Connexions ! Le jeu du plus grand sous-graphe connexe

Julien Bensmail, Foivos Fioravantes, Fionn Mc Inerney, Nicolas Nisse

Connexions ! Le jeu du plus grand sous-graphe connexe†

J. Bensmail[1] et F. Fioravantes[1] et F. Mc Inerney[2] et N. Nisse[1]

[1] *Université Côte d'Azur, Inria, CNRS, I3S, France*
[2] *CISPA Helmholtz Center for Information Security, Saarbrücken, Germany*

Nous définissons le jeu du plus grand sous-graphe connexe. Soit un graphe dont les sommets sont initialement non colorés. Tour-à-tour, le premier joueur, Alice, colore en rouge un sommet non coloré, puis le second joueur, Bob, colore un sommet non coloré en bleu, et ainsi de suite. Le jeu s'achève lorsque tous les sommets du graphe ont été colorés. Le vainqueur est le joueur dont le sous-graphe coloré a la plus grande composante connexe.

Nous prouvons que, si Alice joue optimalement, Bob ne peut jamais gagner, et définissons une classe de graphes infinie, appelés *graphes miroirs*, dans lesquels Bob peut forcer une égalité. Du point de vue complexité, nous montrons ensuite que déterminer l'issue du jeu est PSPACE-complet même lorsque restreint aux graphes bipartis de petit diamètre, et que reconnaître un graphe miroir est GI-difficile. Enfin, nous caractérisons les chemins et cycles dans lesquels Alice gagne et nous prouvons que l'issue du jeu peut être déterminée en temps linéaire dans la classe des cographes.

Mots-clefs : Jeux à deux joueurs dans les graphes, Jeux de connexion, Jeux à score.

1 The Largest Connected Subgraph Game

The two co-chairs of AlgoTel have organised two concurrent social events and they have to decide which of the participants will attend their event. Karine and Quentin take turns choosing a participant to attend their event. Wanting to go down in the history books as the best co-chair of AlgoTel, they are competing to see whose event will satisfy more participants. In this paper, we study this problem through the following game. The participants are represented as the vertices of a social graph, and each edge represents a friendship between the two participants it connects. Since interactions are more likely to take place between friends and indirect friends (*e.g.*, friends of friends), in the game, Karine and Quentin aim for their respective largest connected component (corresponding to the people attending their event) to be bigger than the other's.

Games in which players strive to create connected structures are known as *connection games*. Several of these games are well-known, such as the game of *Hex*, introduced independently by Hein and Nash in the 1940s. This game is played by two players on a rhombus-shaped board tiled by hexagons, with two of the opposing sides of the board coloured red and the other two coloured blue. In each round, the first player colours an uncoloured hexagonal tile red, and then, the second player colours an uncoloured tile blue. The first (second, resp.) player wins if he manages to connect the red (blue, resp.) sides of the board with red (blue, resp.) tiles. Another well-known connection game is the *Shannon switching game*, invented by Shannon in the 1950s. In this game, the first player has the goal of connecting two distinguished vertices in a graph, while the second player wants to make sure this never happens. Traditionally, the players take turns selecting edges of the graph, with the first player winning if there is a path consisting of only his edges between the two distinguished vertices, but a variant where the players select vertices (and obtain all their incident edges) also exists. However, not all connection games involve connecting sides of a board or two vertices in a graph. Generally, connection games tend to be very difficult complexity-wise (a main reason they are played and studied), with the majority of them being PSPACE-complete. For example, the general

†This work has been supported by the European Research Council (ERC) consolidator grant No. 725978 SYSTEMATICGRAPH, the STIC-AmSud project GALOP, the PHC Xu Guangqi project DESPROGES, and the UCA$^{\text{JEDI}}$ Investments in the Future project managed by the National Research Agency (ANR-15-IDEX-01). A full version of the paper can be found at https://hal.inria.fr/hal-03137305.

neighbourised Hex and the Shannon switching game on vertices, are both PSPACE-complete [Rei81, ET76]. That being said, the Shannon switching game on edges is polynomial-time solvable [BW70].

Games in which the player with the largest *score* wins, are called *scoring games* (*e.g.*, [MW11]). The score in these games is an abstract quantity usually measured in an abstract unit called *points*. Players gain points depending on the rules of the game. For example, in the orthogonal colouring game [AHMN19], a player's score is equal to the number of coloured vertices in their copy of the graph at the end of the game. Recently, some papers have started to build a general theory around scoring games (see, *e.g.*, [LNNS16]).

In this paper, we introduce and study a scoring version of a connection game that we call the *largest connected subgraph game* (LCSG). The game is played on an undirected graph G by two players, Alice and Bob. Initially, none of the vertices are coloured. Then, in each round, Alice first colours an uncoloured vertex of G red, and then, Bob colours an uncoloured vertex of G blue. Note that each vertex can only be coloured once, and its colour cannot be modified. The game ends when all of the vertices of G have been coloured. If there is a connected red subgraph such that its order (number of vertices) is strictly greater than the order of any connected blue subgraph, then Alice wins. If there is a connected blue subgraph such that its order is strictly greater than the order of any connected red subgraph, then Bob wins. Otherwise, the game is a draw. The graph G is said to be *A-win* (*B-win*, resp.) if there exists a winning strategy for Alice (for Bob, resp.), *i.e.*, a strategy that makes Alice (Bob, resp.) win regardless of the strategy of Bob (Alice, resp.). Otherwise, the graph is *AB-draw*. Given a graph G, the goal of the LCSG Problem is to decide the outcome of the game on G and to design a winning strategy (if any) or a drawing strategy for the players.

We begin by providing some general results for the LCSG Problem in Section 2, *i.e.*, that Bob can never win, that the game is a draw in a large class of graphs that we call *reflection graphs*, and that, unfortunately, recognising reflection graphs is GI-hard (*i.e.*, as hard as the Graph Isomorphism problem). We also prove, in that section, that deciding the outcome of the game is PSPACE-complete in general. Then, in Section 3, we study the game in particular graph classes, providing the resolution of the game for paths and cycles, as well as a linear-time algorithm for solving the game in cographs. We conclude with some open questions.

2 Possible outcomes and computational complexity

We first show that any graph is either A-win or AB-draw and give examples of simple infinite families of graphs for both outcomes. Then, we discuss the complexity of deciding this outcome.

By a classical strategy-stealing argument, it is easy to prove the following theorem. Roughly speaking, if the second player had a winning strategy, it would be sufficient for the first player to act as the second player, *i.e.*, to steal the winning strategy of the second player. In short, Bob can never win.

Theorem 2.1. *There does not exist a graph that is B-win, i.e., every graph is either A-win or AB-draw.*

It is easy to see that there are an infinite number of A-win graphs as any star is A-win (consider any strategy where Alice starts by colouring the center of the star). There are also an infinite number of AB-draw graphs, since any graph of even order with two universal vertices is AB-draw (consider any strategy where Bob, during his first turn, colours an uncoloured universal vertex). We can actually define a much richer class of graphs that are AB-draw. A *reflection graph* is any graph whose vertices can be partitioned into two sets $U = \{u_1, \dots, u_n\}$ and $V = \{v_1, \dots, v_n\}$ such that :

1. the subgraph induced by the vertices of U is isomorphic to the subgraph induced by the vertices of V, and the bijection $U \to V$, where v_i is the image of u_i for all $1 \leq i \leq n$, is an isomorphism ;
2. for any two vertices $u_i \in U$ and $v_j \in V$, if the edge $u_i v_j$ exists, then the edge $u_j v_i$ also exists.

Lemma 2.2. *Any reflection graph G is AB-draw.*

Indeed, if G is a reflection graph, then a drawing strategy for Bob is as follows : whenever Alice colours u_i (v_i, resp.), Bob colours v_i (u_i, resp.). Note that any path, cycle, or Cartesian product of two graphs, is a reflection graph if its order is even. However, recognising reflection graphs is, unfortunately, not an easy problem. In particular, we prove that it is GI-hard, essentially meaning that it is unlikely to be polynomial-time solvable unless the same holds for the Graph Isomorphism problem as well.

Finally, via a (non-trivial) reduction from the classical PSPACE-complete problem POS CNF, we prove :

Theorem 2.3. *The LCSG Problem is PSPACE-complete in bipartite graphs with diameter at most* 5.

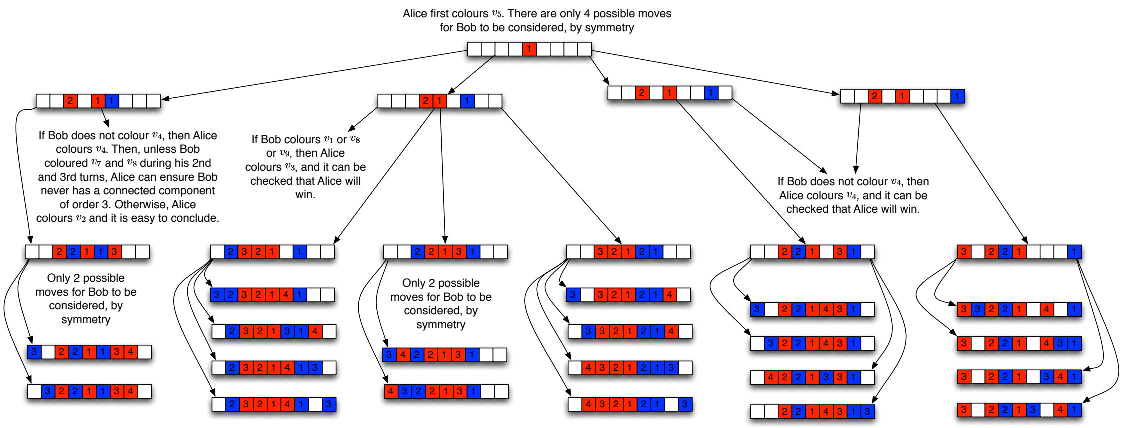

FIGURE 1: Winning strategy for Alice in P_9. The squares represent the vertices v_1 to v_9 from left to right. A number i in a red (blue, resp.) square indicates this vertex is the i^{th} vertex coloured by Alice (Bob, resp.). Each arrow corresponds to a move of Bob and then one of Alice. The last moves in each case are omitted as it is easy to check the last possibilities.

3 Some polynomial cases : paths, cycles, and cographs

An interesting aspect of the LCSG is that, depending on the graph's structure, Alice and Bob might have to consider different kinds of strategies. We illustrate this by paths and cycles, where the players must restrict their opponent's components, and cographs, where they must grow a unique monochromatic component.

Theorem 3.1. *Let $n \in \mathbb{N}$. The path P_n is A-win if and only if $n \in \{1,3,5,7,9\}$.*

Sketch of proof. Let $P_n = (v_1,\dots,v_n)$. By Lemma 2.2, P_n is *AB*-draw if n is even since any even-order path is a reflection graph. It is easy to see that, if $n \in \{1,3,5,7\}$, then Alice wins by colouring the center of P_n on the first turn. The winning strategy for Alice in the case $n = 9$ is a bit more involved and is described in Figure 1. Assume now that n is odd and $n \geq 11$; we show that P_n is *AB*-draw.

We first show that, in any path P_n, the following strategy (not necessarily ending in a draw) for Bob ensures that the largest connected red subgraph is of order at most 2, even if Alice plays first and v_1 is already coloured red. Indeed, whenever Alice colours a vertex v_j (for $1 \leq j \leq n$), Bob colours the vertex v_{j-1} if possible. If v_{j-1} is already coloured or does not exist, then Bob colours v_k, where $k > j$ is the smallest integer such that v_k is not coloured yet.

Now, let v_j, $1 \leq j \leq n$, be the first vertex Alice colours. Since $n \geq 11$, there are at least 5 vertices to the left or right of v_j, say, w.l.o.g., to the left of v_j (*i.e.*, $5 < j \leq n$). Bob colours v_{j-1}. Let $Q = (v_1,\dots,v_{j-1})$ and $Q' = (v_j,\dots,v_n)$. Now, when Alice plays in Q (in Q', resp.), Bob then plays in Q (in Q', resp.), and both games are considered independently (since v_{j-1} is blue and v_j is red). Considering Q' as a path with one of its ends initially coloured red, and applying the arguments of the previous paragraph, Bob can ensure that Alice cannot create a connected red component of order more than 2 in Q'. On the other hand, we prove that Bob can ensure a draw with a blue component of order at least 2 in Q since Q has at least 5 vertices. □

Theorem 3.2. *Let $n \in \mathbb{N}$. The cycle C_n is A-win if and only if n is odd.*

Sketch of proof. If n is even, then C_n is a reflection graph, and so, it is *AB*-draw by Lemma 2.2. Also, if $n \leq 5$, the result is obvious, so let us assume that $n > 5$ and odd. We describe a winning strategy for Alice.

First, let us assume (independently of how this configuration eventually appears) that after $x \geq 3$ turns of each of Alice and Bob, vertices $v_1,\dots,v_x$ are coloured red, vertices v_n and v_{x+1} are coloured blue, and any $x-2$ other vertices in $\{v_{x+2},\dots,v_{n-1}\}$ are coloured blue. Note that it is now Alice's turn. It can be shown through case analysis and induction on n, that Alice can ensure that any connected blue component is of order less than x in the subgraph induced by $(v_{x+1},\dots,v_n)$. Therefore, in that situation, Alice wins.

Now, let Alice first colour v_1. If Bob does not colour a neighbour of v_1 (say Bob colours v_j with $3 < j < n$, since $n \geq 5$ and odd), then, on her second turn, Alice colours v_2. During the next rounds, while it is possible, Alice colours a neighbour of the connected red component. When it is not possible anymore, either the connected red component is of order $\lceil n/2 \rceil$ or it is of order at least 3 and we are in the situation of the above paragraph. In both cases, Alice wins.

Thus, after Alice colours her first vertex (say v_2), Bob must colour a neighbour of that vertex (say v_1). By induction on the number $t \geq 1$ of rounds, let us assume that the game reaches, after t rounds, a configuration where, for every $1 \leq i \leq t$, the vertices v_{2i-1} are coloured blue and the vertices v_{2i} are coloured red. If $t = \lfloor n/2 \rfloor$, then Alice finally colours v_n (recall that n is odd) and wins. Otherwise, let Alice colour v_{2t+2}.

If Bob then colours v_{2t+1}, then we are back to the previous situation for $t' = t+1$. Then, eventually, Alice wins by induction on $n - 2t$. Otherwise, if Bob does not colour v_{2t+1}, then Alice colours v_{2t+1} and then continues to grow the connected red component containing v_{2t+1} while possible. When it is not possible anymore, note that removing (or contracting) the vertices v_2 to v_{2t}, leads us back to the situation of the first paragraph of this proof (with a connected red component of order at least 3) and, therefore, Alice wins. □

Now, we address the LCSG Problem in P_4-free graphs, also known as *cographs*, which can be defined recursively as follows. The one-vertex graph K_1 is a cograph. Let G_1 and G_2 be two cographs. Then, the *disjoint union* $G_1 + G_2$ is a cograph. Moreover, the *join* $G_1 \oplus G_2$, obtained from $G_1 + G_2$ by adding all the possible edges between the vertices of G_1 and the vertices of G_2, is a cograph. Note that a decomposition of a cograph can be computed in linear time [CPS85]. We show that the outcome of the LCSG Problem can be decided in linear time in cographs. This is established mainly through an inductive procedure. However, in our induction, the outcome (A-win or AB-draw) for smaller graphs is sometimes insufficient to decide the outcome for larger graphs. To deal with this issue, we refine the outcome of the game by defining $\mathscr{A}^*$ as the set of graphs for which Alice has a strategy that ensures a single connected red component.

Theorem 3.3. *Let G be a cograph. There exists a linear-time algorithm that decides whether G is A-win or AB-draw, and whether $G \in \mathscr{A}^*$ or not.*

Sketch of proof. The proof is by induction on $n = |V(G)|$. If $n = 1$, then G is clearly A-win and $G \in \mathscr{A}^*$. Let us assume that $n > 1$, and that the outcome (being A-win or AB-draw, and belonging to $\mathscr{A}^*$) of the game can be decided in linear time for any cograph of order less than n. There are two cases to be considered. Either $G = G_1 \oplus G_2$ for some cographs G_1 and G_2, or $G = G_1 + G_2 + \ldots + G_m$, where, for every $1 \leq i \leq m$ ($m \geq 2$), G_i is either a single vertex or is a cograph obtained from the join of two other cographs. By treating these two cases separately (note that considering the disjoint union of two arbitrary cographs appears to be insufficient), it can be computed in linear time if G is A-win or AB-draw, and whether $G \in \mathscr{A}^*$ or not. □

Further work. Several directions for further work are of interest. For example, it would be interesting to study the game in other graph classes such as trees and interval graphs. Since grids of even order are reflexion graphs and so AB-draw by Lemma 2.2, grids of odd order are also interesting. Just as reflection graphs are a large class of graphs that are AB-draw, another direction would be to find a large class of graphs that are A-win. Graphs of odd order in which Alice can always construct a single connected red component are A-win, and so, perhaps a class of dense graphs of odd order would be a prime candidate.

Références

[AHMN19] S. D. Andres, M. Huggan, F. Mc Inerney, and R. J. Nowakowski. The orthogonal colouring game. *Theoretical Computer Science*, 795 :312–325, 2019.

[BW70] J. Bruno and L. Weinberg. A constructive graph-theoretic solution of the shannon switching game. *IEEE Transactions on Circuit Theory*, 17(1) :74–81, 1970.

[CPS85] D. G. Corneil, Y. Perl, and L. K. Stewart. A linear recognition algorithm for cographs. *SIAM Journal on Computing*, 14(4) :926–934, 1985.

[ET76] S. Even and R. E. Tarjan. A combinatorial problem which is complete in polynomial space. *J. ACM*, 23(4) :710–719, 1976.

[LNNS16] U. Larsson, J. P. Neto, R. J. Nowakowski, and C. P. Santos. Guaranteed scoring games. *The Electronic Journal of Combinatorics*, 23(3), 2016.

[MW11] P. Micek and B. Walczak. A graph-grabbing game. *Comb., Prob. & Comput.*, 20(4) :623–629, 2011.

[Rei81] S Reisch. Hex ist PSPACE-vollständig. *Acta Informatica*, 15(2) :167–191, 1981.