Parameterized Algorithms using Matroids

Lecture III: Advance Applications of Representative Sets

Saket Saurabh

The Institute of Mathematical Sciences, India
and University of Bergen, Norway,

ADFOCS 2013, MPI, August 5–9, 2013

This lecture is based on the following paper:

Stefan Kratsch and Magnus Wahlström, Representative Sets and Irrelevant Vertices: New Tools for Kernelization, FOCS 2012, 450-459.

Given: A a matroid $(M, \mathcal{I})$, and a family of p-sized subsets from $\mathcal{I}$:

$$S_1, S_2, \ldots, S_t$$

Given: A a matroid $(M, \mathcal{I})$, and a family of p-sized subsets from $\mathcal{I}$:

$$S_1, S_2, \ldots, S_t$$

Want: A subfamily $\widehat{\mathcal{F}}$ of $\mathcal{F}$ such that:

Given: A a matroid $(M, \mathcal{I})$, and a family of p-sized subsets from $\mathcal{I}$:

$$S_1, S_2, \ldots, S_t$$

Want: A subfamily $\widehat{\mathcal{F}}$ of $\mathcal{F}$ such that:

For any $X \subseteq [n]$ of size at most q,

if there is a set S in $\mathcal{F}$ such that $X \cap S = \emptyset$ and $X \cup S \in \mathcal{I}$,
then there is a set $\widehat{S}$ in $\widehat{\mathcal{F}}$ such that $X \cap \widehat{S} = \emptyset$ and $X \cup \widehat{S} \in \mathcal{I}$.

Given: A a matroid $(M, \mathcal{I})$, and a family of p-sized subsets from $\mathcal{I}$:

$$S_1, S_2, \ldots, S_t$$

There is a subfamily $\widehat{\mathcal{F}}$ of $\mathcal{F}$ of size at most $\binom{p+q}{p}$ such that:

For any $X \subseteq [n]$ of size at most q,

if there is a set S in $\mathcal{F}$ such that $X \cap S = \emptyset$ and $X \cup S \in \mathcal{I}$,
then there is a set $\widehat{S}$ in $\widehat{\mathcal{F}}$ such that $X \cap \widehat{S} = \emptyset$ and $X \cup \widehat{S} \in \mathcal{I}$.

Lovász, 1977

Given: A a matroid $(M, \mathcal{I})$, and a family of p-sized subsets from $\mathcal{I}$:

$$S_1, S_2, \ldots, S_t$$

There is an efficiently computable subfamily $\widehat{\mathcal{F}}$ of $\mathcal{F}$ of size at most $\binom{p+q}{p}$ such that:

For any $X \subseteq [n]$ of size at most q,

if there is a set S in $\mathcal{F}$ such that $X \cap S = \emptyset$ and $X \cup S \in \mathcal{I}$,
then there is a set $\widehat{S}$ in $\widehat{\mathcal{F}}$ such that $X \cap \widehat{S} = \emptyset$ and $X \cup \widehat{S} \in \mathcal{I}$.

Márx (2009) and Fomin, Lokshtanov, Saurabh (2013)

Summary.

We have at hand a p-uniform collection of independent sets, $\mathcal{F}$ and a number q. Let X be any set of size at most q. For any set $S \in \mathcal{F}$, if:

a X is disjoint from S, and

b X and S together form an independent set,

then a q-representative family $\widehat{\mathcal{F}}$ contains a set $\widehat{S}$ that is:

a disjoint from X, and

b forms an independent set together with X.

Such a subfamily is called a q-representative family for the given family.

Digraph Pair Problem

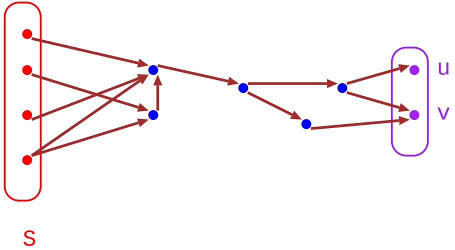
Digraph Pair Cut Problem

Input: A directed graph $D = (V, A)$, a source vertex $s \in V$ and a set $\mathcal{P}$ of pairs of vertices.

Parameter: k

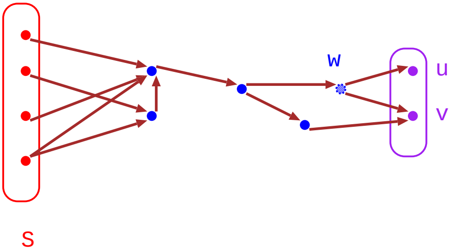
Question: Does there exist a set $X \subseteq V \setminus \{s\}$ of size at most k such that every pair in $\mathcal{P}$ is not reachable from s in $D \setminus X$?

Reachable pair : A pair of vertices, say (u, v) such that both are reachable by paths (need not be disjoint) from S.

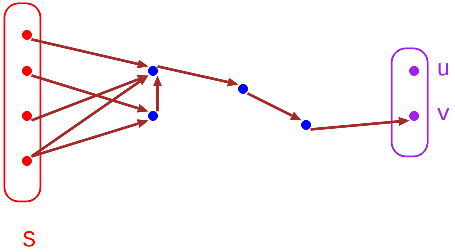


Reachable pair : A pair of vertices, say (u, v) such that both are reachable by paths (need not be disjoint) from S.

Want to delete vertex w.



Reachable pair : A pair of vertices, say (u, v) such that both are reachable by paths (need not be disjoint) from S.
Deleting w makes the pair (u, v) non-reachable from S.



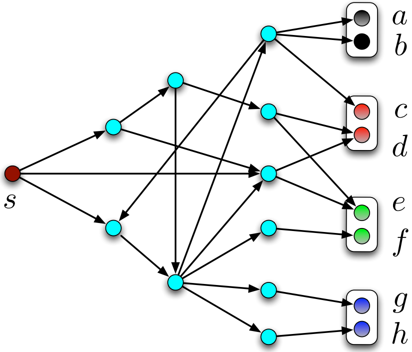
Digraph Pair Cut Problem

Input: A directed graph $D = (V, A)$, a source vertex $s \in V$ and a set $\mathcal{P}$ of pairs of vertices.

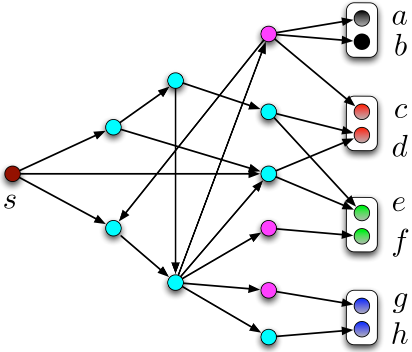
Parameter: k

Question: Does there exist a set $X \subseteq V \setminus \{s\}$ of size at most k such that every pair in $\mathcal{P}$ is not reachable from s in $D \setminus X$?

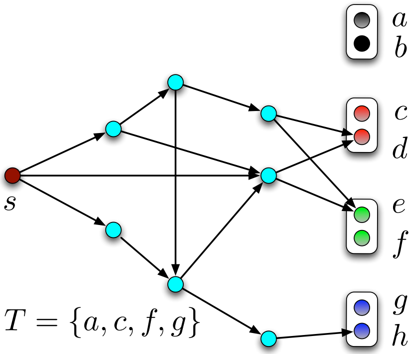
Important Observation



•

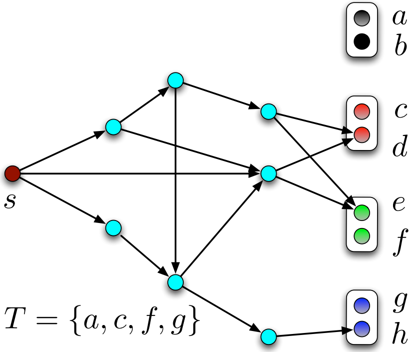


- Let X be a solution to the problem.

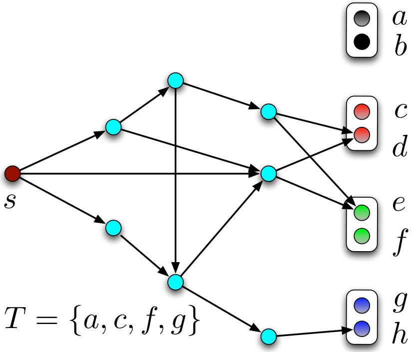


- Let X be a solution to the problem.
- Clearly no pair $(u, v) \in \mathcal{P}$ is reachable from s in $D \setminus X$.

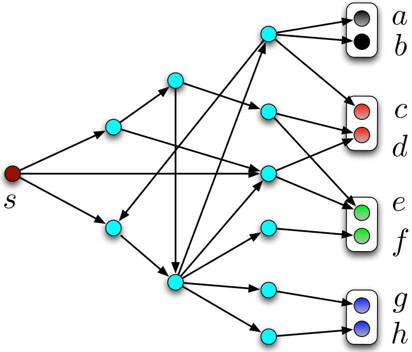
Important Observation



- Let X be a solution to the problem.
- Let T be a set consisting of vertices, say $u \in \{u, v\}$, from each pair (u, v), such that there is no path from s to u in $D \setminus X$.



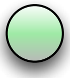
- Let X be a solution to the problem.
- Clearly, X is a s-T separator in D. In fact, X could be any minimum cut between s and T in D.



- Let X be a solution to the problem and $T = \{a, c, f, g\}$.
- Clearly, X is a s-T separator in D. In fact, X could be any minimum cut between s and T in D.

A first attempt at an FPT algorithm: Branching Algorithm

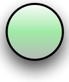
$$(D, s, T = \emptyset)$$



1. Initialise a set $T = \emptyset$

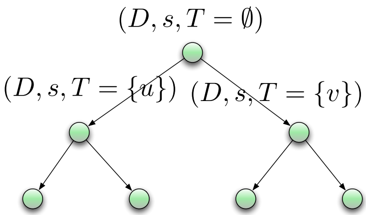
A first attempt at an FPT algorithm: Branching Algorithm

$$(D, s, T = \emptyset)$$



1. Initialise a set $T = \emptyset$
2. If the size of the (s, T)-minimum cut is at least $k + 1$, then we stop and say NO.
3. If there is an (s, T)-minimum cut C of size at most k such that no pairs of $\mathcal{P}$ are reachable from s, return YES.

A first attempt at an FPT algorithm: Branching Algorithm



1. Initialise a set $T = \emptyset$
2. If the size of the (s, T)-minimum cut is at least $k + 1$, then we stop and say NO.
3. If there is an (s, T)-minimum cut C of size at most k such that no pairs of $\mathcal{P}$ are reachable from s, return YES.
4. Else, there is a pair $(u, v) \in \mathcal{P}$ which is reachable from s in $D \setminus C$
5. Pick any such reachable pair and make a two-way branch for adding u or v to T. Return to step 2

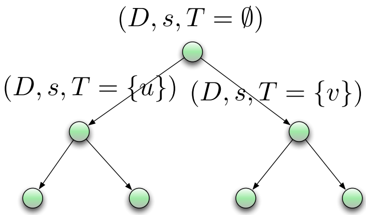
Drawbacks

We do not know how many iterations are required before all pairs of $\mathcal{P}$ become nonreachable from s. The algorithm could take $2^{|\mathcal{P}|}$ time.

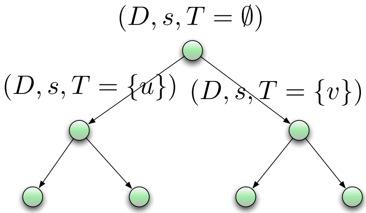
A new strategy

- Show that some parameter, which has to be positive in any graph, drops at every iteration of the branching algorithm.

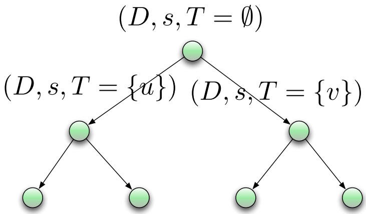
- Show that some parameter, which has to be positive in any graph, drops at every iteration of the branching algorithm.
- Parameter be $\mu = k - \lambda$. Here λ is the size of a (s, T)-minimum cut for the local T of an iteration.



- Show that some parameter, which has to be positive in any graph, drops at every iteration of the branching algorithm.
- Suppose, at the beginning of iteration i we find a (s, T)-minimum cut C, we find a reachable pair (u, v) in $D \setminus C$.

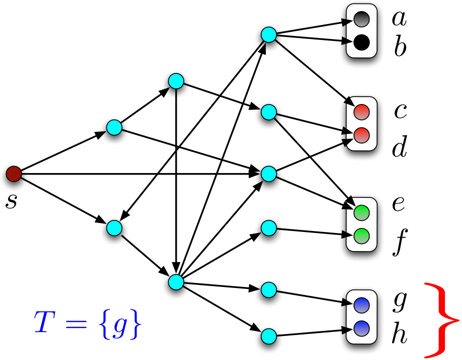


- Show that some parameter, which has to be positive in any graph, drops at every iteration of the branching algorithm.
- Suppose, at the beginning of iteration i we find a (s, T)-minimum cut C, we find a reachable pair (u, v) in $D \setminus C$.
- Look at any one of the branches (say the one which picks u for T). The size of the minimum cut in the $(i+1)^{st}$ iteration could be of the same size as C.



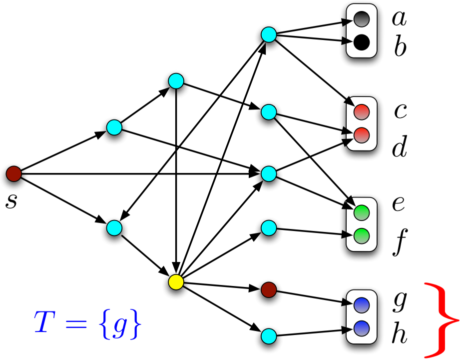
- Show that some parameter, which has to be positive in any graph, drops at every iteration of the branching algorithm.
- Suppose, at the beginning of iteration i we find a (s, T)-minimum cut C, we find a reachable pair (u, v) in $D \setminus C$.
- Is there a minimum cut which will strictly increase in size in every step of the iteration, on both the branches?

Yes there is!



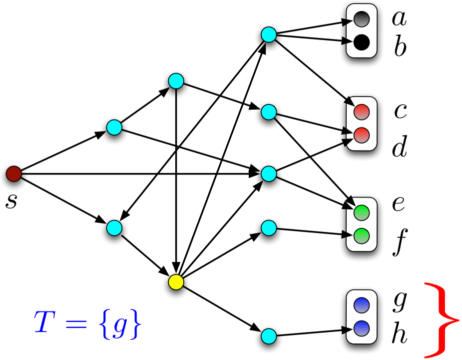
- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.

Yes there is!



- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.
- It seems natural that find a minimum cut that is "closest" to S – as this may help in disconnecting other pairs.

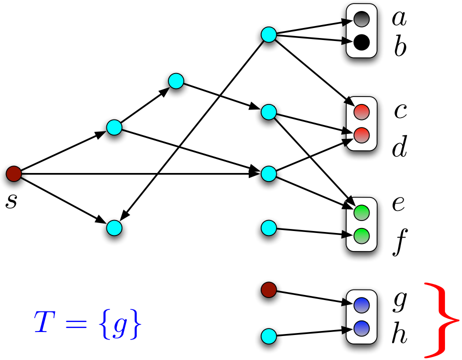
Yes there is!



- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.

- It seems natural that find a minimum cut that is "closest" to S – as this may help in disconnecting other pairs.

Yes there is!



- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.

- It seems natural that find a minimum cut that is "closest" to S – as this may help in disconnecting other pairs.

Yes there is!

- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.

- Closest set : A set $X \subseteq V$ is closest to S if X is the unique (S, X)-mincut . That is, the only cut of size at most $|X|$, for paths from S to X, is X itself. X is called closest set.

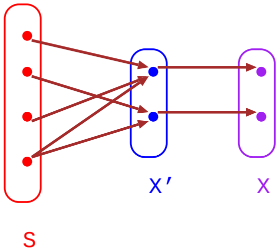
Yes there is!

- Input is a digraph $D = (V, A)$ and a set S of vertices (here $S = \{s\}$) that we will call source set. Want to disconnect T from S such that it helps in disconnecting other pairs.

- Closest set : A set $X \subseteq V$ is closest to S if X is the unique (S, X)-mincut . That is, the only cut of size at most $|X|$, for paths from S to X, is X itself. X is called closest set.

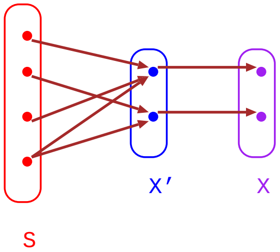
- Closest set of a set T: For any set of vertices T, the induced closest set $C(T)$ is the unique (S, T)-mincut which is closest to S. Clearly, if X is closest set then $C(X) = X$.

- S is the source set; X' is the closest set of X; X' is a closest set.



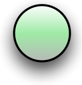
Example

- S is the source set; X' is the closest set of X; X' is a closest set.
- Analogy with important separators.



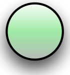
Improved Branching Algorithm

$$(D, s, T = \emptyset)$$

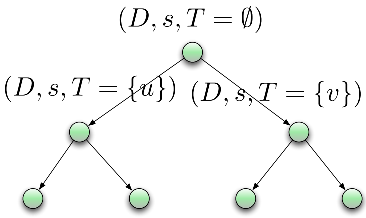


1. Initialise a set $T = \emptyset$

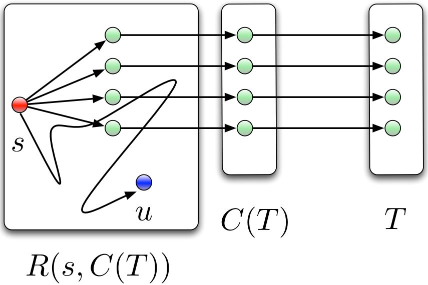
$$(D, s, T = \emptyset)$$



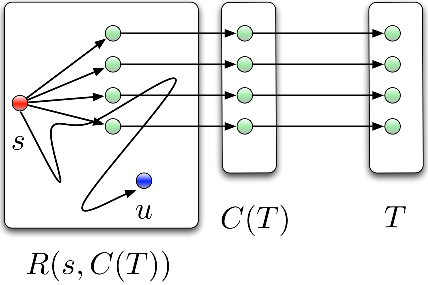
1. Initialise a set $T = \emptyset$
2. If the size of the (s, T)-mincut is at least $k + 1$, then we stop and say NO.
3. If there is an closest (s, T)-set $C(T)$ of size at most k such that no pairs of $\mathcal{P}$ are reachable from s, return YES.



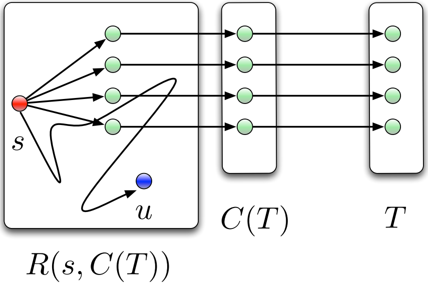
1. Initialise a set $T = \varnothing$
2. If the size of the (s, T)-mincut is at least $k + 1$, then we stop and say NO.
3. If there is an closest (s, T)-set $C(T)$ of size at most k such that no pairs of $\mathcal{P}$ are reachable from s, return YES.
4. Else, there is a pair $(u, v) \in \mathcal{P}$ which is reachable from s in $D \setminus C(T)$
5. Pick any such reachable pair and make a two-way branch for adding u or v to T. Return to step 2



- In iteration i let $C(T) = C$ be the closest (s, T) set and let $(u, v) \in \mathcal{P}$ be reachable from s in $D \setminus C$.
- Pick any branch (say the branch where u is picked in T). Any minimum cut C' of $(s, T \cup u)$ is also a cut for (s, T), so $|C'| \geqslant |C|$. Want to show $|C'| > |C|$

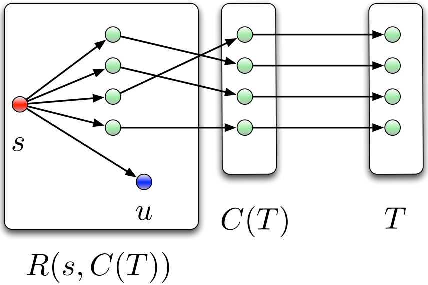


- Consider a mincut between s-$C \cup \{u\}$ in $D[R(s,T) \cup C]$ – say Z.



- Consider a mincut between s-$C \cup \{u\}$ in $D[R(s, T) \cup C]$ – say Z.

- Clearly $|Z| \geqslant |C|$. Suppose $|Z| = |C|$. Then clearly $Z \neq C$ – else it can not disconnect path from s to u. But then it contradicts that $C(T)$ is closest set to s.



- Consider a mincut between s-$C \cup \{u\}$ in $D[R(s,T) \cup C]$ – say Z.
- Suppose $|Z| > |C|$. Then there are $|Z| + 1$ internally vertex disjoint paths from s to $C \cup \{u\}$ in $D[R(s,T) \cup C]$.
- Using this we get that there are $|C| + 1$ internally vertex disjoint paths from s to $T \cup \{u\}$. Thus, $|C'| > |C|$.

Abstracting out a statement from the proof..

Let D be a digraph S and T be two vertex sets and $C(T)$ be the induced closest set. Furthermore, let $R(S, C(T))$ denotes the set of vertices that are reachable from S in $D \setminus C(T)$.

Abstracting out a statement from the proof..

Let D be a digraph S and T be two vertex sets and $C(T)$ be the induced closest set. Furthermore, let $R(S, C(T))$ denotes the set of vertices that are reachable from S in $D \setminus C(T)$. Then

> for every vertex $u \in R(S, C(T))$ we have that there are $|C| + 1$ vertex disjoint paths (internally vertex disjoint if $S = \{s\}$) from S to $C \cup \{v\}$ in $D[R(S, C(T)) \cup C(T)]$.

Analysis:

- If the algorithm finds a set of size at most k then that is a solution for the Digraph Pair Cut problem.

Analysis:

- If the algorithm finds a set of size at most k then that is a solution for the Digraph Pair Cut problem.
- Suppose the answer returned is NO. Can there be a solution set that the algorithm has missed? (Think about it!)

Analysis:

- If the algorithm finds a set of size at most k then that is a solution for the Digraph Pair Cut problem.
- Suppose the answer returned is NO. Can there be a solution set that the algorithm has missed? (Think about it!)
- Algorithm runs in $2^k n^{\mathcal{O}(1)}$ time.

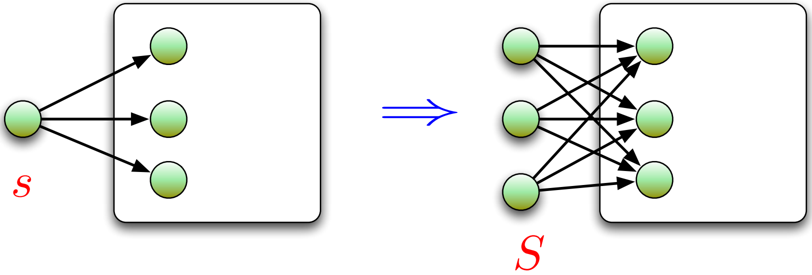
Digraph Pair Cut Problem: Kernel

- The number of pairs in the input set $\mathcal{P}$ could be as large as $\mathcal{O}(n^2)$.

Digraph Pair Cut Problem: Kernel

- The number of pairs in the input set $\mathcal{P}$ could be as large as $\mathcal{O}(n^2)$.
- Notice that if we have a solution X of size at most k, then the closest set $C(X)$ from s is also a solution.

First attempt



- Let $\mathcal{U}$ be the set of vertices that appear in pairs of $\mathcal{P}$. Need to make sure that we find a solution which does not contain s: we make $k+1$ copies of s (and give the same adjacencies) and call this set S the source set.

- Look at the gammoid (D, S, U) (source set $\mathcal{S} = S$ and sink set $\mathcal{T} = U$) and look at its representation matrix A.

- Look at the gammoid (D, S, U) (source set $\mathcal{S} = S$ and sink set $\mathcal{T} = U$) and look at its representation matrix A.
- Consider a subset of columns which correspond to a set W of vertices such that $\forall (u, v) \in \mathcal{P}, W \cap (u, v) \neq \emptyset$ and such that the rank of these columns is at most k, then we know that the minimum (S, W) cut is a solution to the Digraph Pair cut problem.

First attempt

- Look at the gammoid (D, S, U) (source set $\mathcal{S} = S$ and sink set $\mathcal{T} = U$) and look at its representation matrix A.
- Consider a subset of columns which correspond to a set W of vertices such that $\forall (u, v) \in \mathcal{P}, W \cap (u, v) \neq \emptyset$ and such that the rank of these columns is at most k, then we know that the minimum (S, W) cut is a solution to the Digraph Pair cut problem.
- But, since U could be a very large set, the representation matrix A could be large!

- Suppose we knew that the size of $\mathcal{P}$ was small, then the representation of the gammoid (D, S, U) is a compression for Digraph Pair cut .
- If $|\mathcal{P}|$ is very large, then we want to find a small subset of $\mathcal{P}$, such that making this set of pairs non-reachable is as good as making all pairs of $\mathcal{P}$ nonreachable.

- Suppose we knew that the size of $\mathcal{P}$ was small, then the representation of the gammoid (D, S, U) is a compression for Digraph Pair cut .
- If $|\mathcal{P}|$ is very large, then we want to find a small subset of $\mathcal{P}$, such that making this set of pairs non-reachable is as good as making all pairs of $\mathcal{P}$ nonreachable.

We SEEM to be looking for something like a representative set for the set $\mathcal{P}$ of pairs.

Keep a certificate for every k sized subset that tells why it can not a solution.

What does it mean?

Keep a certificate for every k sized subset that tells why it can not be a solution.

Consider Vertex Cover

What does it mean?

Keep a certificate for every k sized subset that tells why it can not be a solution.

Consider Vertex Cover

- What is a certificate that a particular k-sized subset is not a solution?

What does it mean?

Keep a certificate for every k sized subset that tells why it can not be a solution.

Consider Vertex Cover

- What is a certificate that a particular k-sized subset is not a solution?
- An edge that it does not cover – or intersects!

What does it mean?

Keep a certificate for every k sized subset that tells why it can not be a solution.

Consider Vertex Cover

- What is a certificate that a particular k-sized subset is not a solution?
- An edge that it does not cover – or intersects!
- So keep a subset of edges, say W, such that every for every k-sized subset that is not a solution there is a corresponding witness in W.

Keep a certificate for every k sized subset that tells why it can not be a solution.

Consider Vertex Cover

- What is a certificate that a particular k-sized subset is not a solution?
- An edge that it does not cover – or intersects!
- So keep a subset of edges, say W, such that every for every k-sized subset that is not a solution there is a corresponding witness in W.

Idea is to find this desired W using appropriate matroids.

What does it mean continues with refinement..?

- Sometimes we can also describe a potential solution by saying a subset of size at most k that looks like $\cdots$.

What does it mean continues with refinement..?

- Sometimes we can also describe a potential solution by saying a subset of size at most k that looks like $\cdots$.

Keep a certificate for every k sized subset that looks like $\cdots$ that tells why it can not be a solution.

What does it mean continues with refinement..?

- Sometimes we can also describe a potential solution by saying a subset of size at most k that looks like $\cdots$.

Idea is to find this desired W using appropriate matroids.

- The idea is to encode the desired witness as an independent set of an appropriate matroid. Clearly, the size of the solution + constraint gives a lower bound on the rank of the matroid.

Applying the idea to Digraph Pairs

Digraph D, vertex sets S and pairs $\mathcal{P}$.

Digraph D, vertex sets S and pairs $\mathcal{P}$.

- Solution are k-sized subset. In fact if X is a solution then induced closest set $C(X)$ is also a solution.

Digraph D, vertex sets S and pairs $\mathcal{P}$.

- Solution are k-sized subset. In fact if X is a solution then induced closest set $C(X)$ is also a solution.
- Let us keep witness for why a particular closest set X (of size at most k) to S is not a solution.

Digraph D, vertex sets S and pairs $\mathcal{P}$.

- Solution are k-sized subset. In fact if X is a solution then induced closest set $C(X)$ is also a solution.
- Let us keep witness for why a particular closest set X (of size at most k) to S is not a solution.
- A set X is not a solution because a pair $(u, v) \in \mathcal{P}$ is reachable from S in $D \setminus X$.

Applying the idea to Digraph Pairs

Digraph D, vertex sets S and pairs $\mathcal{P}$.

- Solution are k-sized subset. In fact if X is a solution then induced closest set $C(X)$ is also a solution.
- Let us keep witness for why a particular closest set X (of size at most k) to S is not a solution.
- A set X is not a solution because a pair $(u, v) \in \mathcal{P}$ is reachable from S in $D \setminus X$.
- So there are $|X| + 1$ vertex disjoint paths from S to $X \cup \{u\}$ in $D[R(S, X) \cup X]$ as well as $|X| + 1$ vertex disjoint paths from S to $X \cup \{v\}$ in $D[R(S, X) \cup X]$.

Digraph D, vertex sets S and pairs $\mathcal{P}$.

- Solution are k-sized subset. In fact if X is a solution then induced closest set $C(X)$ is also a solution.
- Let us keep witness for why a particular closest set X (of size at most k) to S is not a solution.
- A set X is not a solution because a pair $(u, v) \in \mathcal{P}$ is reachable from S in $D \setminus X$.
- So there are $|X| + 1$ vertex disjoint paths from S to $X \cup \{u\}$ in $D[R(S, X) \cup X]$ as well as $|X| + 1$ vertex disjoint paths from S to $X \cup \{v\}$ in $D[R(S, X) \cup X]$.

A closest set X is not a solution if and only if there exists a pair $(u, v) \in \mathcal{P}$ such that S is linked to $X \cup \{u\}$ and S is linked to $X \cup \{v\}$.

Digraph D, vertex sets S and pairs $\mathcal{P}$.

> A closest set X is not a solution if and only if there exists a pair $(u, v) \in \mathcal{P}$ such that S is linked to $X \cup \{u\}$ and S is linked to $X \cup \{v\}$.

So we encode this to get our desired $\mathcal{W}$.

Defining the problem in terms of a Matroid

- Build a matroid M, consisting of 2 disjoint copies of the gammoid (D, S). Call the first gammoid – M_1 – (D^1, S^1) and the second – M_2 – (D^2, S^2). Refer to all objects of gammoid i with superscript i. Thus, $M = M_1 \oplus M_2$.

- Build a matroid M, consisting of 2 disjoint copies of the gammoid (D, S). Call the first gammoid – M_1 – (D^1, S^1) and the second – M_2 – (D^2, S^2). Refer to all objects of gammoid i with superscript i. Thus, $M = M_1 \oplus M_2$.
- Let

$$\mathcal{P}_m = \{(u^1, v^2) \mid (u, v) \in \mathcal{P}\}.$$

Compute $2k$-representative for $\mathcal{P}_m$. There is a representative set $\widehat{\mathcal{P}}_m$ of $\mathcal{P}_m$ that extends all independent sets of M of size at most $2k$. Size of $\widehat{P}_m$ is at most $\mathcal{O}(k^2)$.

- Build a matroid M, consisting of 2 disjoint copies of the gammoid (D, S). Call the first gammoid – M_1 – (D^1, S^1) and the second – M_2 – (D^2, S^2). Refer to all objects of gammoid i with superscript i. Thus, $M = M_1 \oplus M_2$.
- Let
$$\mathcal{P}_m = \{(u^1, v^2) \mid (u, v) \in \mathcal{P}\}.$$
Compute $2k$-representative for $\mathcal{P}_m$. There is a representative set $\widehat{\mathcal{P}}_m$ of $\mathcal{P}_m$ that extends all independent sets of M of size at most $2k$. Size of $\widehat{P}_m$ is at most $\mathcal{O}(k^2)$.

Let $\mathcal{P}'$ be the set of pairs in $\mathcal{P}$ whose corresponding pairs are in $\widehat{P}_m$.

Finishing the proof...

Lemma

$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

Lemma

$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

$\Rightarrow$ Obvious as $\mathcal{P}' \subseteq \mathcal{P}$.

Lemma

$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

$\Leftarrow$ Let X be a solution to the problem – assume that X is a closest set to S.

Lemma

$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

$\Leftarrow$ Let X be a solution to the problem – assume that X is a closest set to S. If X is not a solution then there exists a pair $(u, v) \in \mathcal{P}$ such that S is linked to $X \cup \{u\}$ and S is linked to $X \cup \{v\}$.

Lemma

$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

$\Leftarrow$ Let X be a solution to the problem – assume that X is a closest set to S. If X is not a solution then there exists a pair $(u, v) \in \mathcal{P}$ such that S is linked to $X \cup \{u\}$ and S is linked to $X \cup \{v\}$. Since $\mathcal{P}'$ corresponds to $2k$ representative we have that there exists a pair $(u', v') \in \mathcal{P}'$ such that S is linked to $X \cup \{u'\}$ and S is linked to $X \cup \{v'\}$.

Lemma
$(G, \mathcal{P}, k)$ is a yes instance if and only if $(G, \mathcal{P}', k)$ is a yes instance.

$\Leftarrow$ Let X be a solution to the problem – assume that X is a closest set to S. If X is not a solution then there exists a pair $(u, v) \in \mathcal{P}$ such that S is linked to $X \cup \{u\}$ and S is linked to $X \cup \{v\}$. Since $\mathcal{P}'$ corresponds to $2k$ representative we have that there exists a pair $(u', v') \in \mathcal{P}'$ such that S is linked to $X \cup \{u'\}$ and S is linked to $X \cup \{v'\}$. Contradiction that X is a solution to $(G, \mathcal{P}', k)$!

Cut-Covering Problem

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

Clearly $Z = V(D)$ suffices!

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

Cut-Covering Problem
Input: A digraph D and vertex subsets S and T.
Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

It is not yet clear what this small should be. We will see at the end that it is
not too large.

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

- Which vertices must be in the set Z?

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

- Which vertices must be in the set Z?
- If a vertex w appears in every minimum cut between some $A \subseteq S$ and $B \subseteq T$, then it must be in Z.

Cut-Covering Problem
Input: A digraph D and vertex subsets S and T.
Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

- Which vertices must be in the set Z?
- If a vertex w appears in every minimum cut between some $A \subseteq S$ and $B \subseteq T$, then it must be in Z.
- These vertices are called essential vertices

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

- Which vertices must be in the set Z?
- If a vertex w appears in every minimum cut between some $A \subseteq S$ and $B \subseteq T$, then it must be in Z.
- These vertices are called essential vertices

We will show that just having these "essential vertices in Z are almost sufficient."

Cut-Covering Problem

Input: A digraph D and vertex subsets S and T.

Question: Find a set Z (as small as possible) such that for any $A \subseteq S, B \subseteq T$, Z contains a minimum (A, B)-cut.

- Which vertices must be in the set Z?

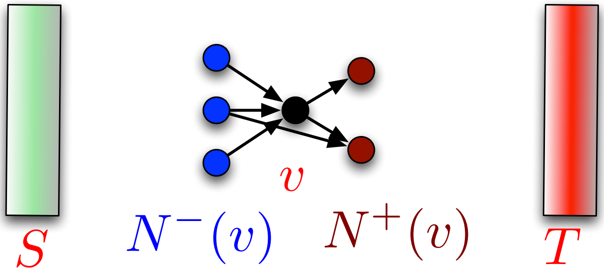
We will show that just having these "essential vertices in Z are almost sufficient."

More precisely we will show that (a) either all the vertices are essential; or (b) we can obtain an equivalent instance of the problem with strictly smaller number of vertices.

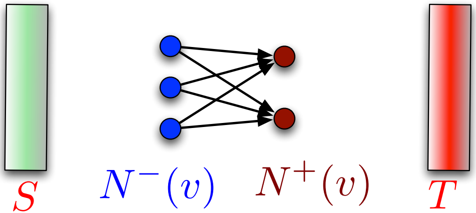
- Question 1: How to find the set of essential vertices?
- Question 2: If there are non-essential vertices then how do we obtain the equivalent instance.

- Question 1: How to find the set of essential vertices?
- Question 2: If there are non-essential vertices then how do we obtain the equivalent instance.

We first answer Question 2.



- Let v be a non-essential vertex.



Transformed Digraph

- Delete v and transform D to digraph D' such that there is a complete bipartite graph between the in-neighbours $N^-(v)$ and out-neighbours $N^+(v)$ of v, with edges directed from $N^-(v)$ to $N^+(v)$.

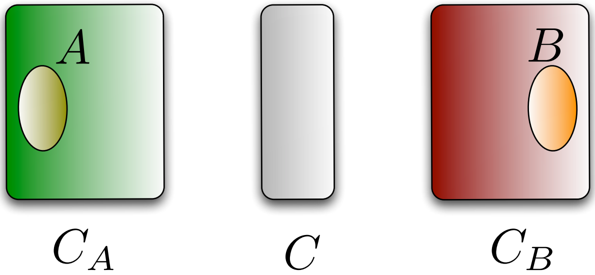
Want to argue that the size of minimum cuts remains exactly the same for D and D'. In fact, we show that a minimum cut in the new graph D' is actually a minimum cut in D itself.

Want to argue that the size of minimum cuts remains exactly the same for D and D'. In fact, we show that a minimum cut in the new graph D' is actually a minimum cut in D itself.

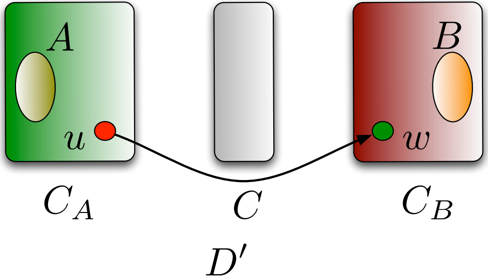
This implies our construction.

Since for every $A \subseteq S$ and $B \subseteq T$ there is a minimum cut that avoids v, we have that D and D' are equivalent instance of Cut-Covering Problem.

Bounding the cut in D'

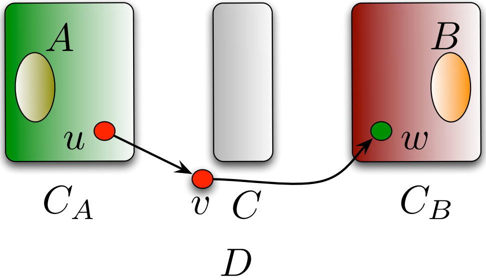


- Take a minimum cut C of (A, B) in D that did not contain v. Such a cut exists. Let C_A, C_B be the components containing A and B respectively in $D \setminus C$.

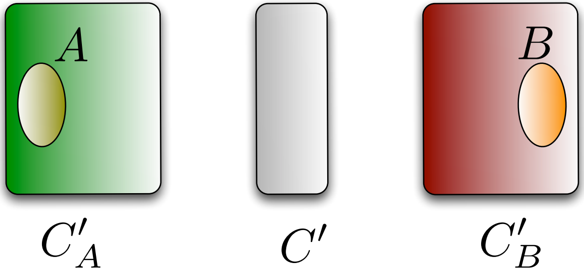


- Suppose this is not a cut of A, B in D'. This implies that the transformation introduced an edge from a vertex $u \in C_A$ to $w \in C_B$.
- This happens if $u \in N^-(v)$ and $w \in N^+(v)$.

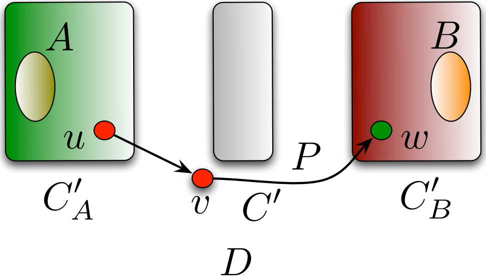
Bounding the cut in D'



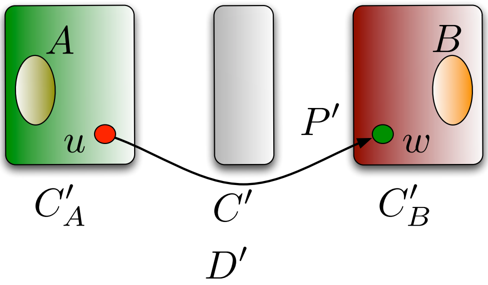
- This implies that there was a path from A to B through u, v, w in $D \setminus C$ (contradiction to C being an (A, B)-cut in D).
- So, for any (A, B) size of a minimum cut in D' is at most the size of a minimum cut in D.



- Take a cut C' of (A, B) in D'.



- Suppose this is not a cut of A, B in D. This implies there is a path P from A to B in $D \setminus C'$ and $v \in P$.
- This happens if $u \in N^{-}(v) \cap P$ and $w \in N^{+}(v) \cap P$ and $u, w \notin C'$.



- In D' there was an arc $a = (u, w)$ and a path $P' = PuawP$ from A to B avoiding C' (contradiction to C' being an (A, B)-cut on D').
- So, for any (A, B) size of a minimum cut in D' is equal to the size of a minimum cut in D.

- Start from the given graph D.

- Start from the given graph D.
- Iteratively throw out a nonessential vertex of the present graph and make the above transformation, that preserves the size of the minimum cut between any $A \subseteq S, B \subseteq T$.

Algorithm for finding set Z

- Start from the given graph D.
- Iteratively throw out a nonessential vertex of the present graph and make the above transformation, that preserves the size of the minimum cut between any $A \subseteq S, B \subseteq T$.
- Stop when there are no more nonessential vertices in the current graph.

- Notice that there may a nonessential vertex of D that became essential in one of the iterations.

Remarks

- Notice that there may a nonessential vertex of D that became essential in one of the iterations.
- An essential vertex remains essential throughout the algorithm: we showed that by the property of the transformation from D to D', any minimum cut of D' is a minimum cut of D.

Remarks

- Notice that there may a nonessential vertex of D that became essential in one of the iterations.
- An essential vertex remains essential throughout the algorithm: we showed that by the property of the transformation from D to D', any minimum cut of D' is a minimum cut of D.
- By the property of the transformation, the final graph contains a minimum cut in D for any A, B.

- Question 1: How to find the set of essential vertices?
- Question 2: If there are non-essential vertices then how do we obtain the equivalent instance.

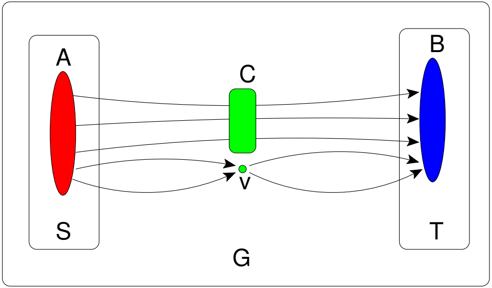
- Question 1: How to find the set of essential vertices?
- Question 2: If there are non-essential vertices then how do we obtain the equivalent instance.

We now answer Question 1.

Essential Vertices

- Recall that we have a directed graph $D = (V, E)$ and two sets of vertices S and T. A vertex is called essential for $A \subseteq S$ and $B \subseteq T$ if it occurs in every minimum (A, B) cut

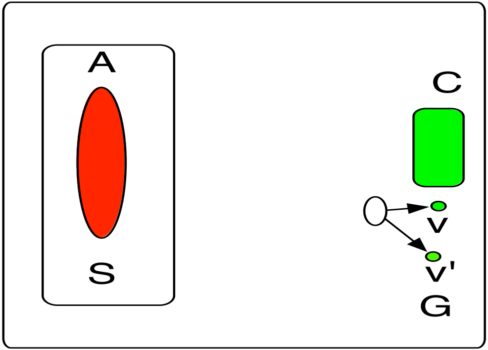
How do essential vertices look like



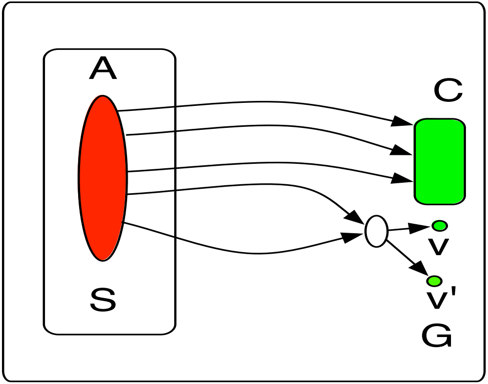
Lemma

Suppose that v is essential for A and B and let C be any minimum (A, B) cut. Then,

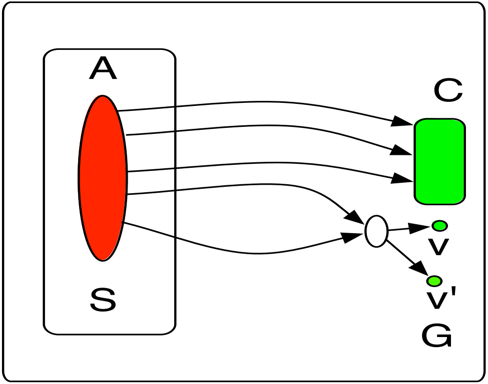
1. there is a set of $|C| + 1$ paths from A to C in $R(A, C)$ which are pairwise vertex disjoint, except for 2 of these paths which intersect in v and
2. there is a set of $|C| + 1$ paths from C to B in $NR(A, C)$ which are pairwise vertex disjoint, except for 2 of these paths which intersect in v.



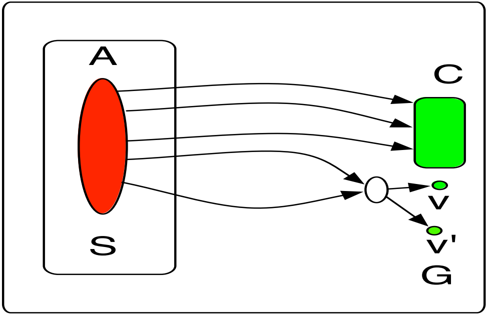
- Construct the graph G' by taking $G[R(A, C)] \cup C$ and adding a new vertex v' and adding all arcs from the in-neighborhood of v to v'.



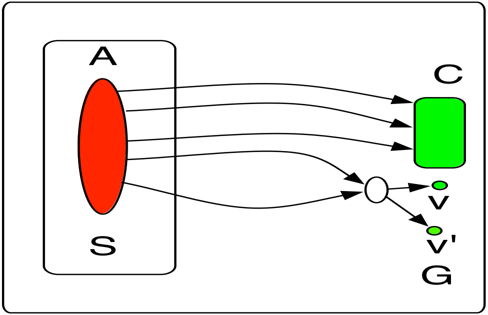
- What is the value of the maximum flow from A to $C \cup v'$ in G'?



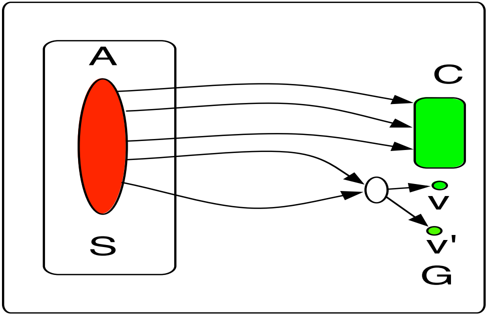
- What is the value of the maximum flow from A to $C \cup v'$ in G'?
- If this value is $|C| + 1$, then we are done!



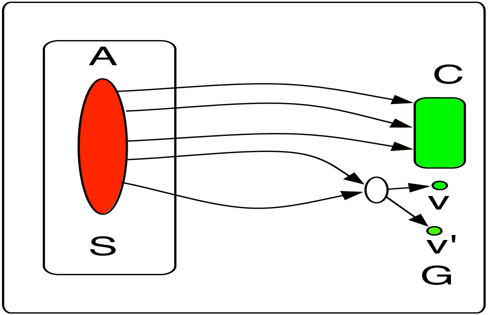
- Value of the max flow is not $|C| + 1 \implies$ an A-$(C \cup v')$ separator Z of size at most $|C|$.



- Value of the max flow is not $|C| + 1 \implies$ an A-$(C \cup v')$ separator Z of size at most $|C|$.
- If Z contains v and v', then at least one of the vertex disjoint paths from A to $C \setminus v$ is not hit. $\implies$ Z does not contain both v and v'.

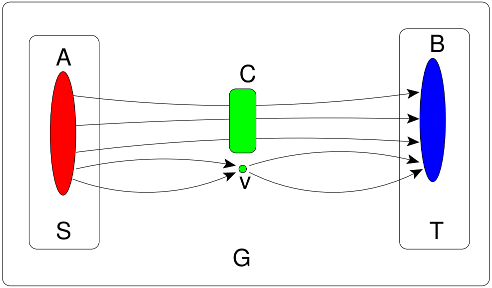


- Value of the max flow is not $|C|+1 \implies$ an A-$(C \cup v')$ separator Z of size at most $|C|$.
- If Z contains neither v nor v', then Z is a minimum (A, B) cut disjoint from $v \implies$ contradiction.

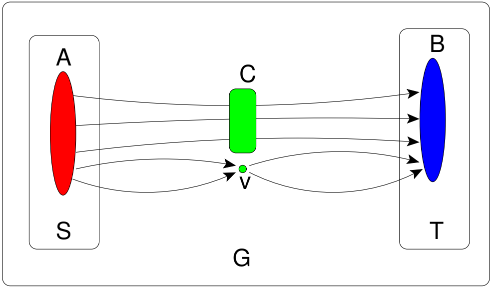


- Value of the max flow is not $|C| + 1 \implies$ an A-$(C \cup v')$ separator Z of size at most $|C|$.
- If Z contains v but not v', then v' is reachable from A in $G' \setminus Z \implies$ contradiction.

- Recall that we have a directed graph $G(V, E)$ and two sets of vertices S and T. A vertex is called essential if it occurs in every minimum (A, B) cut, for some $A \in S$ and $B \in T$.

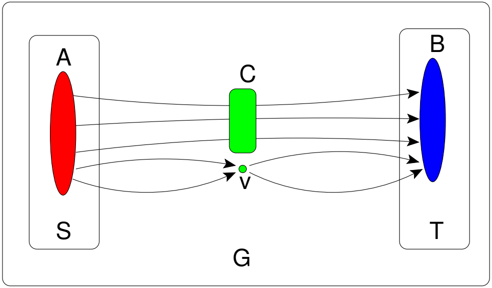


- Recall that we have a directed graph $G(V, E)$ and two sets of vertices S and T. A vertex is called essential if it occurs in every minimum (A, B) cut, for some $A \in S$ and $B \in T$.



- We wish to compute the set of essential vertices, Z in the graph G.

- Recall that we have a directed graph $G(V, E)$ and two sets of vertices S and T. A vertex is called essential if it occurs in every minimum (A, B) cut, for some $A \in S$ and $B \in T$.



- We wish to compute the set of essential vertices, Z in the graph G.
- It will be sufficient to compute a set $R(G)$ such that $Z \subseteq R(G)$, and $R(G)$ is of bounded size.

Observe the following :

Let r the size of the minimum (S, T) cut. Observe that the size of any (A, B) cut is bounded by r.

Observe the following :

Let r the size of the minimum (S, T) cut. Observe that the size of any (A, B) cut is bounded by r.

We will compute Z in the following manner :

- We will describe a linear matroid M.

Observe the following :

Let r the size of the minimum (S, T) cut. Observe that the size of any (A, B) cut is bounded by r.

We will compute Z in the following manner :

- We will describe a linear matroid M.
- Then we will describe a family $\mathcal{F}$ of independent sets of rank 3, such that each independent set corresponds to a vertex of G.

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

Constrcution of the Matroid M

The Matroid M is a direct-sum of the following three matroids.

- $M[0]$ is the uniform matroid of rank r.
  It is defined on the universe $V[0]$, where $V[0]$ is a copy of V. For a vertex $v \in V$, we will use $v[0]$ to denote the corresponding vertex in $V[0]$.

Constrcution of the Matroid $\mathcal{M}$

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S. It is defined as follows,

Constrcution of the Matroid $\mathcal{M}$

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

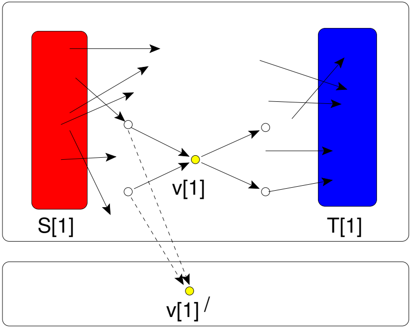
- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S. It is defined as follows,

- Let $G[1]$ be the graph constructed as follows,

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S. It is defined as follows,

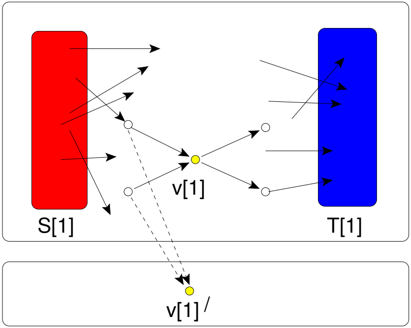
- Let $G[1]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[1]$ with vertex set $V[1]$.



The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S. It is defined as follows,

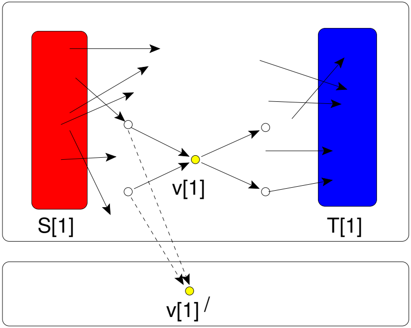
- Let $G[1]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[1]$ with vertex set $V[1]$.
  - Introduce a sink only vertex $v[1]'$ for every vertex $v[1]$.



The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S. It is defined as follows,

- Let $G[1]$ be the graph constructed as follows,
    - Make a copy of the graph G, called $G[1]$ with vertex set $V[1]$.
    - Introduce a sink only vertex $v[1]'$ for every vertex $v[1]$.
- $\mathcal{M}[1]$ is a gammoid $(V[1] \bigcup V[1]', I[1])$ where $I[1]$ consists of all the vertices linked to the set $S[1]$.



Constrcution of the Matroid $\mathcal{M}$

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S.
- $\mathcal{M}[2]$ is a gammoid defined using T. It is defined as follows,

Constrcution of the Matroid M

The Matroid M is a direct-sum of the following three matroids.

- $M[0]$ is the uniform matroid of rank r.
- $M[1]$ is a gammoid defined using S.
- $M[2]$ is a gammoid defined using T. It is defined as follows,

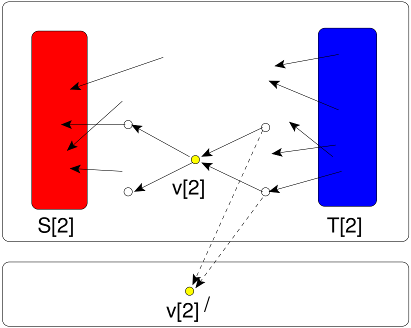
- Let $G[2]$ be the graph constructed as follows,

Constrcution of the Matroid M

The Matroid M is a direct-sum of the following three matroids.

- $M[0]$ is the uniform matroid of rank r.
- $M[1]$ is a gammoid defined using S.
- $M[2]$ is a gammoid defined using T. It is defined as follows,

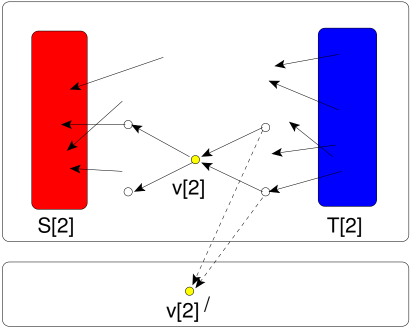
- Let $G[2]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[2]$ with vertex set $V[2]$.



The Matroid M is a direct-sum of the following three matroids.

- $M[0]$ is the uniform matroid of rank r.
- $M[1]$ is a gammoid defined using S.
- $M[2]$ is a gammoid defined using T. It is defined as follows,

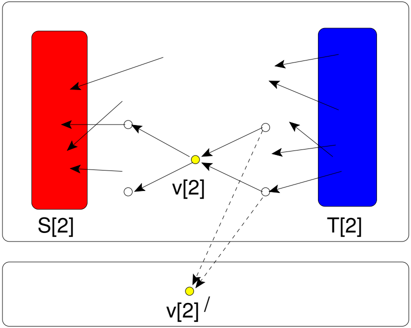
- Let $G[2]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[2]$ with vertex set $V[2]$.
  - Reverse every arc of $G[2]$.



The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S.
- $\mathcal{M}[2]$ is a gammoid defined using T. It is defined as follows,

- Let $G[2]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[2]$ with vertex set $V[2]$.
  - Reverse every arc of $G[2]$.
  - Introduce a sink only vertex $v[2]'$ for every $v[2]$.

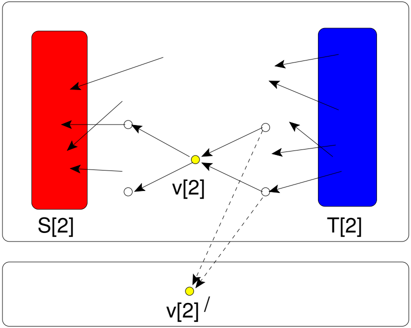


Constrcution of the Matroid $\mathcal{M}$

The Matroid $\mathcal{M}$ is a direct-sum of the following three matroids.

- $\mathcal{M}[0]$ is the uniform matroid of rank r.
- $\mathcal{M}[1]$ is a gammoid defined using S.
- $\mathcal{M}[2]$ is a gammoid defined using T. It is defined as follows,

- Let $G[2]$ be the graph constructed as follows,
  - Make a copy of the graph G, called $G[2]$ with vertex set $V[2]$.
  - Reverse every arc of $G[2]$.
  - Introduce a sink only vertex $v[2]'$ for every $v[2]$.
- $\mathcal{M}[2]$ is a gammoid $(V[2] \bigcup V[2]', I[2])$ where $I[2]$ consists of all the vertices linked to the set $T[2]$.



The Matroid M is a direct-sum of the following three matroids.

- $M[0]$ is the uniform matroid of rank r.
- $M[1]$ is a gammoid defined using S.
- $M[2]$ is a gammoid defined using T.
- Observe that the rank of M is $|S| + |T| + r$.

Proof: The family $\mathcal{F}$ and the set $R(G)$

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in $\mathcal{M}$.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.
- It is sufficient to show the following :

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.
- It is sufficient to show the following :

For every essential vertex q, there is a independent set $C_q \in M$ such that,

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.
- It is sufficient to show the following :

For every essential vertex q, there is a independent set $C_q \in M$ such that,

- $f(q)$ and C_q are disjoint, and $f(q) \cup C_q$ is an independent set in M.

Proof: The family $\mathcal{F}$ and the set $R(G)$

- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.
- It is sufficient to show the following :

For every essential vertex q, there is a independent set $C_q \in M$ such that,

- $f(q)$ and C_q are disjoint, and $f(q) \cup C_q$ is an independent set in M.
- For any other vertex $s \in V$,
  - either $f(s) \cup C_q$ is not independent,

Proof: The family $\mathcal{F}$ and the set $R(G)$

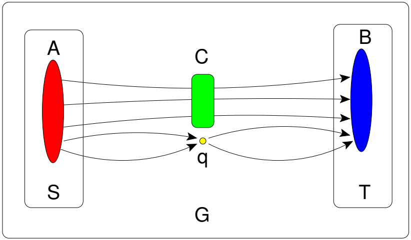
- The set $\mathcal{F}$ is defined as follows,
  - For vertex $v \in V$ let $f(v) = \{v[0], v[1]', v[2]'\}$.
  - Observe that, $f(v)$ is an independent set of rank 3 in M.
  - $\mathcal{F} = \{f(v) \mid v \in V \setminus (S \cup T)\}$.
- We compute $\widehat{\mathcal{F}}$ which is a $|S| + |T| + r - 3$ representative set for $\mathcal{F}$.
- Let $R(G) = \{v \in V | f(v) \in \widehat{\mathcal{F}}\}$
- We have to show that every essential vertex is in $R(G)$.
- It is sufficient to show the following :

For every essential vertex q, there is a independent set $C_q \in M$ such that,

- $f(q)$ and C_q are disjoint, and $f(q) \cup C_q$ is an independent set in M.
- For any other vertex $s \in V$,
  - either $f(s) \cup C_q$ is not independent,
  - or $f(s)$ and C_q are not disjoint.

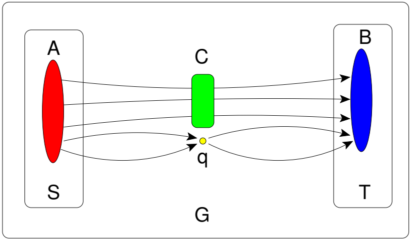
PROOF: THE SET C_q.

- Let q be an essential vertex in G, w.r.t $A \subseteq S$ and $B \subseteq T$. And let C be a A, B minumum cut.



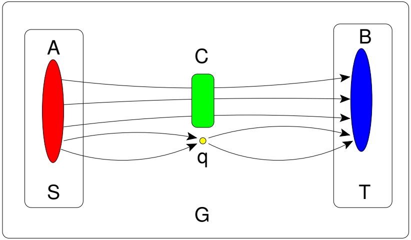
Proof: The set C_q.

- Let q be an essential vertex in G, w.r.t $A \subseteq S$ and $B \subseteq T$. And let C be a A, B minumum cut.
- Recall that by Proposition $|C| \leqslant |A|, |B|$.



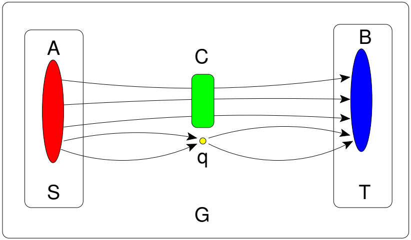
Proof: The set C_q.

- Let q be an essential vertex in G, w.r.t $A \subseteq S$ and $B \subseteq T$. And let C be a A, B minumum cut.
- Recall that by Proposition $|C| \leqslant |A|, |B|$.
- Let C_q be the union of $(C[0] \setminus \{q[0]\})$, $(S[1] \setminus A[1]) \cup C[1]$ and $(T[2] \setminus B[2]) \cup C[2]$.

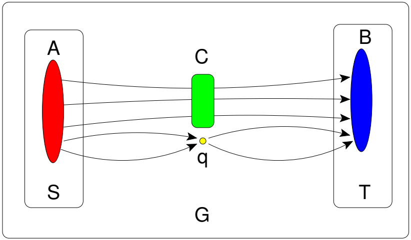


Proof: The set C_q.

- Let q be an essential vertex in G, w.r.t $A \subseteq S$ and $B \subseteq T$. And let C be a A, B minumum cut.
- Recall that by Proposition $|C| \leqslant |A|, |B|$.
- Let C_q be the union of $(C[0] \setminus \{q[0]\})$, $(S[1] \setminus A[1]) \cup C[1]$ and $(T[2] \setminus B[2]) \cup C[2]$.
- Observe that C_q is an independent set of rank at most $(|S| + |T| + r - 3)$.



- Observe that $f(q) = \{q[0], q[1]', q[2]'\}$ and C_q are disjoint.



Proof: The set C_q.

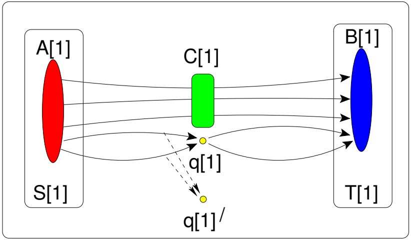
- We will show that $f(q) \cup C_q$ is an independent set.

Proof: The set C_q.

- We will show that $f(q) \cup C_q$ is an independent set.
  - $q[0] \cup (C[0] \setminus q[0])$ is independent in $M[0]$.

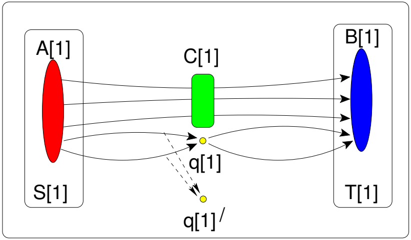
Proof: The set C_q.

- We will show that $f(q) \cup C_q$ is an independent set.
  - $q[0] \cup (C[0] \setminus q[0])$ is independent in $M[0]$.
  - By Proposition there are two vertex disjoint paths from A to q in $G \setminus (C \setminus \{q\})$.



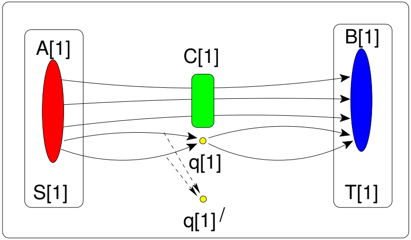
Proof: The set C_q.

- We will show that $f(q) \cup C_q$ is an independent set.
    - $q[0] \cup (C[0] \setminus q[0])$ is independent in $M[0]$.
    - By Proposition there are two vertex disjoint paths from A to q in $G \setminus (C \setminus \{q\})$. Therefore $(S[1] \setminus A[1]) \cup C[1] \cup \{q[1]'\}$ is independent in $M[1]$.



Proof: The set C_q.

- We will show that $f(q) \cup C_q$ is an independent set.
  - $q[0] \cup (C[0] \setminus q[0])$ is independent in $M[0]$.
  - By Proposition there are two vertex disjoint paths from A to q in $G \setminus (C \setminus \{q\})$. Therefore $(S[1] \setminus A[1]) \cup C[1] \cup \{q[1]'\}$ is independent in $M[1]$.
  - Similarly, $(T[2] \setminus B[2]) \cup C[2] \cup \{q[2]'\}$ is an independent set.



Proof: The set C_q.

- Now for any other vertex s, one of the following three cases happen,

Proof: The set C_q.

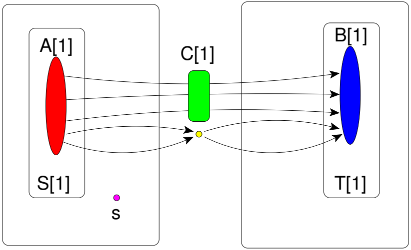
- Now for any other vertex s, one of the following three cases happen,
  - Either $s \in C$.

Proof: The set C_q.

- Now for any other vertex s, one of the following three cases happen,
  - Either $s \in C$.
    Therefore, $f(s)$ and C_q have $s[0]$ as a common element.

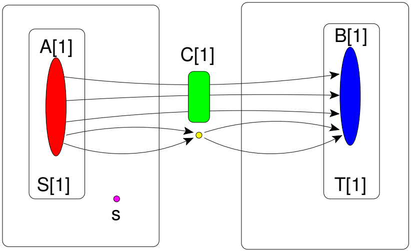
Proof: The set C_q.

- Now for any other vertex s, one of the following three cases happen,
    - Either $s \in C$.
    - Or s is not reachable from A in $G \setminus C$.



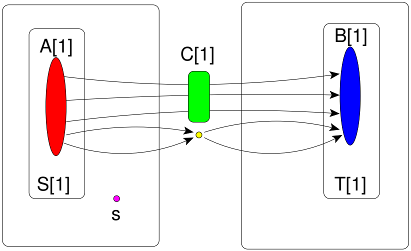
Proof: The set C_q.

- Now for any other vertex s, one of the following three cases happen,
  - Either $s \in C$.
  - Or s is not reachable from A in $G \setminus C$.
    Therefore, all paths from $A[1]$ to $q[1]'$ must pass through $C[1]$. So $f(s) \cup C_q$ is not an independent set.



Proof: The set C_q.

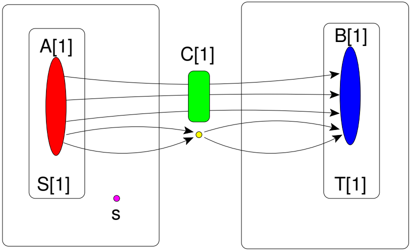
- Now for any other vertex s, one of the following three cases happen,
  - Either $s \in C$.
  - Or s is not reachable from A in $G \setminus C$.
  - Or s is not reachable from B in $G \setminus C$.



Proof: The set C_q.

- Now for any other vertex s, one of the following three cases happen,
  - Either $s \in C$.
  - Or s is not reachable from A in $G \setminus C$.
  - Or s is not reachable from B in $G \setminus C$.

    Therefore, all paths from $B[2]$ to $q[2]'$ must pass through $C[2]$. So $f(s) \cup C_q$ is not an independent set.



Proof: The set C_q.

- Therefore for every essential vertex q, $f(q)$ is present in $\widehat{\mathcal{F}}$ and q itself is present in $R(G)$.
- Since the size of $\widehat{\mathcal{F}}$ is bounded by $(|S| + |T| + r)^3$, we have that the size of $R(G)$ is bounded by the same quantity.

Theorem

Let G be a directed graph and $X \subseteq V$ a set of terminals. In polynomial time one can identify a set Z of $\mathcal{O}(|X|^3)$ vertices such that for any $S, T, R \subseteq X$, a minimum (S, T)-vertex cut in $G \setminus R$ is contained in Z.

Theorem

Let G be a directed graph and $X \subseteq V$ a set of terminals. In polynomial time one can identify a set Z of $\mathcal{O}(|X|^3)$ vertices such that for any $S, T, R \subseteq X$, a minimum (S, T)-vertex cut in $G \setminus R$ is contained in Z.

Odd Cycle Transversal: Get Q – the approximate solution of size $\mathcal{O}(k^{1.5})$ and compute Z. Delete all the vertices of $G \setminus Z$ and take parity torso for Z. Return this as an equivalent instance.

Theorem

Let G be a directed graph and $X \subseteq V$ a set of terminals. In polynomial time one can identify a set Z of $\mathcal{O}(|X|^3)$ vertices such that for any $S, T, R \subseteq X$, a minimum (S, T)-vertex cut in $G \setminus R$ is contained in Z.

Odd Cycle Transversal: Get Q – the approximate solution of size $\mathcal{O}(k^{1.5})$ and compute Z. Delete all the vertices of $G \setminus Z$ and take parity torso for Z. Return this as an equivalent instance.

Digraph Pair

Exercise :D

Final Slide

Thank You!

Any Questions?