (0,1,1) (1,1,0) (0,0,1)

Definition
A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \ldots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \ldots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \ldots, v_n\rangle \leq k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example
V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) **1** (0,1,1) (1,1,0) (0,0,1)

$$\dim(\langle(1,0,0)\rangle \cap \langle(0,1,1),(1,1,0),(0,0,1)\rangle) = 1$$

Definition
A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \le k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example

V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) **1** (0,1,1) **3** (1,1,0) (0,0,1)

$$\dim(\langle(1,0,0),(0,1,1)\rangle \cap \langle(1,1,0),(0,0,1)\rangle) = 3$$

Definition

A set V of n vectors has ***path-width at most k*** if there exists a permutation $v_1, v_2, \ldots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \ldots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \ldots, v_n\rangle \leq k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example

V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) 1 (0,1,1) 3 (1,1,0) **1** (0,0,1)

$$\dim(\langle(1,0,0),(0,1,1),(1,1,0)\rangle \cap \langle(0,0,1)\rangle) = 1$$

Definition

A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \ldots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \ldots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \ldots, v_n\rangle \leq k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example
V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) 1 (0,1,1) 3 (1,1,0) **1** (0,0,1)

Thus, V has path-width at most 3.

Definition
A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \ldots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \ldots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \ldots, v_n\rangle \leq k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Example

V={(1,0,0), (0,1,1), (1,1,0), (0,0,1)}

1) (1,0,0) 1 (0,1,1) 3 (1,1,0) 1 (0,0,1)

2) (1,0,0) 1 (1,1,0) 1 (0,1,1) 1 (0,0,1)

Thus, V has path-width at most 1.

Definition

A set V of n vectors has ***path-width at most k*** if
there exists a permutation $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \leq k.$$

Note that such permutation is called a ***path-decomposition of width at most k***.

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
path-width (of matroids) ≤ k	path-decomposition of width ≤ k

Note that since we only consider `F-representable matroids' with a fixed finite field F, we can say that a **matroid** is a **set of vectors** in F^r.

Decision version

Input : a set V of n vectors in F^r and a nonnegative integer k

Output : YES if there exists a permutation $v_1, v_2, \ldots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \ldots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \ldots, v_n\rangle \leq k$$

NO otherwise.

Note that such permutation is called a path-decomposition of width at most k.

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
path-width (of matroids) ≤ k	path-decomposition of width ≤ k

Note that since we only consider `F-representable matroids' with a fixed finite field F, we can say that a **matroid** is a **set of vectors** in F^r.

Constructive version

Input : a set V of n vectors in F^r and a nonnegative integer k

Output : a permutation $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \le k$$

if it exists.

Note that such permutation is called a path-decomposition of width at most k.

Decision algorithm vs Constructive algorithm

Decision algorithm	Constructive algorithm
path-width (of matroids) ≤ k	path-decomposition of width ≤ k

Note that since we only consider \`F-representable matroids' with a fixed finite field F, we can say that a **matroid** is a **set of vectors** in F^r.

Constructive version

Input : a set V of n vectors in F^r and a nonnegative integer k

Output : a permutation $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \le k$$

if it exists.

TIME

Note that such permutation is called a path-decomposition of width at most k.

Fixed parameter tractable algorithm

Dominating Set
Input : an n-vertex graph G and a nonnegative integer k

Output : YES if there exists a dominating set of size at most k
NO otherwise.

Fixed parameter tractable algorithm

Dominating Set
Input : an n-vertex graph G and a nonnegative integer k

Output : YES if there exists a dominating set of size at most k
NO otherwise.

We can solve this problem by computing **all possible sets** in time $O(n^{O(k)})$.

Fixed parameter tractable algorithm

Dominating Set
Input : an n-vertex graph G
Parameter : a nonnegative integer k
Output : YES if there exists a dominating set of size at most k
NO otherwise.

Want to solve a problem in **time** $f(k)n^c$
where c is a fixed constant.

(polynomial in terms of n)

Decision algorithm $\overset{?}{\Leftrightarrow}$ Constructive algorithm

Decision algorithm $\overset{!}{\Leftarrow}$ Constructive algorithm

If we know an embedding of G on the plane,
then G is planar.

If we know a proper k-coloring of G,
then the chromatic number of G is at most k.

If we know a dominating set of size k in G,
then the dominating number of G is at most k.

Decision algorithm $\overset{?}{\Rightarrow}$ Constructive algorithm

Problem
Input : a graph G
Question : Is G planar?

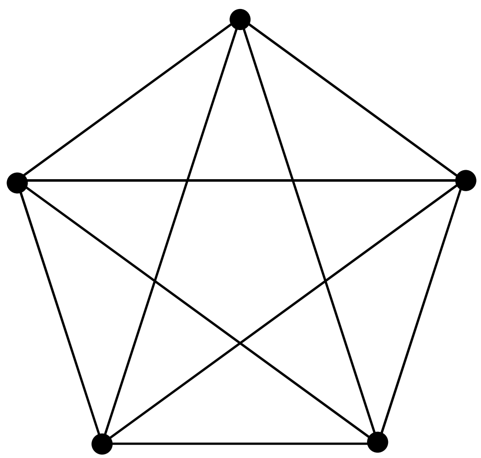
Decision algorithm $\overset{?}{\Rightarrow}$ Constructive algorithm

Problem
Input : a graph G
Question : Is G planar?

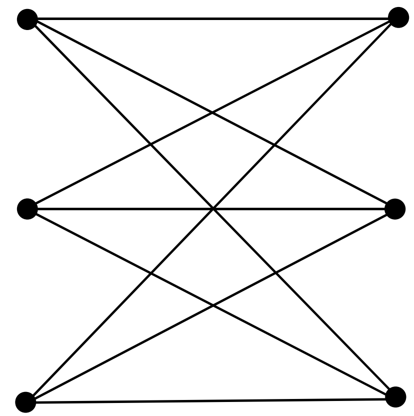
Wagner's theorem(1937)

A graph G is planar if and only if G contains no K_5, $K_{3,3}$ as a minor.

We say G contains H as a minor if
H can be obtained from G
by 1) deleting vertices,
2) deleting edges, or
3) contracting edges.



K_5



$K_{3,3}$

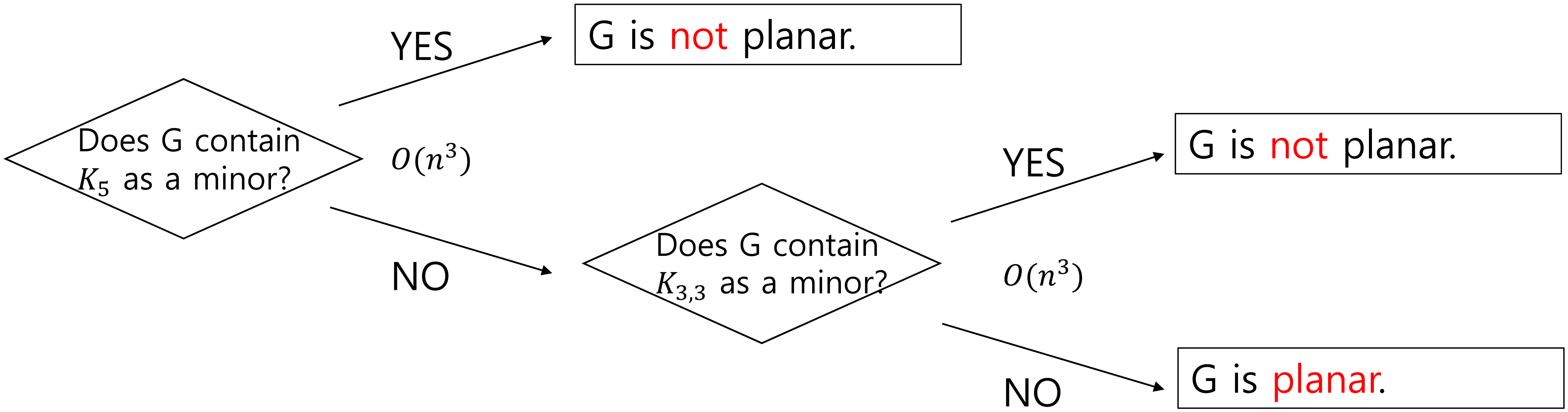
Decision algorithm $\overset{?}{\Rightarrow}$ Constructive algorithm

Problem
Input : a graph G
Question : Is G planar?

Wagner's theorem(1937)

A graph G is planar if and only if G contains no K_5, $K_{3,3}$ as a minor.

(Decision) Algorithm :



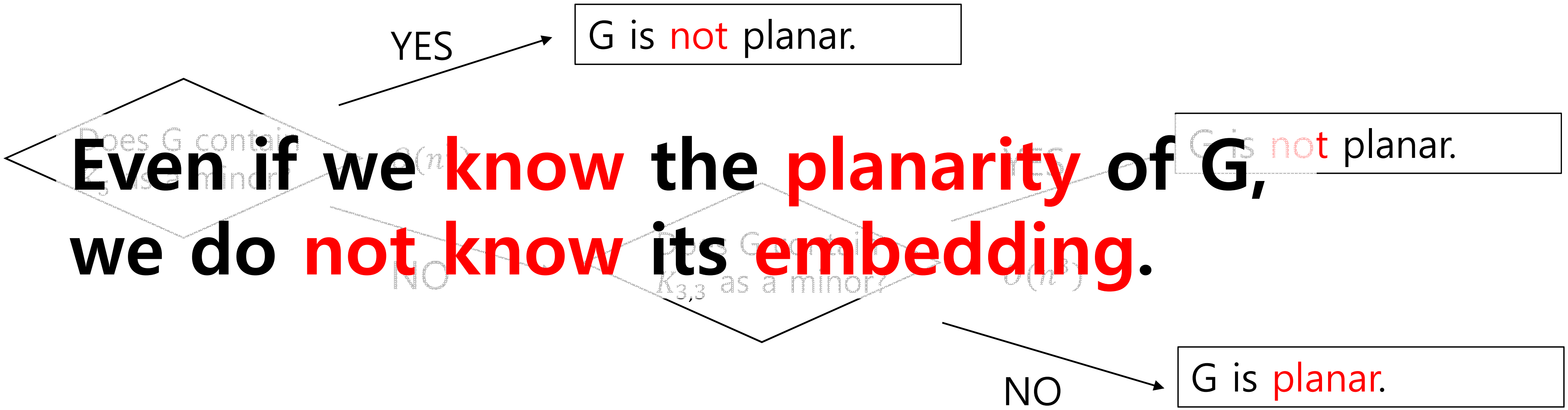
Decision algorithm $\overset{?}{\Rightarrow}$ Constructive algorithm

Problem
Input : a graph G
Question : Is G planar?

Wagner's theorem(1937)

A graph G is planar if and only if G contains no K_5, $K_{3,3}$ as a minor.

(Decision) Algorithm :



Even if we know the planarity of G,
we do not know its embedding.

Decision algorithm for path-width of matroids

Problem

Input : an F-representable matroid M (a set of vectors)

Parameter : a nonnegative integer k

Question : Is the path-width of M at most k?

Decision algorithm for path-width of matroids

Problem
Input : an F-representable matroid M (a set of vectors)
Parameter : a nonnegative integer k
Question : Is the path-width of M at most k?

Theorem(Geelen, Gerards, and Whittle, 2002)

An F-representable matroid M has path-width at most k if and only if M contains no $M_1, M_2, \ldots, M_t$ as a minor.

Theorem(Hlineny, 2005)

One can test whether M contains a fixed matroid N as a minor.

Decision algorithm for path-width of matroids

Problem
Input : an F-representable matroid M (a set of vectors)
Parameter : a nonnegative integer k
Question : Is the path-width of M at most k?

Theorem(Geelen, Gerards, and Whittle, 2002)

An F-representable matroid M has path-width at most k if and only if M contains no $M_1, M_2, \ldots, M_t$ as a minor.

Theorem(Hlineny, 2005)

One can test whether M contains a fixed matroid N as a minor.

A decision algorithm is known.
However, a constructive algorithm is new.

Main results

Constructive algorithm for path-width of a set V of n vectors in F^r

Input : a set V of n vectors in F^r
Parameter : a nonnegative integer k
Output : A path-decomposition $v_1, v_2, \dots, v_n$ of V satisfying that for all i

$$\dim\langle v_1, v_2, \dots, v_i\rangle \cap \langle v_{i+1}, v_{i+2}, \dots, v_n\rangle \leq k$$

if it exists.

Constructive algorithm for path-width of a set W of n subspaces of F^r

Input : a set W of n subspaces of F^r
Parameter : a nonnegative integer k
Output : A path-decomposition $W_1, W_2, \dots, W_n$ of W satisfying that for all i

$$\dim(W_1 + \cdots + W_i) \cap (W_{i+1} + \cdots + W_n) \leq k$$

if it exists.

Main results

We **give the first constructive algorithm** for path-width of a set W of n subspaces of F^r.

Input : a set W of n subspaces of F^r
Parameter : a nonnegative integer k
Output : A permutation $W_1, W_2, \ldots, W_n$ of W satisfying that for all i

$$\dim(W_1 + \cdots + W_i) \cap (W_{i+1} + \cdots + W_n) \leq k$$

if it exists.

Theorem(J., Kim, and Oum, 2015+)

Let F be a fixed finite field. Given an input n subspaces of F^r and a parameter k, in time $O(n^3)$, we can either find a path-decomposition $W_1, W_2, \ldots, W_n$ of the subspaces such that

$$\dim(W_1 + \cdots + W_i) \cap (W_{i+1} + \cdots + W_n) \leq k \text{ for all i, or}$$

confirm that the path-width is larger than k.

Main results

Theorem(J., Kim, and Oum, 2015+)

Let F be a fixed finite field. There is an $O(n^3)$-time algorithm that,
for an input n-element F-represented **matroid** and a parameter k,
decides whether its path-width is at most k
and if so, outputs a path-decomposition of width at most k.

Theorem(J., Kim, and Oum, 2015+)

There is an $O(n^3)$-time algorithm that,
for an input n vectors and a parameter k,
decides whether the trellis-width of a linear **code** generated by these vectors is at most k
and if so, outputs a linear layout of width at most k.

Theorem(J., Kim, and Oum, 2015+)

There is an $O(n^3)$-time algorithm that,
for an input n-vertex **graph** and a parameter k,
decides whether its linear rank-width is at most k
and if so, outputs a linear rank-decomposition of width at most k.

Proof ideas

Dynamic programming
Typical sequence
Subspace analysis

김은정(프랑스 국립과학연구센터)

엄상일(카이스트)

Proof ideas

Dynamic programming
Typical sequence
Subspace analysis

김은정(프랑스 국립과학연구센터)

엄상일(카이스트)

Thank you