Testing Branch-width

Sang-il Oum

School of Mathematics
Georgia Institute of Technology

January 23, 2006

Joint work with

Paul Seymour
Princeton University

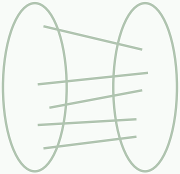
A function $f : 2^V \to \mathbb{Z}$ is a connectivity function if

(i) $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$, (submodular)

(ii) $f(X) = f(V \setminus X)$, (symmetric)

(iii) $f(\emptyset) = 0$.

$v(X)$ =number of vertices meeting both X and $E \setminus X$.



$e(X)$ =number of edges meeting both X and $V \setminus X$.

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.

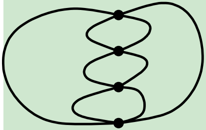
For a graph G, let $A =$ adjacency matrix. $\rho_G(X) = \text{rank}\, A[X, V \setminus X]$.

A function $f : 2^V \to \mathbb{Z}$ is a connectivity function if

(i) $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$, (submodular)

(ii) $f(X) = f(V \setminus X)$, (symmetric)

(iii) $f(\emptyset) = 0$.



$v(X)$ =number of vertices meeting both X and $E \setminus X$.

$e(X)$ =number of edges meeting both X and $V \setminus X$.

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.

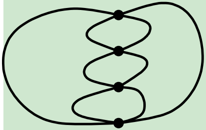
For a graph G, let A = adjacency matrix. $\rho_G(X) = \text{rank } A[X, V \setminus X]$.

A function $f : 2^V \to \mathbb{Z}$ is a connectivity function if

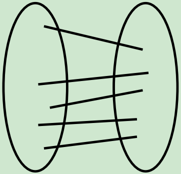
(i) $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$, (submodular)

(ii) $f(X) = f(V \setminus X)$, (symmetric)

(iii) $f(\emptyset) = 0$.



$v(X)$ =number of vertices meeting both X and $E \setminus X$.



$e(X)$ =number of edges meeting both X and $V \setminus X$.

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.

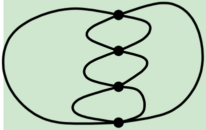
For a graph G, let A = adjacency matrix. $\rho_G(X) = \text{rank}\, A[X, V \setminus X]$.

A function $f : 2^V \to \mathbb{Z}$ is a connectivity function if

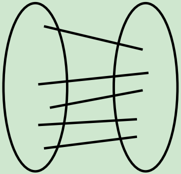
(i) $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$, (submodular)

(ii) $f(X) = f(V \setminus X)$, (symmetric)

(iii) $f(\emptyset) = 0$.



$v(X)$ =number of vertices meeting both X and $E \setminus X$.



$e(X)$ =number of edges meeting both X and $V \setminus X$.

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.

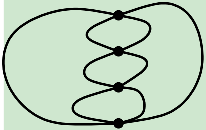
For a graph G, let $A =$ adjacency matrix.
$\rho_G(X) = \operatorname{rank} A[X, V \setminus X]$.

A function $f : 2^V \to \mathbb{Z}$ is a connectivity function if

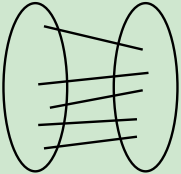
(i) $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$, (submodular)

(ii) $f(X) = f(V \setminus X)$, (symmetric)

(iii) $f(\emptyset) = 0$.



$v(X)$ =number of vertices meeting both X and $E \setminus X$.

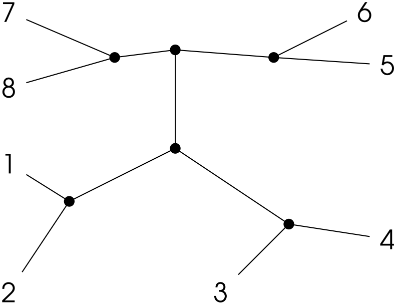


$e(X)$ =number of edges meeting both X and $V \setminus X$.

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.

For a graph G, let A = adjacency matrix. $\rho_G(X) = \operatorname{rank} A[X, V \setminus X]$.

Branch-decomposition of f: a pair (T, L) of a *subcubic tree* T and a *bijection* $L : V \to \{\text{leaves of T}\}$.



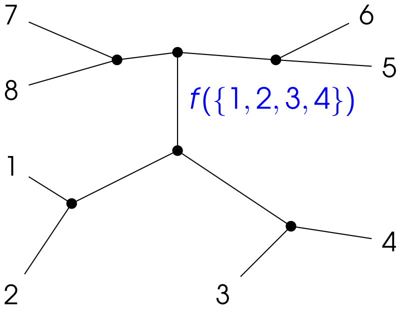
Branch-width

Carving-width

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$. Branch-width of matroids.

For a graph G, let $A =$ adjacency matrix. $\rho_G(X) = \text{rank}\, A[X, V \setminus X]$. Rank-width of graphs

Branch-decomposition of f: a pair (T, L) of
a *subcubic tree* T and a *bijection* $L : V \to \{\text{leaves of T}\}$.



Width of an edge e of T: $f(A_e)$
(A_e, B_e) is a partition of V given by deleting e.

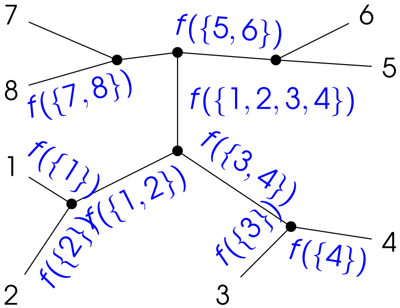
Branch-width

Carving-width

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.
Branch-width of matroids.

For a graph G, let A = adjacency matrix.
$\rho_G(X) = \text{rank}\, A[X, V \setminus X]$.
Rank-width of graphs

Branch-decomposition of f: a pair (T, L) of a *subcubic tree* T and a *bijection* $L : V \to \{\text{leaves of T}\}$.



Width of an edge e of T: $f(A_e)$ (A_e, B_e) is a partition of V given by deleting e.

Width of (T, L): $\max_e \text{width}(e)$

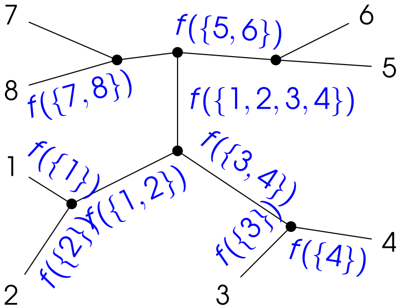
Branch-width

Carving-width

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.
Branch-width of matroids.

For a graph G, let $A =$ adjacency matrix.
$\rho_G(X) = \text{rank } A[X, V \setminus X]$.
Rank-width of graphs

Branch-decomposition of f: a pair (T, L) of a *subcubic tree* T and a *bijection* $L : V \to \{\text{leaves of T}\}$.



Width of an edge e of T: $f(A_e)$
(A_e, B_e) is a partition of V given by deleting e.

Width of (T, L): $\max_e \text{width}(e)$

Branch-width: $min_{(T,L)}$ width(T, L).
(If $|V| \le 1$, then branch-width=0)

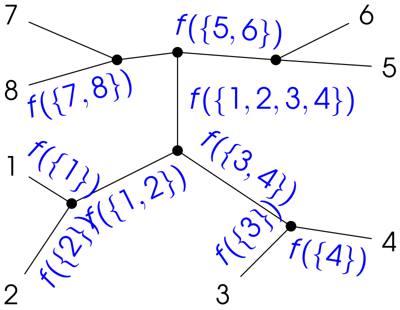
Branch-width

Carving-width

$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.
Branch-width of matroids.

For a graph G, let $A =$ adjacency matrix.
$\rho_G(X) = \text{rank}\, A[X, V \setminus X]$.
Rank-width of graphs

Branch-decomposition of f: a pair (T, L) of a *subcubic tree* T and a *bijection* $L: V \to \{\text{leaves of T}\}$.

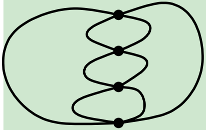


Width of an edge e of T: $f(A_e)$
(A_e, B_e) is a partition of V given by deleting e.

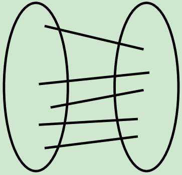
Width of (T, L): $\max_e \text{width}(e)$

Branch-width: $\min_{(T,L)} \text{width}(T, L)$.
(If $|V| \leq 1$, then branch-width=0)

Branch-width



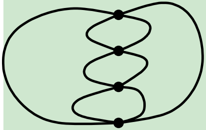
Carving-width



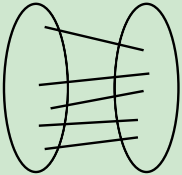
$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.
Branch-width of matroids.

For a graph G, let $A =$ adjacency matrix.
$\rho_G(X) = \text{rank } A[X, V \setminus X]$.
Rank-width of graphs

Branch-width



Carving-width



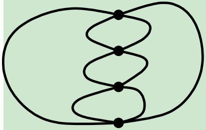
$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$.
Branch-width of matroids.

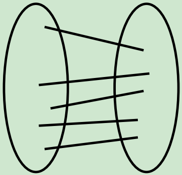
For a graph G, let $A =$ adjacency matrix.
$\rho_G(X) = \operatorname{rank} A[X, V \setminus X]$.
Rank-width of graphs

Testing Branch-width $\leq k$ for fixed k

- Branch-width of graphs: Linear (Bodlaender, Thilikos '97)
- Carving-width of graphs: Linear (Thilikos, Serna, Bodlaendar '00)
- Branch-width of matroids represented over a fixed finite field: $O(|E(\mathcal{M})|^3)$ (Hliněný '05)
- Rank-width of graphs: $O(|V(G)|^3)$ (Oum '05)

Poly-time algorithm to test branch-width$\leq k$ for any connectivity functions? *assuming that f is given by an oracle*.





$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$. Branch-width of matroids.

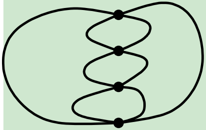
For a graph G, let $A =$ adjacency matrix. $\rho_G(X) = \operatorname{rank} A[X, V \setminus X]$. Rank-width of graphs

Testing Branch-width $\leq k$ for fixed k

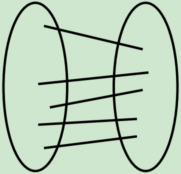
- Branch-width of graphs: Linear (Bodlaender, Thilikos '97)
- Carving-width of graphs: Linear (Thilikos, Serna, Bodlaendar '00)
- Branch-width of matroids represented over a fixed finite field: $O(|E(\mathcal{M})|^3)$ (Hliněný '05)
- Rank-width of graphs: $O(|V(G)|^3)$ (Oum '05)

Poly-time algorithm to test branch-width$\leq k$ for any connectivity functions? *assuming that f is given by an oracle.*

Branch-width



Carving-width



$\mathcal{M}$: matroid, $\lambda(X) = r(X) + r(E(\mathcal{M}) - X) - r(E(\mathcal{M}))$. Branch-width of matroids.

For a graph G, let $A =$ adjacency matrix. $\rho_G(X) = \text{rank}\, A[X, V \setminus X]$. Rank-width of graphs

Testing Branch-width $\leq k$ for fixed k

- Branch-width of graphs: Linear (Bodlaender, Thilikos '97)
- Carving-width of graphs: Linear (Thilikos, Serna, Bodlaendar '00)
- Branch-width of matroids represented over a fixed finite field: $O(|E(\mathcal{M})|^3)$ (Hliněný '05)
- Rank-width of graphs: $O(|V(G)|^3)$ (Oum '05)

Poly-time algorithm to test branch-width$\leq k$ for any connectivity functions? *assuming that f is given by an oracle*.

f-tangle of order $k+1$ (Robertson and Seymour)

A set $\mathcal{T}$ of subsets of V satisfying

(T1) If $f(X) \le k$, then $X \in \mathcal{T}$ or $V \setminus X \in \mathcal{T}$.

(T2) If $A, B, C \in \mathcal{T}$, then $A \cup B \cup C \ne V$.

(T3) $V \setminus \{v\} \notin \mathcal{T}$ for all $v \in V$.

Robertson, Seymour ('91)

Branch-width$\le k$ if and only if no f-tangle of order $k+1$ exists.

Naive algorithm: Choose one from X or $V \setminus X$ if $f(X) \le k$ and see whether (T2) and (T3) are satisfied.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \le k$, then $C \in \mathcal{T}$.

(L3) If $|X| \le 1$ and $f(X) \le k$, then $X \in \mathcal{T}$.

f-tangle of order $k+1$ (Robertson and Seymour)

A set $\mathcal{T}$ of subsets of V satisfying

(T1) If $f(X) \leq k$, then $X \in \mathcal{T}$ or $V \setminus X \in \mathcal{T}$.

(T2) If $A, B, C \in \mathcal{T}$, then $A \cup B \cup C \neq V$.

(T3) $V \setminus \{v\} \notin \mathcal{T}$ for all $v \in V$.

Robertson, Seymour ('91)

Branch-width$\leq k$ if and only if no f-tangle of order $k+1$ exists.

Naive algorithm: Choose one from X or $V \setminus X$ if $f(X) \leq k$ and see whether (T2) and (T3) are satisfied.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

f-tangle of order $k+1$ (Robertson and Seymour)

A set $\mathcal{T}$ of subsets of V satisfying

(T1) If $f(X) \leq k$, then $X \in \mathcal{T}$ or $V \setminus X \in \mathcal{T}$.

(T2) If $A, B, C \in \mathcal{T}$, then $A \cup B \cup C \neq V$.

(T3) $V \setminus \{v\} \notin \mathcal{T}$ for all $v \in V$.

Robertson, Seymour ('91)

Branch-width $\leq k$ if and only if no f-tangle of order $k+1$ exists.

Naive algorithm: Choose one from X or $V \setminus X$ if $f(X) \leq k$ and see whether (T2) and (T3) are satisfied.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

f-tangle of order $k+1$ (Robertson and Seymour)

A set $\mathcal{T}$ of subsets of V satisfying

(T1) If $f(X) \le k$, then $X \in \mathcal{T}$ or $V \setminus X \in \mathcal{T}$.

(T2) If $A, B, C \in \mathcal{T}$, then $A \cup B \cup C \neq V$.

(T3) $V \setminus \{v\} \notin \mathcal{T}$ for all $v \in V$.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \le k$, then $C \in \mathcal{T}$.

(L3) If $|X| \le 1$ and $f(X) \le k$, then $X \in \mathcal{T}$.

THM: An f-tangle of order $k+1$ exists if and only if a loose f-tangle of order $k+1$ exists.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

Naive algorithm to find a loose f-tangle

(1) Begin with $\mathcal{T} = \{X : |X| \leq 1, f(X) \leq k\}$.

(2) Test (L1).
If it fails, then no loose f-tangle of order $k+1$.

(3) Test (L2).
If it fails, then find C and add it to $\mathcal{T}$. Go back to 2.

(4) $\mathcal{T}$ is a loose f-tangle of order $k+1$.

Problem: $|\mathcal{T}|$ can be exponentially large.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

Naive algorithm to find a loose f-tangle

(1) Begin with $\mathcal{T} = \{X : |X| \leq 1, f(X) \leq k\}$.

(2) Test (L1).
If it fails, then no loose f-tangle of order $k+1$.

(3) Test (L2).
If it fails, then find C and add it to $\mathcal{T}$. Go back to 2.

(4) $\mathcal{T}$ is a loose f-tangle of order $k+1$.

Problem: $|\mathcal{T}|$ can be exponentially large.

loose f-tangle of order $k+1$

A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \le k$, then $C \in \mathcal{T}$.

(L3) If $|X| \le 1$ and $f(X) \le k$, then $X \in \mathcal{T}$.

Naive algorithm to find a loose f-tangle

(1) Begin with $\mathcal{T} = \{X : |X| \le 1, f(X) \le k\}$.

(2) Test (L1).
If it fails, then no loose f-tangle of order $k+1$.

(3) Test (L2).
If it fails, then find C and add it to $\mathcal{T}$. Go back to 2.

(4) $\mathcal{T}$ is a loose f-tangle of order $k+1$.

Problem: $|\mathcal{T}|$ can be exponentially large.

loose f-tangle of order $k+1$

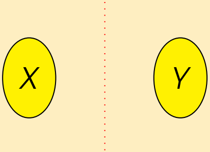
A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

Lemma



Let $f_{\min}(A, B) = \min\{f(X) : A \subseteq X \subseteq V \setminus B\}$.
If $f_{\min}(X, Y) = m$, then $\exists Z$ such that

(i) $f(Z) = m$

(ii) $X \subseteq Z \subseteq V \setminus Y$.

Conversely, if $f(Z) = m$, then $\exists X, Y$ such that

(i) $|X|, |Y| \leq m$ and $X \subseteq Z \subseteq V \setminus Y$,

(ii) $f_{\min}(X, Y) = m$.

loose f-tangle of order $k+1$

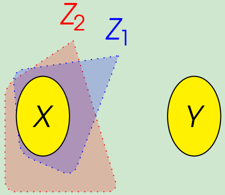
A set $\mathcal{T}$ of subsets of V satisfying

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

Lemma 2



Suppose $f_{\min}(X, Y) = m$, $X \subseteq Z_1, Z_2 \subseteq V \setminus Y$.
If

$$f(Z_1) = f(Z_2) = m,$$

then

$$f(Z_1 \cup Z_2) = m.$$

loose f-tangle of order $k+1$

(L1) $V \notin \mathcal{T}$.

(L2) If $A, B \in \mathcal{T}$, $C \subseteq A \cup B$, and $f(C) \leq k$, then $C \in \mathcal{T}$.

(L3) If $|X| \leq 1$ and $f(X) \leq k$, then $X \in \mathcal{T}$.

loose f-tangle kit of order $k+1$

A pair (P, μ) where
$P = \{(A, B) : A \cap B = \emptyset, \max(|A|, |B|) \leq f_{\min}(A, B) \leq k.\}$
and $\mu : P \to 2^V$ is a function satisfying the following.

(K1) $\mu(\emptyset, \emptyset) \neq V$ if $(\emptyset, \emptyset) \in P$.

(K2) If $(A, B), (C, D), (E, F) \in P$, $E \subseteq X \subseteq \mu(A, B) \cup \mu(C, D) - F$, and $f_{\min}(E, F) = f(X)$, then $X \subseteq \mu(E, F)$.

(K3) If $|X| \leq 1$, $f(X) \leq 1$,
then there exists $(A, B) \in P$ such that $A \subseteq X \subseteq V \setminus B$,
$f(X) = f_{\min}(A, B)$, and $X \subseteq \mu(A, B)$.

(K1) $\mu(\emptyset,\emptyset) \neq V$ if $(\emptyset,\emptyset) \in P$.

(K2) If $(A,B),(C,D),(E,F) \in P$, $E \subseteq X \subseteq \mu(A,B) \cup \mu(C,D) - F$, and $f_{\min}(E,F) = f(X)$, then $X \subseteq \mu(E,F)$.

(K3) If $|X| \leq 1$, $f(X) \leq 1$,
then there exists $(A,B) \in P$ such that $A \subseteq X \subseteq V \setminus B$,
$f(X) = f_{\min}(A,B)$, and $X \subseteq \mu(A,B)$.

Poly-time algorithm to find a loose f-tangle

(A1) Let $P = \{(A,B) : A \cap B = \emptyset, \max(|A|,|B|) \leq f_{\min}(A,B) \leq k\}$.

(A2) For each $v \in V$, if $0 < f(\{v\}) \leq k$, then find $B \subseteq V \setminus \{v\}$ such that $|B| \leq f_{\min}(\{v\},B) \leq k$. Let $\mu(\{v\},B) = \{v\}$.
Let $\mu(\emptyset,\emptyset) = \{v \in V : f(\{v\}) = 0\}$ if $(\emptyset,\emptyset) \in P$.
For all other $(A,B) \in P$, let $\mu(A,B) = \emptyset$.

(A3) Test (K1). If it fails, then no loose f-tangle kit of order $k+1$.

(A4) Test (K2).
If it fails, then find X and enlarge $\mu(E,F)$. Go back to (A3).

(A5) (P,μ) is a loose f-tangle kit of order $k+1$.

(K1) $\mu(\emptyset, \emptyset) \neq V$ if $(\emptyset, \emptyset) \in P$.

(K2) If $(A, B), (C, D), (E, F) \in P$, $E \subseteq X \subseteq \mu(A, B) \cup \mu(C, D) - F$, and $f_{\min}(E, F) = f(X)$, then $X \subseteq \mu(E, F)$.

(K3) If $|X| \leq 1$, $f(X) \leq 1$,
then there exists $(A, B) \in P$ such that $A \subseteq X \subseteq V \setminus B$,
$f(X) = f_{\min}(A, B)$, and $X \subseteq \mu(A, B)$.

Poly-time algorithm to find a loose f-tangle

(A1) Let $P = \{(A, B) : A \cap B = \emptyset, \max(|A|, |B|) \leq f_{\min}(A, B) \leq k\}$.

(A2) For each $v \in V$, if $0 < f(\{v\}) \leq k$, then find $B \subseteq V \setminus \{v\}$ such that $|B| \leq f_{\min}(\{v\}, B) \leq k$. Let $\mu(\{v\}, B) = \{v\}$.
Let $\mu(\emptyset, \emptyset) = \{v \in V : f(\{v\}) = 0\}$ if $(\emptyset, \emptyset) \in P$.
For all other $(A, B) \in P$, let $\mu(A, B) = \emptyset$.

(A3) Test (K1). If it fails, then no loose f-tangle kit of order $k + 1$.

(A4) Test (K2).
If it fails, then find X and enlarge $\mu(E, F)$. Go back to (A3).

(A5) (P, μ) is a loose f-tangle kit of order $k + 1$.

Poly-time algorithm to find a loose f-tangle

(A1) Let $P = \{(A, B) : A \cap B = \emptyset, \max(|A|, |B|) \le f_{\min}(A, B) \le k\}$.

(A2) For each $v \in V$, if $0 < f(\{v\}) \le k$, then find $B \subseteq V \setminus \{v\}$ such that $|B| \le f_{\min}(\{v\}, B) \le k$. Let $\mu(\{v\}, B) = \{v\}$.
Let $\mu(\emptyset, \emptyset) = \{v \in V : f(\{v\}) = 0\}$ if $(\emptyset, \emptyset) \in P$.
For all other $(A, B) \in P$, let $\mu(A, B) = \emptyset$.

(A3) Test (K1). If it fails, then no loose f-tangle kit of order $k + 1$.

(A4) Test (K2).
If it fails, then find X and enlarge $\mu(E, F)$. Go back to (A3).

(A5) (P, μ) is a loose f-tangle kit of order $k + 1$.

Time Complexity: $O(n^{2k} n n^{6k+1} n n^5 \log n)$

Consequence to Matroids

Poly-time algorithm to test matroid branch-width $\le k$ for fixed k, when the input matroid is given by an independence oracle.

Constructing Branch-decomposition of width $\leq k$

Is it possible to construct the branch-decomposition of width $\leq k$ if there exists one in polynomial time (in $|V|$)? Yes.

Jim Geelen (2005, private communication):
Recursively find a pair $a, b \in V$ such that merging them does not increase branch-width.
We only need $O(n^3)$ calls to testing branch-width at most k.

Further topics

Is it fixed parameter tractable?
In other words, is it possible to have a running time $O(f(k)|V|^c)$ for all k?

Thank you!

Constructing Branch-decomposition of width $\leq k$

Is it possible to construct the branch-decomposition of width $\leq k$ if there exists one in polynomial time (in $|V|$)? Yes.

Jim Geelen (2005, private communication):
Recursively find a pair $a, b \in V$ such that
merging them does not increase branch-width.
We only need $O(n^3)$ calls to testing branch-width at most k.

Further topics

Is it fixed parameter tractable?
In other words, is it possible to have a running time $O(f(k)|V|^c)$ for all k?

Thank you!

Constructing Branch-decomposition of width $\leq k$

Is it possible to construct the branch-decomposition of width $\leq k$ if there exists one in polynomial time (in $|V|$)? Yes.

Jim Geelen (2005, private communication):
Recursively find a pair $a, b \in V$ such that
merging them does not increase branch-width.
We only need $O(n^3)$ calls to testing branch-width at most k.

Further topics

Is it fixed parameter tractable?
In other words, is it possible to have a running time $O(f(k)|V|^c)$ for all k?

Thank you!

Constructing Branch-decomposition of width $\leq k$

Is it possible to construct the branch-decomposition of width $\leq k$ if there exists one in polynomial time (in $|V|$)? Yes.

Jim Geelen (2005, private communication):
Recursively find a pair $a, b \in V$ such that
merging them does not increase branch-width.
We only need $O(n^3)$ calls to testing branch-width at most k.

Further topics

Is it fixed parameter tractable?
In other words, is it possible to have a running time $O(f(k)|V|^c)$ for all k?

Thank you!