Connectivity in the presence of an opponent (extended abstract)

Zihui Liang, Bakh Khoussainov, Toru Takisaka, Mingyu Xiao

The paper introduces two player connectivity games played on finite bipartite graphs. Algorithms that solve these connectivity games can be used as subroutines for solving Müller games. Müller games constitute a well established class of games in model checking and verification. In connectivity games, the objective of one of the players is to visit every node of the game graph infinitely often. The first contribution of this paper is our proof that solving connectivity games can be reduced to the incremental strongly connected component maintenance (ISCCM) problem, an important problem in graph algorithms and data structures. The second contribution is that we non-trivially adapt two known algorithms for the ISCCM problem to provide two efficient algorithms that solve the connectivity games problem. Finally, based on the techniques developed, we recast Horn's polynomial time algorithm that solves explicitly given Müller games and provide the first correctness proof of the algorithm. Our algorithms are more efficient than that of Horn's algorithm. Our solution for connectivity games is used as a subroutine in the algorithm.

Background: explicit Müller games and connectivity games. In the area of logic, model checking, and verification of reactive systems, studying games played on graphs is a key research topic [5]. This is mostly motivated through modelling reactive systems and reductions of model checking problems to games on graphs. Understanding the algorithmic content of determinacy results is also at the core of this research. *Müller games* constitute a well-established class of games for verification. Recall that a Müller game is a tuple $\mathcal{G} = (V_0, V_1, E, \Omega)$, where

- The tuple $G = (V_0 \cup V_1, E)$, called the *arena* of $\mathcal{G}$, is a finite directed bipartite graph so that V_0 and V_1 partition the set $V = V_0 \cup V_1$.
- The set $E \subseteq (V_0 \times V_1) \cup (V_1 \times V_0)$ of edges.
- V_0 and V_1 are sets of vertices (or *nodes*) from which player 0 and player 1, respectively, move.
- $\Omega \subseteq 2^V$ is a collection of winning sets.

The players play a Müller game by moving a given token along the edges of the graph. The token is initially placed on a node $v_0 \in V$. The play proceeds in rounds. At any round of the play, if the token is placed on a player σ's node v, then player σ chooses a sccessor u of v, moves the token to u and the play continues on to the next round. Formally, a *play* (starting from v_0) is a sequence $\rho = v_0, v_1, \dots$ such that $(v_i, v_{i+1}) \in E$ for all $i \in \mathbb{N}$. If a play reaches a node v with no successor, then player 1 wins the play. For an infinite play ρ, set $\mathsf{Inf}(\rho) = \{v \in V \mid \exists^\omega i(v_i = v)\}$. We say player 0 wins the play ρ if $\mathsf{Inf}(\rho) \in \Omega$; otherwise, player 1 wins the play. By the result of Martin [11], Müller games are *determined*; that is, for a given vertex v of a Müller game, either player 0 or 1 has a winning

---

* 対戦相手の存在下における接続性
This is an unreferred paper, whose content is to appear in the proceedings of the 31st Annual European Symposium on Algorithms (ESA 2023).

Zihui Liang, Bakh Khoussainov, Toru Takisaka, Mingyu Xiao, 電子科技大学 計算機科学・工学部, School of Computer Science and Engineering, University of Electronic Science and Technology of China.

strategy from v (i.e., one of the players can enforce their win from v no matter how the opponent makes their transitions). The *solution* of a Müller game is a partition $\{Win_0, Win_1\}$ of the vertex set V, where $v \in Win_\sigma$ if and only if player σ wins the game starting at v, $\sigma \in \{0, 1\}$.

The complexity of solving Müller games depends on the presentation of the games. For many reasonable representations—win-set, Müller, Zielonka DAGs, and Emerson-Lei—the problem is known to be PSPACE hard [8]. Meanwhile, F. Horn [7] provides a polynomial time algorithm that solves *explicitly given Müller games*; here, a Müller game $\mathcal{G} = (V_0, V_1, E, \Omega)$ is explicitly given if V_0, V_1, E, and all sets in Ω are fully presented as input.

Horn's algorithm iteratively calls a subroutine for solving a particular Müller game, which we call a *connectivity game*. Here, a Müller game is called a connectivity game if $\Omega = \{V\}$, i.e., the game requires player 0 to visit every vertex infinitely many often. In a connectivity game $\mathcal{G}$, we always have either $Win_0 = V$ or $Win_0 = \emptyset$; we say player 0 wins $\mathcal{G}$ in the former case, or otherwise player 1 wins $\mathcal{G}$. Connectivity games naturally appear in other algorithms for solving Müller games; they also offer a natural two-player extension of *strongly connected components* (SCCs), which is a fundamental notion in graph theory. To date, the best known time bound for solving connectivity games has been $\mathbf{O}(|V||E|)$ [2] [3] [4] [9].

Contribution. This is an extended abstract of our resent publication [10]. Below we summarize the contribution in the paper.

Contribution 1: state-of-the-art algorithms for solving connectivity games and explicit Müller games. We provide two algorithms for solving connectivity games that run in time $\mathbf{O}((\sqrt{|V_1|} + 1)|E| + |V_1|^2)$ and $\mathbf{O}((|V_1| + |V_0|) \cdot |V_0| \log |V_0|)$, respectively. Both improve the existing time bound $\mathbf{O}(|V||E|)$. The crux of our new algorithms is a reduction of the problem into an *incremental strongly connected component maintenance* (ISCCM) problem, i.e., an online computation problem of SCCs of a graph whose edges may increase. Intuitively, for a given arena $G = (V_0 \cup V_1, E)$, we consider the following procedure:

1. Initialize a directed graph $G' = (V', E')$ with $(V_0 \cup V_1, E_0)$, where E_σ is the collection of all edges from player σ's node. Also initialize an edge set E'_1 with E_1.
2. For each $(v, u) \in E'_1$, if it is a unique edge from v in E'_1, then pop it and add it to E'.
3. Compute SCCs of G' (recall an SCC of G' is a maximal subset $\tilde{V}$ of V' such that there is a path between any distinct vertices in $\tilde{V}$). If there is no non-trivial SCC (i.e., those which has at least two vertices), then stop; otherwise, merge each SCC into a single vertex, merge each identical edges in E'_1 into one, and go to Step 2.

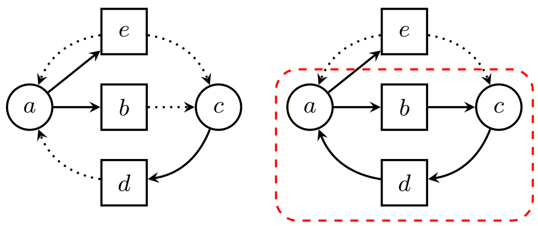


図 1　**Reduction into an ISCCM problem.**

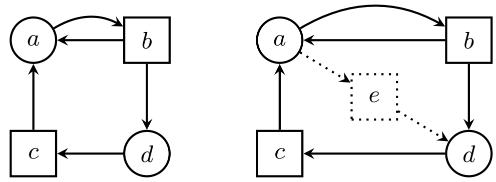
For example, suppose an arena in the left of Figure 1 is given; there, circles and boxes represent nodes of player 0 and 1, respectively. Then G' is initialized with the graph whose edges consist of the thick arrows; and E'_1 is initialized with the set of dotted arrows. The first iteration of Step 2 adds edges (b, c) and (d, a) to G', which make the vertex set $\tilde{V} = \{a, b, c, d\}$ an SCC of G' (see the right of Figure 1). The first iteration of Step 3 merges $\tilde{V}$ into a single vertex, and merges (e, a) and (e, c) into a single edge $(e, \tilde{V})$; the second iteration of Step 2 and 3 further turns G' into a singleton; and then the procedure stops after yet another iteration, with no update on G'.

We have proved that the connectivity game on G is won by player 0 if and only if the above procedure turns G' into a singleton[†1] ([10], Theorem 8). By performing this procedure via two known algorithms for the ISCCM problem [1] [6], we realize the algorithms for solving connectivity games that run in the desired time. The details can be found in Section 2 and 3 of [10]. This result also improves the time bound of Horn's algorithm, which calls an algorithm for solving connectivity games as a subroutine ([10], Theorem 20 and 21).

Contribution 2: a correctness proof of Horn's algorithm. We point out that the correctness proof of Horn's algorithm in [7] has non-trivial flaws, and provide an alternative proof based on a different idea.

The flaws in the proof are related to a certain game transformation in the algorithm. Horn's algorithm solves a Müller game by evaluating the winner of the game for each winning set $W \in \Omega$ one by one. In each evaluation round, the algorithm also possibly transforms the Müller game to a new one (this takes care of the case "player 0 can win, but they cannot choose which winning set to achieve"). For example, suppose a Müller game in the left of Figure 2 is given. Horn's algorithm first pops $W_1 = \{a, b\}$ from Ω, and solves the connectivity game over W_1. The algorithm finds that player 0 wins the connectivity game, while player 1 can exit W_1 in the Müller game (via the edge (b, d)). This fact tells us that, although player 0 cannot enforce the winning condition W_1 exclusively, they may still be able to enforce *either* W_1 or $W_2 = \{a, b, c, d\}$

†1 Strictly speaking, in [10] we consider an equivalent procedure that does not perform actual merging of vertices and edges. There, G' is turned into an SCC instead of a singleton.



$\Omega = \{\{a, b\}, \{a, b, c, d\}\}$ $\Omega = \{\{a, b, c, d, e\}\}$

図 2 Game transformation in Horn's algorithm.

(where the choice of W_1 or W_2 is up to player 1). In such a case, Horn's algorithm transforms the Müller game by: (1) adding a new node e and edges so that player 0 can force player 1 to exit W_1, and (2) swapping each superset W' of W_1 in Ω with $W' \cup \{e\}$. In the example, the original Müller game is transformed into the one in the right of Figure 2. Horn's algorithm then pops $W_2' = \{a, b, c, d, e\}$ from Ω, solves the connectivity game over W_2', and finds player 0 wins it; as player 1 cannot exit W_2' in the (transformed) Müller game, the algorithm reports player 0 wins the original Müller game at each node in $W_2' \cap V = V$.

In Horn's paper [7], the correctness proof of his algorithm is conveyed as if the following is true, which is actually not: *if $W \in \Omega$ is sensible (i.e., if the restriction of the game over W is deadlock-free), then it is still sensible after the transformation of the game.* In our view, there is no easy way to recover the proof in [7]; a discussion on this point is given in Section 6 of [10]. The key observation in our alternative proof is that the transformation in Horn's algorithm preserves the winner of the game at each node ([10], Theorem 17). The details of the correctness proof is given in Section 5 of [10].

参 考 文 献

[1] Bender, M. A., Fineman, J. T., and Gilbert, S.: A new approach to incremental topological ordering, *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, SIAM, 2009, pp. 1108–1115.

[2] Bodlaender, H. L., Dinneen, M. J., and Khoussainov, B.: On game-theoretic models of networks, *Algorithms and Computation: 12th International Symposium, ISAAC 2001 Christchurch, New Zealand, December 19–21, 2001 Proceedings 12*, Springer, 2001, pp. 550–561.

[3] Bodlaender, H. L., Dinneen, M. J., and Khoussainov, B.: Relaxed update and partition network games, *Fundamenta Informaticae*, Vol. 49, No. 4(2002), pp. 301–312.

[4] Dinneen, M. J. and Khoussainov, B.: Update networks and their routing strategies, *International Workshop on Graph-Theoretic Concepts in Computer Science*, Springer, 2000, pp. 127–136.

[5] Grädel, E., Thomas, W., and Wilke, T.: Automata, logics, and infinite Games. LNCS, vol. 2500, 2002.

[6] Haeupler, B., Kavitha, T., Mathew, R., Sen, S., and Tarjan, R. E.: Incremental cycle detection, topological ordering, and strong component maintenance, *ACM Transactions on Algorithms (TALG)*, Vol. 8, No. 1(2012), pp. 1–33.

[7] Horn, F.: Explicit Muller games are PTIME, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2008.

[8] Hunter, P. and Dawar, A.: Complexity bounds for regular games, *International Symposium on Mathematical Foundations of Computer Science*, Springer, 2005, pp. 495–506.

[9] Khaliq, I., Khoussainov, B., and Liu, J.: Extracting winning strategies in update games, *Models of Computation in Context: 7th Conference on Computability in Europe, CiE 2011, Sofia, Bulgaria, June 27-July 2, 2011. Proceedings 7*, Springer, 2011, pp. 142–151.

[10] Liang, Z., Khoussainov, B., Takisaka, T., and Xiao, M.: Connectivity in the presence of an opponent, *31st Annual European Symposium on Algorithms, ESA 2023, September 4-6, 2023, Amsterdam, The Netherlands*, To appear.

[11] Martin, D. A.: Borel determinacy, *Annals of Mathematics*, Vol. 102, No. 2(1975), pp. 363–371.