The Feedback Arc Set Problem

Problem Definition and Discussions on Hardness

Rayhan Rashed (1505006) Nishat Anjum Bristy (1505007)
Tawhidul Hasan (1505008) Sadia Afroz (1505030)
Mursalin Habib (1505049)

Bangladesh University of Engineering and Technology

March 12, 2020

The Problem

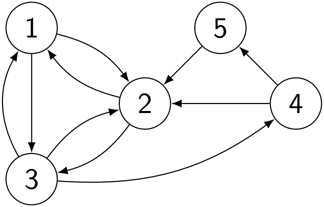
- The problem we will be discussing today is called the feedback arc set problem.
- What is a feedback arc set?

The Problem

- The problem we will be discussing today is called the feedback arc set problem.
- What is a feedback arc set?

The Problem

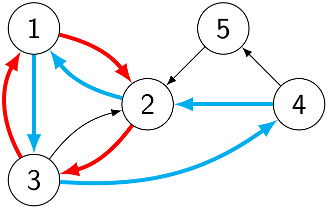
- Consider a directed graph possibly containing one or more cycles.



- What we want to do is remove these cycles.
- The way we do it is through arc deletions.

The Problem

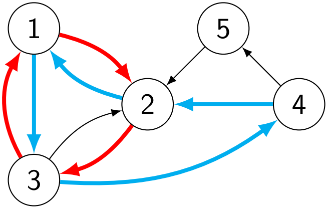
- Consider a directed graph possibly containing one or more cycles.



- What we want to do is remove these cycles.
- The way we do it is through arc deletions.

The Problem

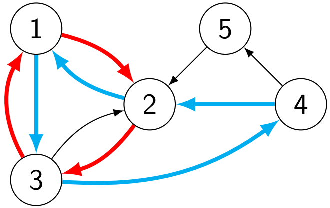
- Consider a directed graph possibly containing one or more cycles.



- What we want to do is remove these cycles.
- The way we do it is through arc deletions.

The Problem

- Consider a directed graph possibly containing one or more cycles.



- What we want to do is remove these cycles.
- The way we do it is through arc deletions.

What is a Feedback Arc Set?

A feedback arc set of a directed graph is a set of arcs whose removal leave the graph acyclic. More formally,

Definition (Feedback Arc Set)

Given a directed graph $G = (V, A)$, a feedback arc set $F \subseteq A$ is a set of arcs such that $G - F$ is acyclic.

What is a Feedback Arc Set?

A feedback arc set of a directed graph is a set of arcs whose removal leave the graph acyclic. More formally,

Definition (Feedback Arc Set)

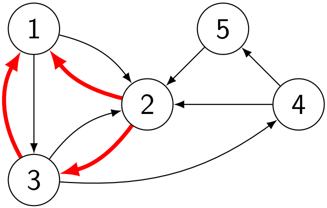
Given a directed graph $G = (V, A)$, a feedback arc set $F \subseteq A$ is a set of arcs such that $G - F$ is acyclic.

What is a Feedback Arc Set?

A feedback arc set of a directed graph is a set of arcs whose removal leave the graph acyclic. More formally,

Definition (Feedback Arc Set)

Given a directed graph $G = (V, A)$, a feedback arc set $F \subseteq A$ is a set of arcs such that $G - F$ is acyclic.

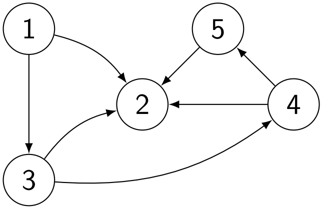


What is a Feedback Arc Set?

A feedback arc set of a directed graph is a set of arcs whose removal leave the graph acyclic. More formally,

Definition (Feedback Arc Set)

Given a directed graph $G = (V, A)$, a feedback arc set $F \subseteq A$ is a set of arcs such that $G - F$ is acyclic.



Problem Definition

- A natural thing we might want to do is the following:

Definition (The Feedback Arc Set Problem (Optimization Version))

Given a directed graph $G = (V, A)$, find a minimum cardinality feedback arc set of G.

- However, for this presentation, we will only be interested in the following decision version:

Definition (The Feedback Arc Set Problem (Decision Version))

Given a directed graph $G = (V, A)$ and an integer k, does G have a feedback arc set of size at most k?

Problem Definition

- A natural thing we might want to do is the following:

Definition (The Feedback Arc Set Problem (Optimization Version))

Given a directed graph $G = (V, A)$, find a minimum cardinality feedback arc set of G.

- However, for this presentation, we will only be interested in the following decision version:

Definition (The Feedback Arc Set Problem (Decision Version))

Given a directed graph $G = (V, A)$ and an integer k, does G have a feedback arc set of size at most k?

Problem Definition

- A natural thing we might want to do is the following:

Definition (The Feedback Arc Set Problem (Optimization Version))

Given a directed graph $G = (V, A)$, find a minimum cardinality feedback arc set of G.

- However, for this presentation, we will only be interested in the following decision version:

Definition (The Feedback Arc Set Problem (Decision Version))

Given a directed graph $G = (V, A)$ and an integer k, does G have a feedback arc set of size at most k?

Problem Definition

- A natural thing we might want to do is the following:

Definition (The Feedback Arc Set Problem (Optimization Version))

Given a directed graph $G = (V, A)$, find a minimum cardinality feedback arc set of G.

- However, for this presentation, we will only be interested in the following decision version:

Definition (The Feedback Arc Set Problem (Decision Version))

Given a directed graph $G = (V, A)$ and an integer k, does G have a feedback arc set of size at most k?

Motivations and Applications

- The first motivation for studying the feedback arc set problem is purely theoretical. It is an interesting problem in its own right.
- Has applications in combinatorial circuit design, where cycles can potentially lead to race conditions.
- Most problems that can be modeled as feedback vertex set problems can also be modeled as feedback arc set problems.
- Can help in deadlock prevention in computer systems.

Motivations and Applications

- The first motivation for studying the feedback arc set problem is purely theoretical. It is an interesting problem in its own right.
- Has applications in combinatorial circuit design, where cycles can potentially lead to race conditions.
- Most problems that can be modeled as feedback vertex set problems can also be modeled as feedback arc set problems.
- Can help in deadlock prevention in computer systems.

Motivations and Applications

- The first motivation for studying the feedback arc set problem is purely theoretical. It is an interesting problem in its own right.
- Has applications in combinatorial circuit design, where cycles can potentially lead to race conditions.
- Most problems that can be modeled as feedback vertex set problems can also be modeled as feedback arc set problems.
- Can help in deadlock prevention in computer systems.

Motivations and Applications

- The first motivation for studying the feedback arc set problem is purely theoretical. It is an interesting problem in its own right.
- Has applications in combinatorial circuit design, where cycles can potentially lead to race conditions.
- Most problems that can be modeled as feedback vertex set problems can also be modeled as feedback arc set problems.
- Can help in deadlock prevention in computer systems.

Hardness Status of the Feedback Arc Set Problem

- The decision version the feedback arc set problem turns out to be *NP*-complete.
- One of the first 21 problems to be proved *NP*-complete (Karp, 1972)[1].
- Here we basically duplicate Karp's result.

Hardness Status of the Feedback Arc Set Problem

- The decision version the feedback arc set problem turns out to be *NP*-complete.
- One of the first 21 problems to be proved *NP*-complete (Karp, 1972)[1].
- Here we basically duplicate Karp's result.

Hardness Status of the Feedback Arc Set Problem

- The decision version the feedback arc set problem turns out to be *NP*-complete.
- One of the first 21 problems to be proved *NP*-complete (Karp, 1972)[1].
- Here we basically duplicate Karp's result.

Proof of *NP*-completeness

- To prove that feedback arc set is *NP*-complete, first we prove that it is in *NP*.
- Given a set (of size at most k) of arcs , we can check if it is a feedback arc set by first deleting the arcs and then running depth-first search to check if there are any cycles left in the graph.
- Clearly, this takes linear time. So, the feedback arc set problem is clearly in *NP*.

Proof of *NP*-completeness

- To prove that feedback arc set is *NP*-complete, first we prove that it is in *NP*.
- Given a set (of size at most k) of arcs , we can check if it is a feedback arc set by first deleting the arcs and then running depth-first search to check if there are any cycles left in the graph.
- Clearly, this takes linear time. So, the feedback arc set problem is clearly in *NP*.

Proof of *NP*-completeness

- To prove that feedback arc set is *NP*-complete, first we prove that it is in *NP*.
- Given a set (of size at most k) of arcs , we can check if it is a feedback arc set by first deleting the arcs and then running depth-first search to check if there are any cycles left in the graph.
- Clearly, this takes linear time. So, the feedback arc set problem is clearly in *NP*.

Proof of *NP*-completeness

- The next thing we do is show that it is *NP*-hard.
- To do so, we show that the feedback arc set problem is at least as hard as the vertex cover problem.

Claim

$$\text{Vertex Cover} \leq_p \text{Feedback Arc Set}$$

Proof of *NP*-completeness

- The next thing we do is show that it is *NP*-hard.
- To do so, we show that the feedback arc set problem is at least as hard as the vertex cover problem.

Claim

VERTEX COVER $\leq_p$ FEEDBACK ARC SET

Proof of *NP*-completeness

- The next thing we do is show that it is *NP*-hard.
- To do so, we show that the feedback arc set problem is at least as hard as the vertex cover problem.

Claim

VERTEX COVER $\leq_p$ FEEDBACK ARC SET

Reduction from Vertex Cover

- Why vertex cover?
- **The Not Illuminating Answer:** That is how *"Reducibility Among Combinatorial Problems"* did it.
- **A Slightly More Insightful Answer:** FAS is a "cover" problem. It makes sense to reduce another "cover" problem to it.

Reduction from Vertex Cover

- Why vertex cover?
- **The Not Illuminating Answer:** That is how *"Reducibility Among Combinatorial Problems"* did it.
- **A Slightly More Insightful Answer:** FAS is a "cover" problem. It makes sense to reduce another "cover" problem to it.

Reduction from Vertex Cover

- Why vertex cover?
- **The Not Illuminating Answer:** That is how *"Reducibility Among Combinatorial Problems"* did it.
- **A Slightly More Insightful Answer:** FAS is a "cover" problem. It makes sense to reduce another "cover" problem to it.

Reduction from Vertex Cover

- In the vertex cover problem, we want to cover all the *edges* with *vertices*.
- In the feedback arc set problem, we want to cover all the *cycles* with *arcs*.
- So, in the reduction process, there should be some sort of correspondence between edges and cycles i.e. *the things we want to cover*, and vertices and arcs i.e. *the things that we want to cover with*.

Reduction from Vertex Cover

- In the vertex cover problem, we want to cover all the *edges* with *vertices*.
- In the feedback arc set problem, we want to cover all the *cycles* with *arcs*.
- So, in the reduction process, there should be some sort of correspondence between edges and cycles i.e. *the things we want to cover*, and vertices and arcs i.e. *the things that we want to cover with.*

Reduction from Vertex Cover

- In the vertex cover problem, we want to cover all the *edges* with *vertices*.
- In the feedback arc set problem, we want to cover all the *cycles* with *arcs*.
- So, in the reduction process, there should be some sort of correspondence between edges and cycles i.e. *the things we want to cover*, and vertices and arcs i.e. *the things that we want to cover with*.

Reduction from Vertex Cover

- An instance of the vertex cover problem is an undirected graph $G = (V, E)$ and an integer k.
- An instance of the feedback arc set problem is a directed graph $G' = (V', A')$ and an integer k'.
- Given any vertex cover instance (G, k), we have to construct from it a feedback arc set instance in polynomial time. Here is how we do it.

Reduction from Vertex Cover

- An instance of the vertex cover problem is an undirected graph $G = (V, E)$ and an integer k.
- An instance of the feedback arc set problem is a directed graph $G' = (V', A')$ and an integer k'.
- Given any vertex cover instance (G, k), we have to construct from it a feedback arc set instance in polynomial time. Here is how we do it.

Reduction from Vertex Cover

- An instance of the vertex cover problem is an undirected graph $G = (V, E)$ and an integer k.
- An instance of the feedback arc set problem is a directed graph $G' = (V', A')$ and an integer k'.
- Given any vertex cover instance (G, k), we have to construct from it a feedback arc set instance in polynomial time. Here is how we do it.

Reduction from Vertex Cover

Step 1 (Creation of "Vertex Arcs")

Split every vertex in G in two i.e. for each vertex $v \in V$, add two vertices v_0 and v_1 in G'. Then add the arc (v_0, v_1).

- We call the arc (v_0, v_1) the arc "corresponding to vertex v".

Reduction from Vertex Cover

Step 1 (Creation of "Vertex Arcs")

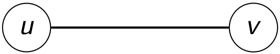
Split every vertex in G in two i.e. for each vertex $v \in V$, add two vertices v_0 and v_1 in G'. Then add the arc (v_0, v_1).

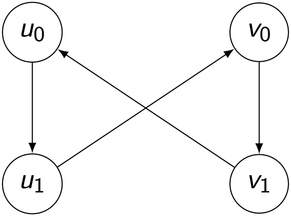
- We call the arc (v_0, v_1) the arc "corresponding to vertex v".

Reduction from Vertex Cover

Step 2 (Creation of "Edge Cycles")

For each edge $e = \{u, v\}$ in E, add the arcs (u_1, v_0) and (v_1, u_0) in A'.



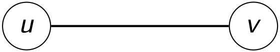


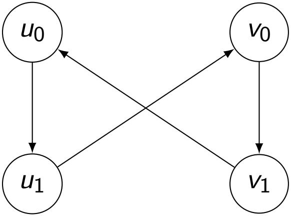
- Note how every edge in G corresponds to a directed cycle of length 4 in G'.

Reduction from Vertex Cover

Step 2 (Creation of "Edge Cycles")

For each edge $e = \{u, v\}$ in E, add the arcs (u_1, v_0) and (v_1, u_0) in A'.





- Note how every edge in G corresponds to a directed cycle of length 4 in G'.

Reduction from Vertex Cover

Step 3 (Set k')

Let $k' = k$.

Clearly, this construction is polynomial.

Claim

G contains a vertex cover of size at most k if and only if G' contains a feedback arc set of size of size at most k.

Reduction from Vertex Cover

Step 3 (Set k')

Let $k' = k$.

Clearly, this construction is polynomial.

Claim

G contains a vertex cover of size at most k if and only if G' contains a feedback arc set of size of size at most k.

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- The proof is in two parts.
- First we show that if G has a vertex cover of size at most k, then G' has a feedback arc set of size at most k.
- Let S be a vertex cover of G with $|S| \leq k$.
- Then $F = \{(v_0, v_1) : v \in S\}$ is a feedback arc set of G'.
- If not, then $G' - F$ contains at least one cycle.
- That cycle uses some arc of the form (u_1, v_0).
- Any cycle that uses the arc (u_1, v_0) has to use both of the arcs (u_0, u_1) and (v_0, v_1).
- So, for such a cycle to exist, $\{(u_0, u_1), (v_0, v_1)\} \cap F = \emptyset$. Therefore, $\{u, v\} \cap S = \emptyset$, a contradiction!

Reduction from Vertex Cover

- Now we prove the opposite direction. If G' contains a feedback arc set of size at most k, then G contains a vertex cover of size at most k.
- Let F be a feedback arc set of G' with $|F| \leq k$.
- Without loss of generality, F only contains arcs of the form (v_0, v_1).
- This is because any cycle that an arc of the form (u_1, v_0) participates in also contains the arc (v_0, v_1). So, deleting the arc (v_0, v_1) instead of the arc (u_1, v_0) removes at least as many cycles (if not more).

Reduction from Vertex Cover

- Now we prove the opposite direction. If G' contains a feedback arc set of size at most k, then G contains a vertex cover of size at most k.
- Let F be a feedback arc set of G' with $|F| \leq k$.
- Without loss of generality, F only contains arcs of the form (v_0, v_1).
- This is because any cycle that an arc of the form (u_1, v_0) participates in also contains the arc (v_0, v_1). So, deleting the arc (v_0, v_1) instead of the arc (u_1, v_0) removes at least as many cycles (if not more).

Reduction from Vertex Cover

- Now we prove the opposite direction. If G' contains a feedback arc set of size at most k, then G contains a vertex cover of size at most k.
- Let F be a feedback arc set of G' with $|F| \leq k$.
- Without loss of generality, F only contains arcs of the form (v_0, v_1).
- This is because any cycle that an arc of the form (u_1, v_0) participates in also contains the arc (v_0, v_1). So, deleting the arc (v_0, v_1) instead of the arc (u_1, v_0) removes at least as many cycles (if not more).

Reduction from Vertex Cover

- Now we prove the opposite direction. If G' contains a feedback arc set of size at most k, then G contains a vertex cover of size at most k.
- Let F be a feedback arc set of G' with $|F| \leq k$.
- Without loss of generality, F only contains arcs of the form (v_0, v_1).
- This is because any cycle that an arc of the form (u_1, v_0) participates in also contains the arc (v_0, v_1). So, deleting the arc (v_0, v_1) instead of the arc (u_1, v_0) removes at least as many cycles (if not more).

Reduction from Vertex Cover

- The vertices corresponding to the arcs in F constitute a vertex cover in G.
- If not, then some edge $\{u, v\}$ of G will be uncovered.
- As a result, F will not contain any arc from the length-4 cycle u_0, u_1, v_0, v_1, u_0, a contradiction!

Reduction from Vertex Cover

- The vertices corresponding to the arcs in F constitute a vertex cover in G.
- If not, then some edge $\{u, v\}$ of G will be uncovered.
- As a result, F will not contain any arc from the length-4 cycle u_0, u_1, v_0, v_1, u_0, a contradiction!

Reduction from Vertex Cover

- The vertices corresponding to the arcs in F constitute a vertex cover in G.
- If not, then some edge $\{u, v\}$ of G will be uncovered.
- As a result, F will not contain any arc from the length-4 cycle u_0, u_1, v_0, v_1, u_0, a contradiction!

Feedback Arc Set is *NP*-complete!

Theorem

The feedback arc set problem is NP-complete.

Feedback Arc Set on Special Classes of Graphs

- We now know the feedback arc set problem is *NP*-complete on general graphs.
- But what if we restrict our attention only to special classes of graphs?.
- Does the problem become any easier?

Feedback Arc Set on Special Classes of Graphs

- We now know the feedback arc set problem is *NP*-complete on general graphs.
- But what if we restrict our attention only to special classes of graphs?.
- Does the problem become any easier?

Feedback Arc Set on Special Classes of Graphs

- We now know the feedback arc set problem is *NP*-complete on general graphs.
- But what if we restrict our attention only to special classes of graphs?.
- Does the problem become any easier?

Feedback Arc Set Problem on Undirected Graphs

- The first thing we do is consider the undirected variant of the problem.
- This can be solved in polynomial time.
- Reduces to finding a spanning tree of the given graph.

Feedback Arc Set Problem on Undirected Graphs

- The first thing we do is consider the undirected variant of the problem.
- This can be solved in polynomial time.
- Reduces to finding a spanning tree of the given graph.

Feedback Arc Set Problem on Undirected Graphs

- The first thing we do is consider the undirected variant of the problem.
- This can be solved in polynomial time.
- Reduces to finding a spanning tree of the given graph.

Feedback Arc Set Problem on Tournaments

- One important class of directed graphs are tournaments.
- Tournaments are orientations of complete graphs.
- The feedback arc set problem on tournaments (FAST) turns out to be *NP*-complete as well.
- The proof, however, is extremely non-trivial.

Feedback Arc Set Problem on Tournaments

- One important class of directed graphs are tournaments.
- Tournaments are orientations of complete graphs.
- The feedback arc set problem on tournaments (FAST) turns out to be *NP*-complete as well.
- The proof, however, is extremely non-trivial.

Feedback Arc Set Problem on Tournaments

- One important class of directed graphs are tournaments.
- Tournaments are orientations of complete graphs.
- The feedback arc set problem on tournaments (FAST) turns out to be *NP*-complete as well.
- The proof, however, is extremely non-trivial.

Feedback Arc Set Problem on Tournaments

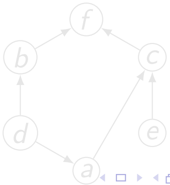
- One important class of directed graphs are tournaments.
- Tournaments are orientations of complete graphs.
- The feedback arc set problem on tournaments (FAST) turns out to be *NP*-complete as well.
- The proof, however, is extremely non-trivial.

Feedback Arc Set Problem on Directed Line Graphs

- Now we turn our attention to a different class of directed graphs called directed line graphs.

Definition (Directed Line Graphs)

Given a directed graph G, we compute its directed line graph $\overline{G}$ as follows: first for every arc $a = (u, v)$ of G, we add a vertex labelled (u, v) in $\overline{G}$. Then we add an arc between the two vertices (u, v) and (w, x) of $\overline{G}$ if and only if $v = w$.

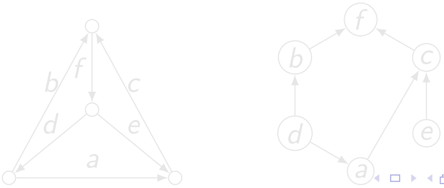


Feedback Arc Set Problem on Directed Line Graphs

- Now we turn our attention to a different class of directed graphs called directed line graphs.

Definition (Directed Line Graphs)

Given a directed graph G, we compute its directed line graph $\overline{G}$ as follows: first for every arc $a = (u, v)$ of G, we add a vertex labelled (u, v) in $\overline{G}$. Then we add an arc between the two vertices (u, v) and (w, x) of $\overline{G}$ if and only if $v = w$.

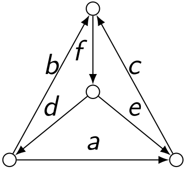


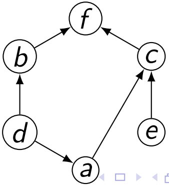
Feedback Arc Set Problem on Directed Line Graphs

- Now we turn our attention to a different class of directed graphs called directed line graphs.

Definition (Directed Line Graphs)

Given a directed graph G, we compute its directed line graph $\overline{G}$ as follows: first for every arc $a = (u, v)$ of G, we add a vertex labelled (u, v) in $\overline{G}$. Then we add an arc between the two vertices (u, v) and (w, x) of $\overline{G}$ if and only if $v = w$.





Feedback Arc Set Problem on Directed Line Graphs

- Clearly, not all directed graphs are directed line graphs.
- However, it is possible to show that even if we restrict our attention to only directed line graphs, the feedback arc set problem remains *NP*-complete (Gavril, 1977)[2].
- This is because of the following chain of reductions.

Feedback Arc set is *NP*-hard on directed line graphs

$$\text{FAS} \leq_p \text{FVS ON DIRECTED LINE GRAPHS} \leq_p \text{FAS ON DIRECTED LINE GRAPHS}$$

Feedback Arc Set Problem on Directed Line Graphs

- Clearly, not all directed graphs are directed line graphs.
- However, it is possible to show that even if we restrict our attention to only directed line graphs, the feedback arc set problem remains *NP*-complete (Gavril, 1977)[2].
- This is because of the following chain of reductions.

Feedback Arc set is *NP*-hard on directed line graphs

$$\text{FAS} \leq_p \text{FVS ON DIRECTED LINE GRAPHS} \leq_p \text{FAS ON DIRECTED LINE GRAPHS}$$

Feedback Arc Set Problem on Directed Line Graphs

- Clearly, not all directed graphs are directed line graphs.
- However, it is possible to show that even if we restrict our attention to only directed line graphs, the feedback arc set problem remains *NP*-complete (Gavril, 1977)[2].
- This is because of the following chain of reductions.

Feedback Arc set is *NP*-hard on directed line graphs

$$\text{FAS} \leq_p \text{FVS ON DIRECTED LINE GRAPHS} \leq_p \text{FAS ON DIRECTED LINE GRAPHS}$$

Feedback Arc Set Problem on Directed Line Graphs

- Clearly, not all directed graphs are directed line graphs.
- However, it is possible to show that even if we restrict our attention to only directed line graphs, the feedback arc set problem remains *NP*-complete (Gavril, 1977)[2].
- This is because of the following chain of reductions.

Feedback Arc set is *NP*-hard on directed line graphs

$$\text{FAS} \leq_p \text{FVS ON DIRECTED LINE GRAPHS} \leq_p \text{FAS ON DIRECTED LINE GRAPHS}$$

Feedback Arc Set Problem on Graphs with Small Cliques

- A directed line graph can only have cliques of size at most three.
- Therefore,the feedback arc set problem is *NP*-hard on graphs whose underlying undirected graph has clique number at most three.

Feedback Arc Set Problem on Graphs with Small Cliques

- A directed line graph can only have cliques of size at most three.
- Therefore,the feedback arc set problem is NP-hard on graphs whose underlying undirected graph has clique number at most three.

Feedback Arc Set Problem on Planar Graphs

- We end this section with a positive result.
- If we restrict our instances to only *planar* directed graphs, the feedback arc set problem can be solved in polynomial time (Lucchesi, 1976) [3].

Feedback Arc Set Problem on Planar Graphs

- We end this section with a positive result.
- If we restrict our instances to only *planar* directed graphs, the feedback arc set problem can be solved in polynomial time (Lucchesi, 1976) [3].

Hardness Results Summary

We summarize the results on hardness in the following table.

Problem	Status	Comments
FAS	in *NP*-complete	Reduces from vertex cover. See Karp (1972).
FAS on Undirected Graphs	in *P*	Reduces to finding spanning trees.
FAS on Tournaments	*NP*-complete	See Charbit, Thomasse and Yeo (2007).
FAS on Directed Line Graphs	*NP*-complete	See Gavril (1977)[2]
FAS on Graphs with Clique Number ≤ 3	*NP*-complete	Follows from the fact that directed line graphs have clique number at most 3.
FAS on planar directed graphs	in *P*	See Lucchesi (1976).

Problems that Feedback Arc Set Reduces To

- Now we take a look at problems that can be reduced from feedback arc set.
- We have already seen such a reduction, namely of the reduction of feedback arc set to the closely related problem feedback vertex set.

Result

$$\text{FEEDBACK ARC SET} \leq_p \text{FEEDBACK VERTEX SET}$$

- However, we could not find any other problem that feedback arc set naturally reduces to. This is because while doing reductions, FAS is not usually people's weapon of choice.

Problems that Feedback Arc Set Reduces To

- Now we take a look at problems that can be reduced from feedback arc set.
- We have already seen such a reduction, namely of the reduction of feedback arc set to the closely related problem feedback vertex set.

Result

FEEDBACK ARC SET $\leq_p$ FEEDBACK VERTEX SET

- However, we could not find any other problem that feedback arc set naturally reduces to. This is because while doing reductions, FAS is not usually people's weapon of choice.

Problems that Feedback Arc Set Reduces To

- Now we take a look at problems that can be reduced from feedback arc set.
- We have already seen such a reduction, namely of the reduction of feedback arc set to the closely related problem feedback vertex set.

Result

$$\textsc{Feedback Arc Set} \leq_p \textsc{Feedback Vertex Set}$$

- However, we could not find any other problem that feedback arc set naturally reduces to. This is because while doing reductions, FAS is not usually people's weapon of choice.

Problems that Feedback Arc Set Reduces To

- Now we take a look at problems that can be reduced from feedback arc set.
- We have already seen such a reduction, namely of the reduction of feedback arc set to the closely related problem feedback vertex set.

Result

$$\textsc{Feedback Arc Set} \leq_p \textsc{Feedback Vertex Set}$$

- However, we could not find any other problem that feedback arc set naturally reduces to. This is because while doing reductions, FAS is not usually people's weapon of choice.

References I

Richard M Karp.
Reducibility among combinatorial problems.
In *Complexity of computer computations*, pages 85–103. Springer, 1972.

Fănică Gavril.
Algorithms on clique separable graphs.
Discrete Mathematics, 19(2):159–165, 1977.

Cláudio Leonardo Lucchesi.
A minimax equality for directed graphs.
1977.