Locally-injective homomorphisms to tournaments

Thomas Bellitto

Thursday, January 12, 2017

LaBRI, University of Bordeaux

Joint work with
Stefan Bard, Gary MacGillivray and Feiran Yang, University of Victoria
Christopher Duffy, Dalhousie University

1. Graph homomorphisms
   - Undirected case
   - Oriented case
2. Local injectivity
   - Definitions
   - Example
3. Results
   - Ios-injective homomorphisms
   - Iot-injective homomorphisms
   - Irreflexive homomorphisms

Homomorphism

Graph homomorphism

A *graph homomorphism* from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$ or G_2-colouring of G_1 is a function $f : V_1 \to V_2$ such that

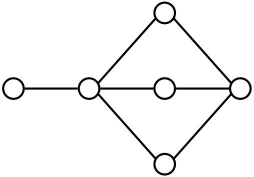
$$\forall (u, v) \in E_1, (f(u), f(v)) \in E_2$$

Colouring

Graph colouring

A n-colouring, or K_n-colouring of a graph $G = (V, E)$ is a function :

$$c : V \mapsto [\![1; n]\!] \quad \text{such that } \forall (u, v) \in E, c(u) \neq c(v)$$

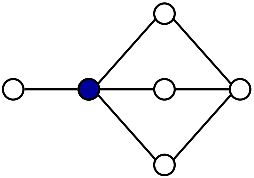


Colouring

Graph colouring

A n-colouring, or K_n-colouring of a graph $G = (V, E)$ is a function :

$$c : V \mapsto [\![1; n]\!] \quad \text{such that } \forall (u, v) \in E, c(u) \neq c(v)$$

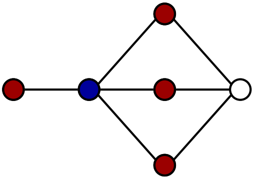


Colouring

Graph colouring

A n-colouring, or K_n-colouring of a graph $G = (V, E)$ is a function :

$$c : V \mapsto [\![1; n]\!] \quad \text{such that } \forall (u, v) \in E, c(u) \neq c(v)$$

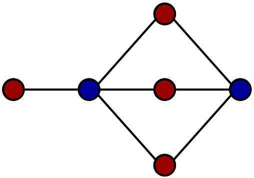


Colouring

Graph colouring

A n-colouring, or K_n-colouring of a graph $G = (V, E)$ is a function :

$$c : V \mapsto [\![1; n]\!] \quad \text{such that } \forall (u, v) \in E, c(u) \neq c(v)$$

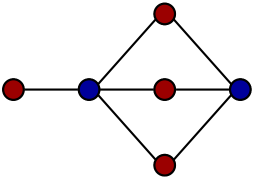


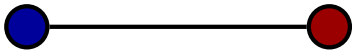
Colouring

Graph colouring

A n-colouring, or K_n-colouring of a graph $G = (V, E)$ is a function :

$$c : V \mapsto [\![1; n]\!] \quad \text{such that } \forall (u, v) \in E, c(u) \neq c(v)$$





Complexity

Theorem (Hell, Nesetril, 1990)

For any given graphs G and H, the problem of determining if there exists a homomorphism from G to H :

- *is NP-complete if H contains no loop and is not bipartite.*
- *is polynomial otherwise.*

Definitions

Oriented graph homomorphism

A *graph homomorphism* from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$, or a G_2-colouring of G_1, is a function $f : V_1 \rightarrow V_2$ such that

$$\forall (u, v) \in E_1, (f(u), f(v)) \in E_2$$

Definitions

Oriented graph homomorphism

A *graph homomorphism* from $G_1 = (V_1, E_1)$ to $G_2 = (V_2, E_2)$, or a G_2-colouring of G_1, is a function $f : V_1 \rightarrow V_2$ such that

$$\forall (u, v) \in E_1, (f(u), f(v))) \in E_2$$

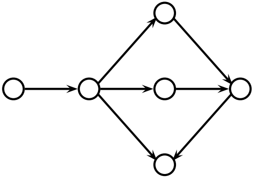
Oriented graph colouring

A *n*-colouring, of a graph $G = (V, E)$ is a function :

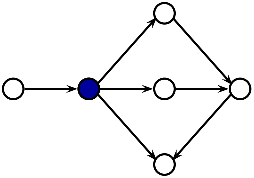
$$c : V \mapsto [\![1; n]\!] \quad \text{such that}$$

$$\begin{cases} \forall (u, v) \in E, c(u) \neq c(v) \\ \forall (u, v) \in E, \forall (x, y) \in E, c(u) \neq c(y) \text{ or } c(v) \neq c(x) \end{cases}$$

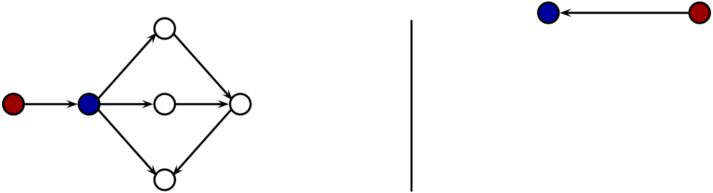
Example



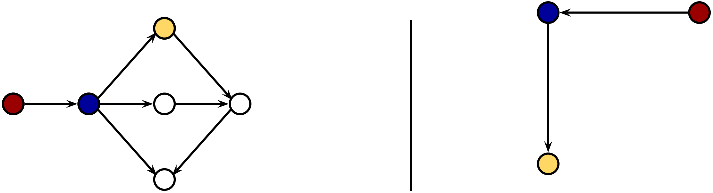
Example



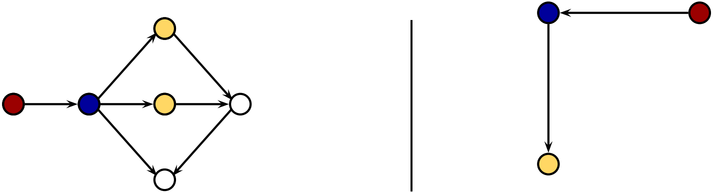
Example



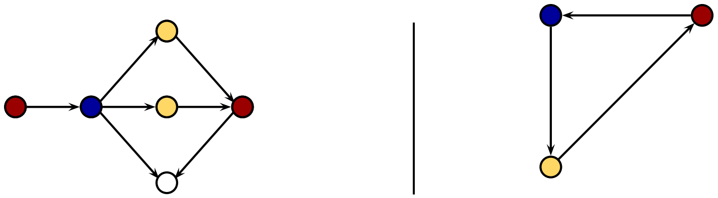
Example



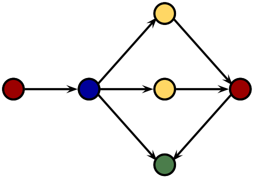
Example

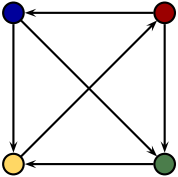


Example

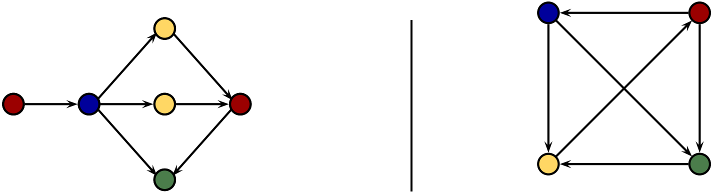


Example





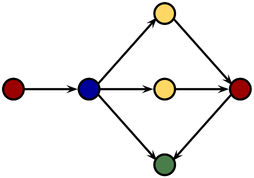
Example

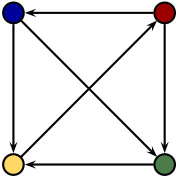


Complexity

No dichotomy result known.

Example





Complexity

No dichotomy result known.

Last minute news

Dichotomy result proposed based on whether H possesses a weak near unanimity function (Marotti and McKenzie, Rafiey, Kinne and Feder).

Several definitions

We are now looking for homomorphisms that are injective on the neighbourhoods of the vertices.
Loops on the target don't make the problem trivial.

Local injectivity

An oriented graph homomorphism f from a graph $G(V, E)$ to a graph G' is locally-injective iff

0. not studied here : $\forall v \in V$, f is injective on $N^-(v)$. (studied by MacGillivray and Swarts (2010))
1. $\forall v \in V$, f is injective on $N^-(v)$ and on $N^+(v)$.
2. $\forall v \in V$, f is injective on $N^-(v) \cup N^+(v)$.
3. $\forall v \in V$, f is injective on $N^-[v] \cup N^+[v] = N^-(v) \cup N^+(v) \cup \{v\}$.

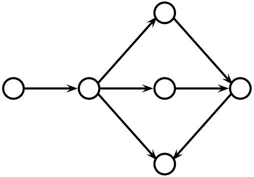
Three problems

1. $\forall v \in V$, f is injective on $N^-(v)$ and on $N^+(v)$.
2. $\forall v \in V$, f is injective on $N^-(v) \cup N^+(v)$.
3. $\forall v \in V$, f is injective on $N^-[v] \cup N^+[v] = N^-(v) \cup N^+(v) \cup \{v\}$.

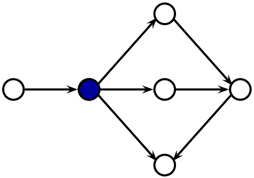
Colouring problems

- first definition, reflexive case : ios.
- second definition, reflexive case : iot.
- irreflexive case.

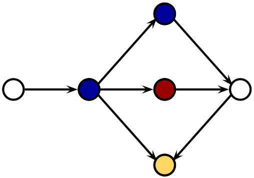
los-injective colouring

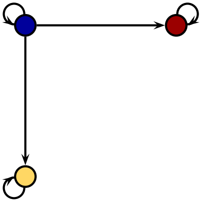


los-injective colouring

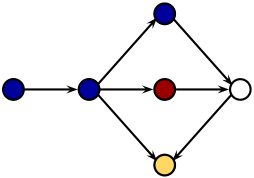


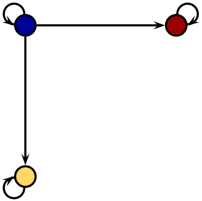
los-injective colouring



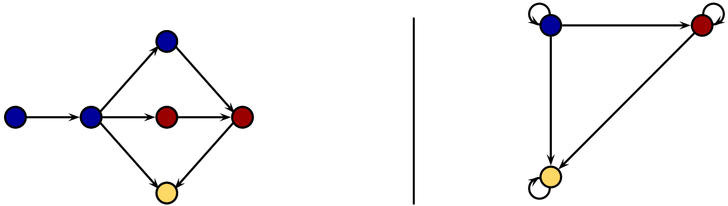


los-injective colouring

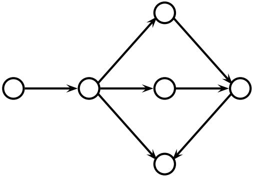




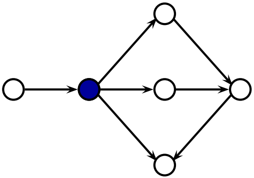
los-injective colouring



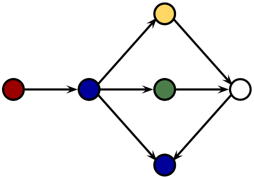
lot-injective colouring

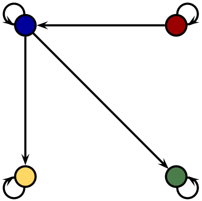


lot-injective colouring

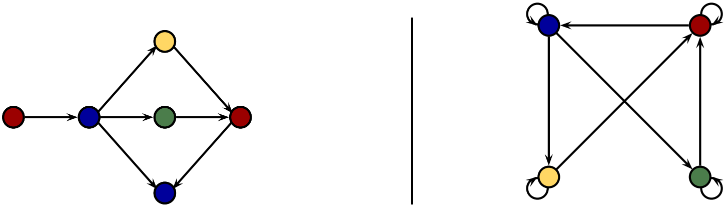


lot-injective colouring

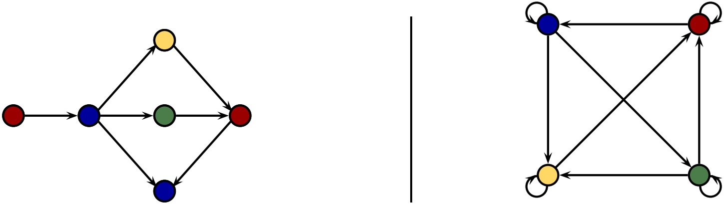




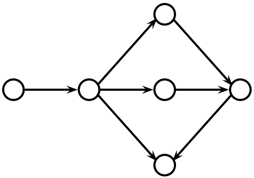
lot-injective colouring



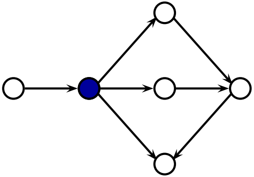
lot-injective colouring



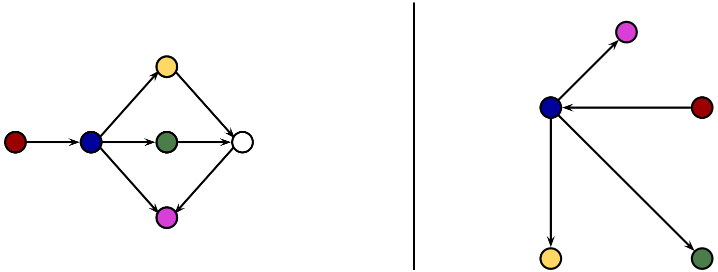
Irreflexive-colouring



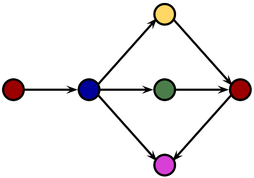
Irreflexive-colouring

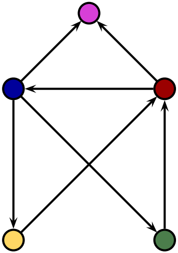


Irreflexive-colouring

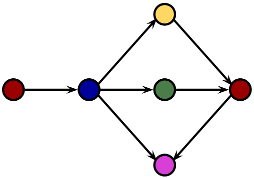


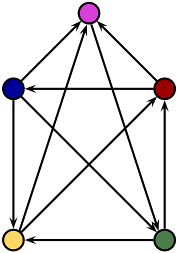
Irreflexive-colouring





Irreflexive-colouring

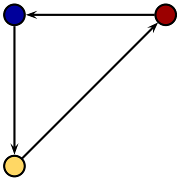


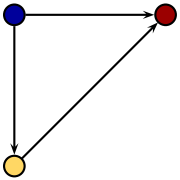


1. Graph homomorphisms
2. Local injectivity
3. Results
   - Ios-injective homomorphisms
   - Iot-injective homomorphisms
   - Irreflexive homomorphisms

Previous results (Campbell, Clarke, MacGillivray)

- Polynomial for the tournaments on two vertices or less
- NP-complete for





General reduction

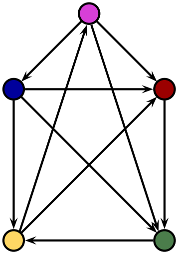
Reduction theorem

For all tournaments T on at least four vertices, for all vertices v of T, ios-injective homomorphisms to the tournament induced by the strict out- (or in -) neighbourhood of v can be reduced to ios-injective homomorphisms to T.

General reduction

Reduction theorem

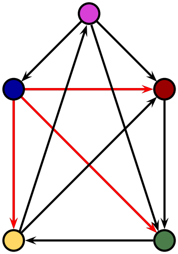
For all tournaments T on at least four vertices, for all vertices v of T, ios-injective homomorphisms to the tournament induced by the strict out- (or in -) neighbourhood of v can be reduced to ios-injective homomorphisms to T.



General reduction

Reduction theorem

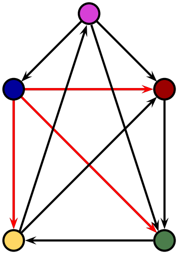
For all tournaments T on at least four vertices, for all vertices v of T, ios-injective homomorphisms to the tournament induced by the strict out- (or in -) neighbourhood of v can be reduced to ios-injective homomorphisms to T.

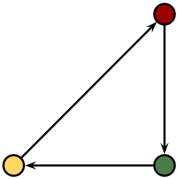


General reduction

Reduction theorem

For all tournaments T on at least four vertices, for all vertices v of T, ios-injective homomorphisms to the tournament induced by the strict out- (or in -) neighbourhood of v can be reduced to ios-injective homomorphisms to T.





General reduction

Corollary

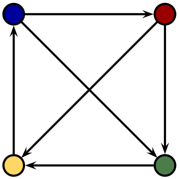
If T is a tournament on n vertices with a vertex v of strict out- or in-degree three or more, then there exists a tournament T' on at least three vertices, that has strictly fewer vertices than T and such that ios-injective homomorphism to T' can be polynomially reduced to ios-injective homomorphism to T.

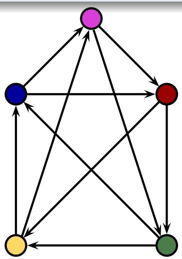
General reduction

Corollary

If T is a tournament on n vertices with a vertex v of strict out- or in-degree three or more, then there exists a tournament T' on at least three vertices, that has strictly fewer vertices than T and such that ios-injective homomorphism to T' can be polynomially reduced to ios-injective homomorphism to T.

NP-complete for





Final results

Complexity of ios-injective homomorphisms to tournaments

For any given graph G and tournament T, the problem of determining if there exists an ios-injective homomorphism from G to T :

- is NP-complete if T has three vertices or more.
- is polynomial if T has two vertices or less.

Final results

Different proofs but same results :

Complexity of iot-injective homomorphisms to tournaments

For any given graph G and tournament T, the problem of determining if there exists an iot-injective homomorphism from G to T :

- is NP-complete if T has three or more vertices.
- is polynomial if T has two vertices or less.

Polynomial algorithms

The problem is polynomial for :

- tournaments on one or two vertices.

Polynomial algorithms

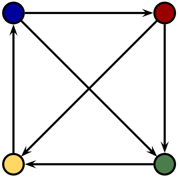
The problem is polynomial for :

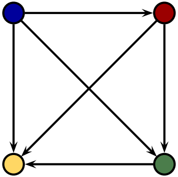
- tournaments on one or two vertices.
- tournaments on three vertices.

Polynomial algorithms

The problem is polynomial for :

- tournaments on one or two vertices.
- tournaments on three vertices.
- some tournaments on four vertices :

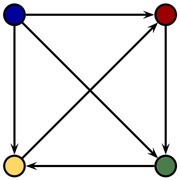


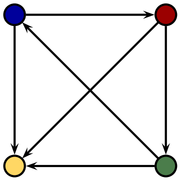


Proof on NP-completeness

NP-complete for :

- the two other tournaments on four vertices :





Proof on NP-completeness

NP-complete for :

- the two other tournaments on four vertices :

- at least 10 (out of 12) tournaments on five vertices.
- the problem has been proven polynomial on no tournament on five vertices or more so far.

Thank you !