Complexity of paths, trails and circuits in arc-colored digraphs

Laurent Gourvès[1], Adria Lyra[2], Carlos Martinhon[2], Jérôme Monnot[1]

1. CNRS - LAMSADE Université Paris Dauphine, France
2. Fluminense Federal University, Brazil

TAMC 2010 - Prague - June 2010

Motivation

Many problems in edge-colored graphs have applications in

- molecular biology
- transportation
- social science

Those problems consist of finding a

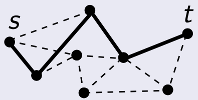
- Eulerian path
- Hamiltoinian path/cycle
- $s - t$ path (where s and t are two given nodes)

whose set of edges must

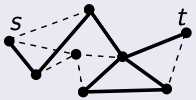
- be monochromatic, or alternate colors, or follow a given pattern
- use a minimum/maximum number of colors
- minimize transitions of colors

Trails

path



trail



A trail allow the repetition of vertices but not the repetition of edges

path $\Rightarrow$ trail but trail $\not\Rightarrow$ path

Some previous work in undirected graphs

PEC = properly edge colored = no two consecutive edges share the same color

Polynomial problems

- Deciding whether an edge-colored graph contains a PEC $s - t$ path [Szeider 00]
- Characterization of edge-colored graphs containing a PEC cycle [Yeo 97], closed trail [Abouelouaoualim et al 08]

NP-complete problems

- Deciding whether an 2-edge-colored graph contains a PEC Hamiltonian cycle, PEC Hamiltonian $s - t$ path [Bang-Jensen & Gutin 06]

The first part of this talk is about...

Typical instance

- a set of c colors $\{1, \cdots, c\}$
- a digraph D^c whose arcs have a color $\{1, \cdots, c\}$

properly arc colored or PAC

solutions in which two consecutive arcs have different colors

The first part of this talk is about...

Typical instance

- a set of c colors $\{1, \cdots, c\}$
- a digraph D^c whose arcs have a color $\{1, \cdots, c\}$

properly arc colored or PAC

solutions in which two consecutive arcs have different colors

Is it polynomial or **NP**-complete to decide whether an arc-colored digraph contains PAC trails, paths and circuits ?

Trail

Input :

- a c-colored digraph D^c
- s and t are two given nodes of $V(D^c)$

Theorem

One can decide in polynomial time whether D^c contains a PAC $s - t$ trail

Min cost flow in an appropriate auxiliary graph

Path

Input :

- a c-colored digraph D^c
- s and t are two given nodes of $V(D^c)$

Theorem

One can decide in polynomial time whether D^c contains a PAC $s - t$ path <u>if there is no cycle</u>

Use Depth First Search

Theorem

If D^c has no PAC circuit, deciding the existence of a PAC $s - t$ path is NP-complete

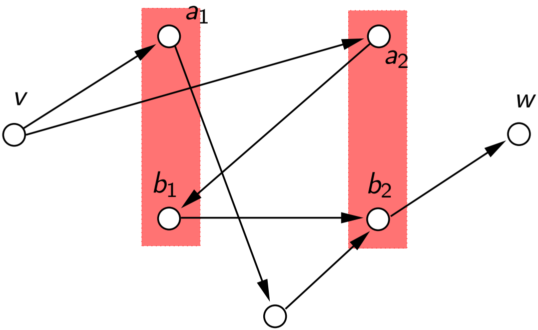
Reduce the *Path with Forbidden Pairs Problem* (PFPP for short)
Input :

- a directed graph D
- a pair of distinct nodes $v, w \in V(D)$
- q "forbidden" pairs of vertices $\{(a_1, b_1), (a_2, b_2), \cdots, (a_q, b_q)\}$, with $(a_i \neq b_i)$

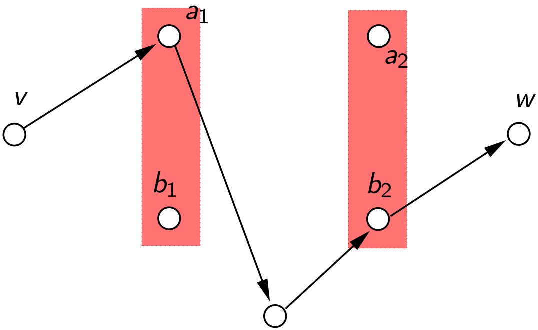
Question : Is there a directed path from v to w and passing through **at most one** vertex of each pair ?

PFPP is **NP**-complete even if D is acyclic and $\{a_i, b_i\} \cap \{a_j, b_j\} = \emptyset$ (disjoint pairs) [Gabow et al 76]

Example of PFPP



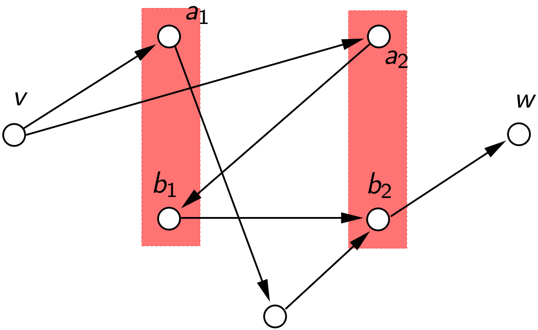
Example of PFPP



Reduction

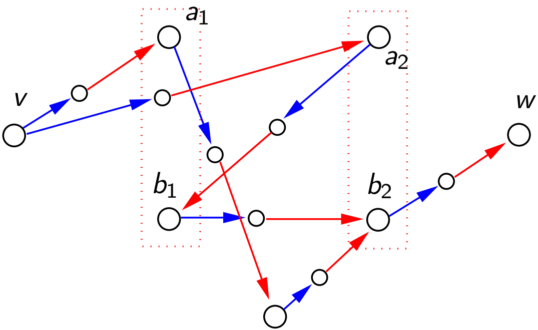
Start from an **acyclic** instance of PFPP and transform it into a 2-arc colored digraph without PAC circuit

wlog. the indegree of v and the outdegree of w are equal to 0



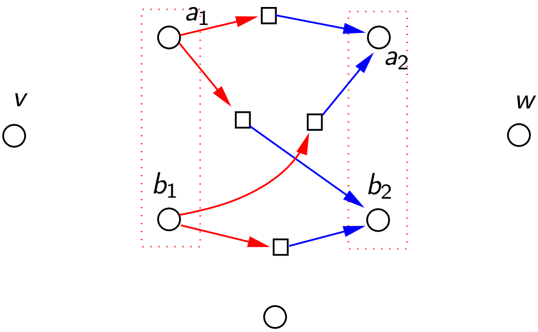
For every arc (x, y) do

- create a new node z_{xy} (see the small circles)
- add a blue arc (x, z_{xy}) and a red arc (z_{xy}, y)
- remove (x, y)

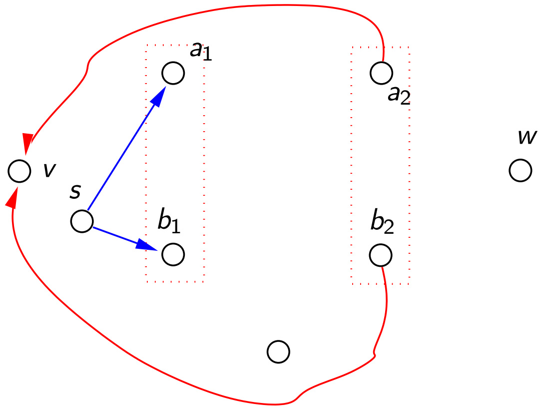


For $i = 1..k-1$ and the nodes a_i, b_i, a_{i+1}, b_{i+1} do

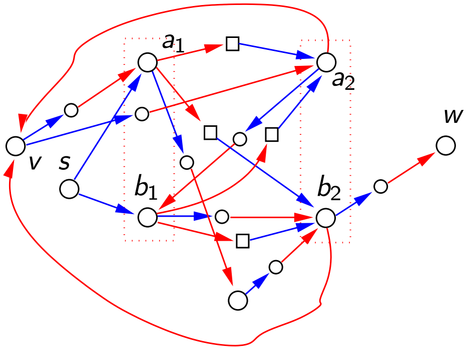
- create 4 new nodes z_i^1, z_i^2, z_i^3 and z_i^4 (see the small squares)
- add a red arc (a_i, z_i^1) and a blue arc (z_i^1, a_{i+1})
- add a red arc (a_i, z_i^2) and a blue arc (z_i^2, b_{i+1})
- add a red arc (b_i, z_i^3) and a blue arc (z_i^3, a_{i+1})
- add a red arc (b_i, z_i^4) and a blue arc (z_i^4, b_{i+1})



- create a node s
- add two blue arcs (s, a_1) and (s, b_1)
- add two red arcs (a_q, v) and (b_q, v)



The transformed graph is as follows

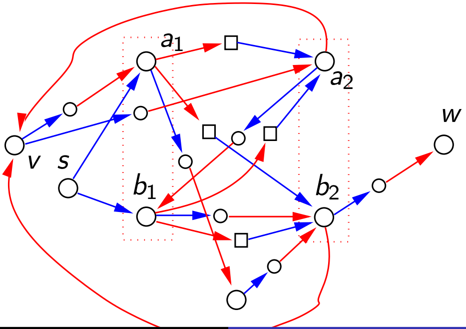


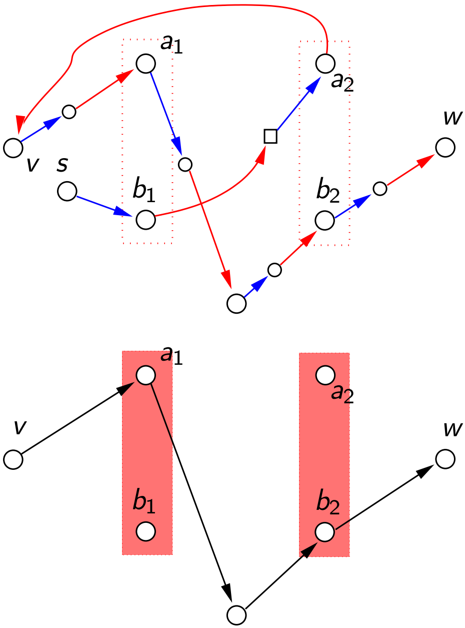
Claims

- no PAC cycle
- There is a PAC $s - w$ path **iff** the original graph admits a $v - w$ path which passes through at most one node per forbidden pair

a PAC $s - w$ path is made of two parts :

1. from s to v, passing through (exactly) one vertex per forbidden pair and using the small squares
2. from v to w, passing through **at most one** vertex per forbidden pair and using the small circles

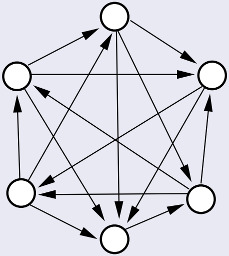



a_1
a_2
v
s
w
b_1
b_2
a_1
a_2
v
w
b_1
b_2

Tournaments

Definition

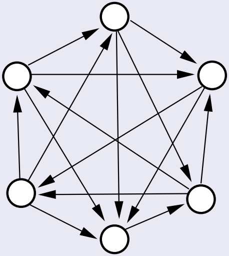
Take a complete undirected graph and assign a direction to each edge



Tournaments

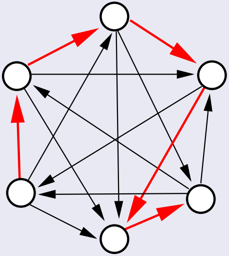
Definition

Take a complete undirected graph and assign a direction to each edge



Rédei's Theorem

Every tournament has a Hamiltonian path



the starting and ending points are not fixed

Previous works

[Bang-Jensen et al 92]

Given s and t, one can decide in polynomial time whether a tournament contains a Hamiltonian $s - t$ path

[Bang-Jensen & Gutin 06] [Feng et al 06]

Given s and t, one can decide in polynomial time whether a complete edge-colored graph contains a PEC Hamiltonian $s - t$ path

Previous works

[Bang-Jensen et al 92]

Given s and t, one can decide in polynomial time whether a tournament contains a Hamiltonian $s - t$ path

[Bang-Jensen & Gutin 06] [Feng et al 06]

Given s and t, one can decide in polynomial time whether a complete edge-colored graph contains a PEC Hamiltonian $s - t$ path

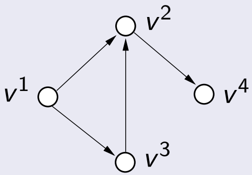
Question

What about PAC Hamiltonian $s - t$ paths in arc-colored tournaments ?

Theorem

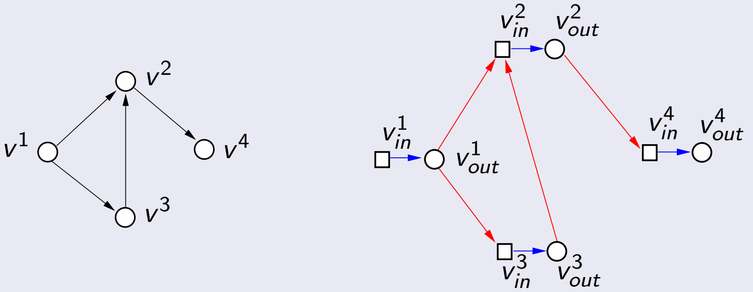
Given s and t, deciding whether a 2-arc colored tournament contains a PAC $s - t$ path is **NP**-complete

Reduce the Hamiltonian $s - t$ path problem in digraphs (NP-complete)



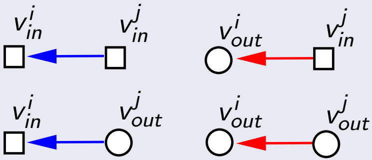
s must be v^1 and t must be v^n ($n = |V|$)

- Replace every vertex v^i by v^i_{in} (square) and v^i_{out} (circle)
- put a blue arc between v^i_{in} and v^i_{out} for all i
- put a red arc between v^i_{out} and v^j_{in} for all arc (i, j) existing in the original graph



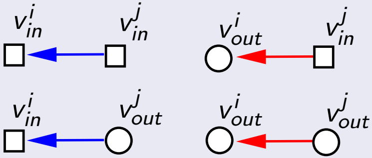
These red and blue arcs are called "original arcs" in the following

Assume $i < j$ and for every missing arc, use the following rule to complete the digraph and get a tournament

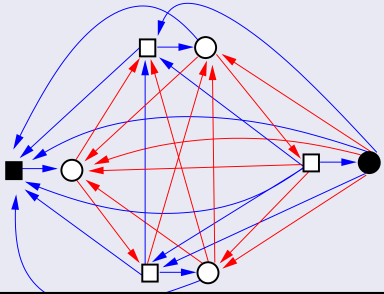


These arcs are called the "missing arcs" in the following

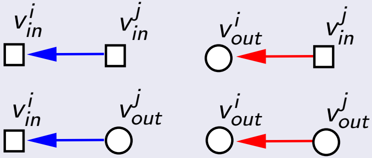
Assume $i < j$ and for every missing arc, use the following rule to complete the digraph and get a tournament



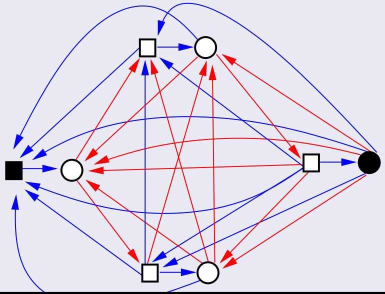
These arcs are called the "missing arcs" in the following



Assume $i < j$ and for every missing arc, use the following rule to complete the digraph and get a tournament



These arcs are called the "missing arcs" in the following



Claim

No PAC path from v^1_{in} to v^n_{out} can use a missing arc

backward argument

Final remarks

First part

The proof of NP-completeness can be extended to planar graphs with $\Omega(|V|^2)$ colors

Second part

The proof of NP-completeness of the first part can be used to show that deciding whether a c-arc-colored tournament contains a PAC circuit visiting a given vertex is NP-complete

weak version of an open problem left by Gutin, Sudako & Yeo '98 (no given vertex)