



**HAL**  
open science

## The Closed Geodetic Game: algorithms and strategies

Antoine Dailly, Harmender Gahlawat, Zin Mar Myint

### ► To cite this version:

Antoine Dailly, Harmender Gahlawat, Zin Mar Myint. The Closed Geodetic Game: algorithms and strategies. International Workshop on Combinatorial Algorithms, Jul 2025, Bozeman, United States. pp.273-286, <10.1007/978-3-031-98740-3\_20>. <hal-05177776>

**HAL Id: hal-05177776**

**<https://hal.science/hal-05177776v1>**

Submitted on 22 Jul 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

# The Closed Geodetic Game: algorithms and strategies

Antoine Dailly<sup>1,2</sup>, Harmender Gahlawat<sup>1,3</sup>, and Zin Mar Myint<sup>4</sup>

<sup>1</sup> Université Clermont Auvergne, CNRS, Mines de Saint-Étienne,  
Clermont-Auvergne-INP, LIMOS, 63000 Clermont-Ferrand, France

<sup>2</sup> Université Clermont Auvergne, INRAE, UR TSCF, 63000, Clermont-Ferrand,  
France

<sup>3</sup> Laboratoire G-SCOP (CNRS, Univ. Grenoble Alpes), Grenoble, France

<sup>4</sup> Indian Institute of Technology Dharwad, India

**Abstract.** The *geodetic closure* of a set  $S$  of vertices of a graph  $G$  is the set of all vertices in shortest paths between pairs of vertices of  $S$ . A set  $S$  of vertices in a graph is *geodetic* if its geodetic closure contains all the vertices of the graph.

Several authors have studied variants of games around constructing geodetic sets. The most studied of those, GEODETIC GAME, was introduced by Harary in 1984 and developed by Buckley and Harary in 1985. It is an *achievement* game: both players construct together a geodetic set by alternately adding vertices to the set, the winner being the one who plays last. However, this version of the game allows the players to select vertices that already are in the geodetic closure of the current set.

We study the more natural variant, called CLOSED GEODETIC GAME, where the players alternate adding to  $S$  vertices that are not in the geodetic closure of  $S$ . This variant was also introduced in another, less-noticed, paper by Buckley and Harary in 1985, and only studied since then in the context of trees by Araujo *et al.* in 2024. We provide a full characterization of the Sprague-Grundy values (equivalence values for games) of graph classes such as paths and cycles, of the outcomes of the Cartesian product of several graphs in function of their individual outcomes, and give polynomial-time algorithms to determine the Sprague-Grundy values in cactus and block graphs.

**Keywords:** Geodetic Set · Geodetic Game · Polynomial-time algorithms · Cartesian Product · Block Graphs · Cactus Graphs · Combinatorial Games

## 1 Introduction

### 1.1 A history of geodetic games

Let  $G(V, E)$  be a simple, undirected graph with vertex set  $V$  and edge set  $E$ . Given two vertices  $u, v \in V$ , let  $\mathcal{I}(u, v)$  denote the set of vertices that lie on some shortest path with  $u$  and  $v$  as endpoints. Given a subset  $S \subseteq V$  of vertices, its

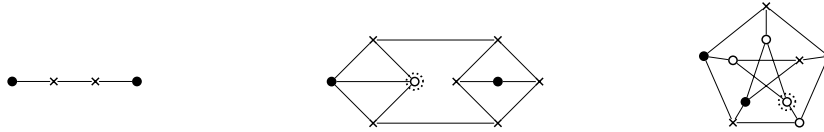


Fig. 1: Black vertices belong to the set  $S$ , crossed vertices belong to the geodetic closure  $(S)$ , white vertices are uncovered vertices yet. However, if we add the dotted vertices to  $S$ , then  $V = (S)$ , *i.e.*,  $S$  becomes a geodetic set.

*geodetic closure*, denoted by  $(S)$ , is defined as the set of vertices in all shortest paths between every pair of vertices of  $S$ . More formally,  $(S) = \bigcup_{u,v \in S} \mathcal{I}(u,v)$ .

A set  $S$  of vertices of  $G(V, E)$  is called a *geodetic set of  $G$*  if  $(S) = V$  (see Figure 1 for examples). The problem of finding a geodetic set of minimum size was defined in the 1980s [19,6,9,18] and has been well-studied and is known to be NP-hard for general graphs [18,3]. The problem has been also well-studied on graph classes and displays quite an interesting behaviour. In particular, the problem stays NP-hard even for subcubic partial grids [11], for interval graphs [10], and admits a polynomial time algorithm for solid grids [11], outerplanar graphs [21], and proper interval graphs [14].

Interestingly, the game version of the geodetic number was introduced in parallel of its combinatorial counterpart. The first mention of a game based around geodetic sets was in 1984, by Harary [17]: two players alternate adding new vertices of a graph  $G(V, E)$  to a set  $S$ , the game ends when  $(S) = V$ . Two winning conditions are proposed: in the *achievement* setting, the player who plays the last move wins; while they lose in the *avoidance* setting. Achievement and avoidance correspond to the *normal* and *misère* settings of combinatorial games, respectively. We will focus exclusively on achievement games.

With respect to combinatorial game theory, the first question to ask is, **given an input graph, decide which player wins**. This is called computing the *outcome* of the game. Note that, since the game is *impartial* (both players are indistinguishable), there are only two possible outcomes: either the first player wins (denoted by  $\mathcal{N}$ ) or the second player wins (denoted by  $\mathcal{P}$ ). Furthermore, when playing on disjoint connected components, each component is independent (selecting a vertex in a component has no influence whatsoever on the other components). If two components are  $\mathcal{P}$ , then, playing on their disjoint union is also  $\mathcal{P}$  (the second player always applies their winning strategy on the component on which the first player played a move); if one component is  $\mathcal{P}$  and the other is  $\mathcal{N}$ , then the first player wins by applying their winning strategy on the  $\mathcal{N}$  component, becoming the second player on the disjoint union of two  $\mathcal{P}$  components. However, outcomes are not enough to solve the disjoint union of two  $\mathcal{N}$  components: we need to use *Sprague-Grundy values*, which are equivalence classes for games (and can be mapped to nonnegative integers). More details will be given in Section 1.3. Hence, a stronger question is, **for a given input graph, compute its Sprague-Grundy value**.

## 1.2 GEODETIC GAME, CLOSED GEODETIC GAME and other related variants

In his seminal paper [17], Harary discussed several possible rulesets for games based around the notion of geodesics. Two of those rulesets were then expanded in two 1985 papers by Buckley and Harary, and constitute the main variants. All games are played on a graph  $G(V, E)$ .

In the first paper [8], the authors define GEODETIC GAME. In this game, two players alternately add vertices to a set  $S$ , until  $(S) = V$ . They give the outcome of the game for complete graphs, cycles, (generalized) wheels, complete bipartite graphs and  $n$ -cubes. This line of research was expanded later in [22] which improved on the wheels' result, and in [20], which studied complete multipartite graphs, hypercubes, and graphs with a unique minimum-size geodetic set (such as trees, split graphs or coronas).

In the second paper [7], they introduce CLOSED GEODETIC GAME. In this game, the players alternatively add to  $S$  vertices that are not in  $(S)$  as it is at the moment they are selecting it. They study the same graph classes as in [8]. Surprisingly, this more natural variant was left unstudied for decades, probably due to the difficulty of access of [7] compared to [8]. Recently, Araujo *et al.* [2] studied several geodetic game variants, including CLOSED GEODETIC GAME (which they called *geodesic closed interval game*), for which they provided a linear-time algorithm computing the Sprague-Grundy value of a tree.

Note that GEODETIC GAME and CLOSED GEODETIC GAME can have different outcomes on even very simple graphs. One such example is illustrated in Figure 2. Since the players are allowed to pick covered vertices in GEODETIC GAME, the parity of the graph (and of covered vertices) plays a bigger role in determining the outcome compared to CLOSED GEODETIC GAME. Indeed, such is the case for Proposition 6 in [20], where graphs with a unique minimum-size geodetic set are solved due to parity, since the simili-passing moves (adding to  $S$  vertices that are in the current geodetic closure) are available.

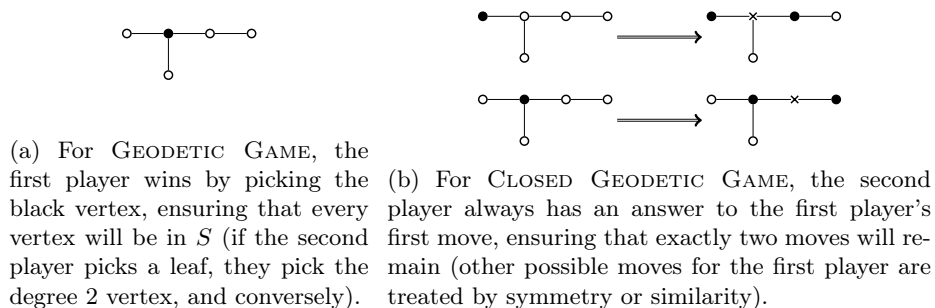


Fig. 2: An example of a graph with different outcomes for GEODETIC GAME and CLOSED GEODETIC GAME.

Other variants were later introduced, such as the *hull game*, where the *convex hull* (instead of the geodetic closure) of  $S$  is considered [4,15]. The outcomes for trees, cycles and complete graphs with rays were determined in [15], while [4] focused on Sprague-Grundy values for (among others) block graphs, lattice graphs, (generalized) wheels and complete multipartite graphs, as well as algebraic properties of the game.

Note that those games can be generalized as hypergraph games [23], where two players alternate selecting vertices of a hypergraph (potentially modifying it or under conditions) until those satisfy some condition related to the hyperedges. In particular, in GEODETIC GAME, the players add hyperedges corresponding to the shortest paths between the selected vertices, until every vertex is in at least one hyperedge.

### 1.3 Game definitions

GEODETIC GAME and CLOSED GEODETIC GAME are impartial games, a subset of combinatorial games which have been extensively studied. Since both players have the same possible moves, impartial games can have two possible *outcomes*: either the first player has a winning strategy ( $\mathcal{N}$ ) or the second player does ( $\mathcal{P}$ ). A position  $P'$  that can be reached by a move from a position  $P$  is called an *option* of  $P$ , the set of options of  $P$  is denoted by  $\text{opt}(P)$ . If every option of  $P$  is  $\mathcal{N}$ , then  $P$  is  $\mathcal{P}$ ; conversely, if any option is  $\mathcal{P}$ , then  $P$  is  $\mathcal{N}$  (the winning strategy for the first player is to play to a  $\mathcal{P}$  option). A refinement of outcomes are the *Sprague-Grundy values* [16,25] (also called *nim-values* or *nimbers*), denoted by  $\mathcal{G}(P)$  for a given position  $P$  and which can be mapped to nonnegative integers. The value of a position can be computed inductively, with  $\mathcal{G}(P) = \text{mex}(\{\mathcal{G}(P') : P' \in \text{opt}(P)\})$ , where  $\text{mex}(X)$  (for a set  $X$  of nonnegative integers) is the smallest nonnegative integer not in  $X$ . A position  $P$  is  $\mathcal{P}$  if and only if  $\mathcal{G}(P) = 0$ . Two positions  $P$  and  $P'$  are said to be *equivalent*, denoted by  $P \equiv P'$ , if  $\mathcal{G}(P) = \mathcal{G}(P')$ .

The *disjoint sum* of two positions  $P$  and  $Q$ , denoted by  $P + Q$ , is a position where each player chooses to play on one of the two positions at their turn. When one of them is finished, the game continues on the other one; the last player to play wins. For CLOSED GEODETIC GAME, a disjoint sum represents the disjoint union of two graphs. Computing the Sprague-Grundy value of a disjoint sum  $P + Q$  can be done by applying the so-called *nim-sum*, which is the bitwise XOR:  $\mathcal{G}(P + Q) = \mathcal{G}(P) \oplus \mathcal{G}(Q)$ . An example is given below for  $7 \oplus 10$ .

$$\begin{array}{r} 7 \rightarrow 111 \\ 10 \rightarrow 1010 \\ \hline 1101 \rightarrow 13 \end{array}$$

For more information and history about combinatorial games, we refer the reader to [1,5,12,24].

Due to the definition of the sum of positions, we can limit our study to connected graphs: studying a graph with several connected components is equivalent to studying each of the components independently and applying the nim-sum.

Furthermore, we will always consider that both players play optimally. Optimal play is understood as applying a winning strategy using as few rounds as possible if possible, and trying to make the game last as long as possible if no winning strategy exists. We will also make use of the following notation:

**Notation 1.** *Let  $G$  be a graph. We also denote the game position by  $G$ . Furthermore, let  $S$  be a subset of vertices of  $G$ , then the position denoted by  $G, S$  is the graph  $G$  where the vertices in  $S$  have already been selected.*

#### 1.4 Structure of the paper

We will begin in Section 2 by proving general results on CLOSED GEODETIC GAME as well as studying how to compute the outcome for the Cartesian product of several graphs in function of the outcomes of the individual graphs. Then, in Section 3, we will study some graph classes where we can predict exactly which vertices will be picked. Afterwards, Section 4 will be focused on highly symmetric graphs such as paths, cycles and grids, and we compute the Sprague-Grundy values for the former two. Finally, in Section 5, we present polynomial-time algorithms based on dynamic programming computing the Sprague-Grundy values of block graphs and cacti. Due to space restrictions, proofs of results marked with (\*) are omitted; they can be found in the full version of the paper [13].

## 2 General results and Cartesian product

In this section, we present a few general results that help the study of CLOSED GEODETIC GAME, be it by stating that some vertices have to be selected, by helping decompose a graph, or by showing how a well-known graph product affects the outcome of graphs.

First, we note that, in CLOSED GEODETIC GAME, due to the fact that vertices can be in the geodetic closure ( $S$ ) without being in the geodetic set  $S$  only if they are on some shortest path between two vertices, some vertices always have to be selected. For example, a leaf (*i.e.*, a degree 1 vertex), can never be in a shortest path between two other vertices. More generally, consider *simplicial* vertices, that is, vertices whose neighbourhood induces a complete subgraph.

**Lemma 1.** *In CLOSED GEODETIC GAME, all simplicial vertices of the graph will have to be selected.*

*Proof.* A simplicial vertex can never be on a shortest path between two other vertices, since all its neighbours are fully connected to each other. Hence, it can never be in a geodetic closure ( $S$ ) if it is not in  $S$  itself.  $\square$

*Articulation points*, that is, vertices whose removal disconnects the graph (also called *cut vertices* or *separating vertices*), are important for the study of CLOSED GEODETIC GAME. The following lemma will often be used to allow for decompositions:

**Lemma 2.** Let  $G$  be a graph with an articulation point  $u$ , such that components  $G_1, \dots, G_k$  are attached at  $u$ . In CLOSED GEODETIC GAME,  $\mathcal{G}(G, \{u\}) = \mathcal{G}(G_1, \{u\}) \oplus \dots \oplus \mathcal{G}(G_k, \{u\})$ .

*Proof.* When  $u$  is selected, any selected vertex  $x$  in  $G_i$  will never help cover any vertex in  $G_j$  for  $j \neq i$ . Indeed, all the shortest paths from  $x$  to any vertex in  $G_j$  will go through  $u$ , so, from the point of view of  $G_j$  selecting  $x$  after  $u$  has no impact. Hence, each  $G_i, \{u\}$  is an independent position, and we are playing on their disjoint union. This is depicted on Figure 3.  $\square$

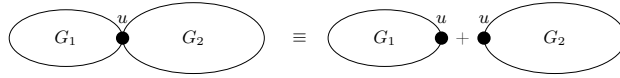


Fig. 3: Selecting an articulation point is equivalent to splitting the graph and playing independently on the components.

We also focus on a classical question whenever studying games: if we know the outcomes of two graphs, can we obtain the outcome of some composition of those graphs? This question stems from the disjoint sum, which is managed through the nim-sum of Sprague-Grundy values. We study how the Cartesian product of graphs interacts with the outcomes for CLOSED GEODETIC GAME.

Given two graphs  $G$  and  $H$ , their *Cartesian product*  $G \square H$  has vertices of the form  $(u, v)$  where  $u \in V(G)$  and  $v \in V(H)$  such that two vertices  $(u_1, v_1)$  and  $(u_2, v_2)$  are adjacent in  $G \square H$  if

- $u_1 = u_2$  and  $v_1$  is adjacent to  $v_2$  in  $H$ , or
- $v_1 = v_2$  and  $u_1$  is adjacent to  $u_2$  in  $G$ .

**Theorem 1.** Let  $G = G_1 \square G_2 \square \dots \square G_n$  be the Cartesian product of  $n$  graphs. The graph  $G$  is  $\mathcal{N}$  for CLOSED GEODETIC GAME if and only if  $G_i$  is  $\mathcal{N}$  for CLOSED GEODETIC GAME for all  $i \in \{1, 2, \dots, n\}$ .

*Proof.* First, we will show that it is enough to prove the statement for  $n = 2$ . To this end, suppose that we have proved  $H = H_1 \square H_2$  is  $\mathcal{N}$  for CLOSED GEODETIC GAME if and only if both  $H_1$  and  $H_2$  are  $\mathcal{N}$ . On the one hand, let  $G = G_1 \square G_2 \square \dots \square G_n$  where  $G_i$  is  $\mathcal{N}$  for all  $i \in \{1, 2, \dots, n\}$ . In this case, we can use our assumption of  $H = H_1 \square H_2$  is  $\mathcal{N}$  if and only if both  $H_1$  and  $H_2$  are  $\mathcal{N}$ , recursively, to conclude that  $G$  is  $\mathcal{N}$ . On the other hand, let  $G = G_1 \square G_2 \square \dots \square G_n$  where  $G_i$  is  $\mathcal{P}$  for some  $i \in \{1, 2, \dots, n\}$ . Let  $G' = G_1 \square \dots \square G_{i-1} \square G_{i+1} \square \dots \square G_n$ . Observe that  $G' \square G_i = G$ . Hence, we can use our assumption of  $H = H_1 \square H_2$  is  $\mathcal{N}$  if and only if both  $H_1$  and  $H_2$  are  $\mathcal{N}$  to conclude that  $G$  is  $\mathcal{P}$ . Therefore, it is enough to prove the statement of our theorem for  $n = 2$ , and henceforth we will focus on this simpler statement. We will need the next two useful claims.

**Claim 1** (\*). Let  $G = G_1 \square G_2$ ,  $x_1, x_2 \in V(G_1)$ ,  $y_1, y_2 \in V(G_2)$ ,  $a$  and  $b$  respectively be the distance between  $x_1$  and  $x_2$  in  $G_1$  and the distance between  $y_1$  and  $y_2$  in  $G_2$ . Then the distance between  $(x_1, y_1)$  and  $(x_2, y_2)$  is  $a + b$  in  $G$ .

**Claim 2.** Let  $G = G_1 \square G_2$ ,  $S_1 \subseteq V(G_1)$ ,  $S_2 \subseteq V(G_2)$ , and  $S = S_1 \times S_2 \subseteq V(G)$ . We have  $(u, v) \in (S)$  if and only if  $(u, v) \in (S_1) \times (S_2)$ .

*Proof of Claim 2.* In one direction, assume that  $(u, v) \in (S)$ . Thus, there exists  $(x_1, y_1), (x_2, y_2) \in S$  such that  $(u, v)$  lies on some shortest path  $P$  between  $(x_1, y_1)$  and  $(x_2, y_2)$ . Let  $P'$  be the subpath of  $P$  that connects  $(x_1, y_1)$  to  $(u, v)$ . Similarly, let  $P''$  be the subpath of  $P$  that connects  $(u, v)$  to  $(x_2, y_2)$ . Note that both  $P'$  and  $P''$  are shortest paths.

Denote the length of a path  $X$  (that is, its number of edges) by  $\text{lt}(X)$ . By Claim 1, we know that there are shortest paths  $P'_1$  in  $G_1$  and  $P'_2$  in  $G_2$  connecting  $x_1$  to  $u$  and  $y_1$  to  $v$ , respectively, satisfying  $\text{lt}(P'_1) + \text{lt}(P'_2) = \text{lt}(P')$ . Similarly, we know that there are shortest paths  $P''_1$  and  $P''_2$  in  $G_2$  connecting  $u$  to  $x_2$  and  $v$  to  $y_2$ , respectively, satisfying  $\text{lt}(P''_1) + \text{lt}(P''_2) = \text{lt}(P'')$ . Let  $P_1$  (resp.,  $P_2$ ) be the path obtained through concatenation of  $P'_1$  and  $P''_1$  (resp.,  $P'_2$  and  $P''_2$ ). Thus, we must have  $\text{lt}(P_1) + \text{lt}(P_2) = \text{lt}(P)$ . This implies that  $P_1$  and  $P_2$  are shortest paths due to Claim 1. Thus,  $(u, v) \in (S)$  implies  $(u, v) \in (S_1) \times (S_2)$ .

In the other direction, assume that  $(u, v) \in (S_1) \times (S_2)$ . Thus, there are  $x_1, x_2 \in S_1$  such that  $u$  lies on the path  $P_1$  connecting  $x_1$  and  $x_2$ . Moreover, without loss of generality, we may assume that the path is the concatenation of the subpaths  $P'_1$  connecting  $x_1$  and  $u$  and  $P''_1$  connecting  $u$  and  $x_2$ . Similarly, there are  $y_1, y_2 \in S_2$  such that  $v$  lies on the path  $P_2$  connecting  $x_1$  and  $x_2$  where  $P_2$  is the concatenation of the subpaths  $P'_2$  connecting  $y_1$  and  $v$  and  $P''_2$  connecting  $v$  and  $y_2$ . By Claim 1, we know that there are two paths  $P'$  connecting  $(x_1, y_1)$  and  $(u, v)$  and  $P''$  connecting  $(u, v)$  and  $(x_2, y_2)$  in  $G$ , and satisfying  $\text{lt}(P'_1) + \text{lt}(P'_2) = \text{lt}(P')$  and  $\text{lt}(P''_1) + \text{lt}(P''_2) = \text{lt}(P'')$ . In particular, this implies that  $P'$  and  $P''$  are shortest paths.

Let  $P$  be the path obtained by concatenation of  $P'$  and  $P''$ . Observe that  $P$  connects  $(x_1, y_1)$  and  $(x_2, y_2)$  and passes through the vertex  $(u, v)$ . Note that

$$\text{lt}(P) = \text{lt}(P') + \text{lt}(P'') = \text{lt}(P'_1) + \text{lt}(P'_2) + \text{lt}(P''_1) + \text{lt}(P''_2) = \text{lt}(P_1) + \text{lt}(P_2).$$

Thus, by Claim 1,  $P$  must be a shortest path. ◻

In one direction, assume that  $G_1$  and  $G_2$  are  $\mathcal{N}$ . We show that  $G = G_1 \square G_2$  is also  $\mathcal{N}$ .

We will play the game simultaneously on  $G$ ,  $G_1$  and  $G_2$ . The vertex sets that will be constructed during the games are  $S$ ,  $S_1$  and  $S_2$ , respectively. We will maintain  $S = S_1 \times S_2$ . The strategy of the first player on  $G$  will be derived from the strategies of the first player on  $G_1$  and  $G_2$ . The second player will play on  $G$ , and the corresponding moves will be replicated on  $G_1$  and  $G_2$ .

Let us explain first how a move is replicated or translated. At any point in the game, if a player selects the vertex  $(u, v)$  in  $G$ , we know that  $(u, v) \notin (S)$  at the time it is selected. However, by Claim 2, it is possible to have either  $u \in (S_1)$  or  $v \in (S_2)$ , but not both. If  $u \in (S_1)$ , then the selection of  $(u, v)$  corresponds to

a “dummy move” in  $G_1$ . If  $u \notin (S_1)$ , then the selection of  $(u, v)$  corresponds to selecting  $u$  in  $G_1$ . Similarly, one can translate the moves on  $G$  to  $G_2$ .

The selection of  $u$  and  $v$  in  $G_1$  and  $G_2$ , respectively, corresponds to the selection of  $(u, v)$  in  $G$ . Note that either  $u \notin (S_1)$  or  $v \notin (S_2)$  at the time of their respective selection, implying  $(u, v) \notin (S) = (S_1) \times (S_2)$  due to Claim 2, and thus selecting  $(u, v)$  is a valid move.

The first player will begin the game by selecting vertices  $u_1, v_1$  in  $G_1, G_2$ , respectively, according to the corresponding winning strategies. In  $G$ , the vertex  $(u_1, v_1)$  is selected by the first player. After that the players take turns to make their moves. Let us assume that the second player has made a move by selecting the vertex  $(u, v)$  in  $G$ . This moves translates to moves in  $G_1, G_2$ . However due to Claim 2, we know that at most one of these translated moves can be a dummy move. Without loss of generality assume that the second player’s move translates to a dummy move in  $G_1$ , and a valid move in  $G_2$ . In that case, the first player makes a dummy move in  $G_1$  and provides a response by selecting  $v'$  in  $G_2$  according to winning strategy. These moves are translated to  $G$  as selection of the vertex  $(u, v')$ , which is a valid move due to Claim 2. If the second player’s move in  $G$  translates to valid moves in both  $G_1$  and  $G_2$ , then the first player responds in both the graphs according to corresponding winning strategies by selecting  $u', v'$  in  $G_1, G_2$ , respectively. This translates to the valid move of selecting  $(u', v')$  in  $G$  by the first player due to Claim 2.

Hence, the first player always has a valid response to the second player’s move. Thus,  $G$  is  $\mathcal{N}$ .

In the other direction, assume that  $G_2$  is  $\mathcal{P}$ . We show that  $G = G_1 \square G_2$  is also  $\mathcal{P}$ . In this case, the second player will simulate the game on both graphs, selecting  $(u, v)$  in  $G \square H$  where  $u$  is either the winning move (if  $G_1$  is  $\mathcal{P}$  or the first player played a non-optimal move) or the same vertex that the first player selected, and  $v$  is their winning move in  $H$  for replying to the first player. The game will end in  $G \square H$  when both  $G$  and  $H$  will be over, and thus the second player will always have an answer to the first player’s moves. This is consistent due to Claim 2.  $\square$

### 3 SHE LOVES ME-SHE LOVES ME NOT situations

SHE LOVES ME-SHE LOVES ME NOT is one of the simplest combinatorial games: the two players alternate removing blossoms from a flower until it is empty. It has become a shorthand for easy, binary combinatorial games where every possible move from the initial position will be played. Note that, in those cases, the Sprague-Grundy values alternate between 0 (for an even number of moves) and 1 (for an odd number of moves), since no real strategy is involved. Some complex combinatorial games can be equivalent to SHE LOVES ME-SHE LOVES ME NOT depending on the initial position. It is in particular the case for CLOSED GEODETIC GAME in some cases:

**Proposition 1 (\*)**. *A game of CLOSED GEODETIC GAME on the complete graph  $K_n$  or a star  $K_{1,n}$  will see all its vertices selected.*



(a) On this graph, the second player wins: if the first player selects a leaf, they select the adjacent internal vertex, and conversely. There is then a pairing strategy for the second player. (b) On this graph, the first player wins by picking the black vertex (this can be shown with case analysis).

Fig. 4: Two graphs of same order (8) with unique minimum-size geodetic sets of same size (5) but different outcomes for CLOSED GEODETIC GAME.

Note that those two graphs have a unique minimum-size geodetic set, so one could think that such parity results could be obtained on any graph family with a unique, well-identified minimum-size geodetic set on an underlying structure. However, such a general structural results would be difficult to achieve, as illustrated by Figure 4: those two graphs are based on the same underlying structure (a triangle with leaves), have the same order and unique minimum-size geodetic sets of the same size, but different outcomes for CLOSED GEODETIC GAME.

For other graphs, the game is not fully reducible to a SHE LOVES ME-SHE LOVES ME NOT, but is still close since, under optimal play, we can determine exactly which vertices will be selected.

**Proposition 2 (\*)**. *Let  $n \geq 2$ . A game of CLOSED GEODETIC GAME on the complete bipartite graph  $K_{2,n}$  will see all the vertices of its larger part selected.*

We can also compute the Sprague-Grundy values of complete bipartite graphs, for which one part will always be fully selected:

**Proposition 3 (\*)**. *Let  $m$  and  $n$  be two integers such that  $m, n \geq 2$ . If  $m$  and  $n$  have the same parity, then,  $\mathcal{G}(K_{m,n}) = 0$ . Otherwise,  $\mathcal{G}(K_{m,n}) = 2$ .*

## 4 Symmetry strategies

The outcome of cycles can be fully derived from a symmetry strategy. Furthermore, we can also characterize the Sprague-Grundy value of a cycle:

**Theorem 2 (\*)**. *For any positive integer  $n$ ,  $\mathcal{G}(C_n) = n \bmod 2$ .*

We can also characterize the Sprague-Grundy value of a cycle with one or two selected vertices:

**Proposition 4 (\*)**. *For any positive integer  $k$ :*

- $\mathcal{G}(C_{2k}, \{u\}) = k$  and  $\mathcal{G}(C_{2k+1}, \{u\}) = 0$  for any vertex  $u$ ;
- $\mathcal{G}(C_{2k+1}, \{u_0, u_i\}) = k + 1 - i$  for  $i \in \{1, \dots, k\}$  (denoting the vertices by  $u_0, \dots, u_{2k}$  in the cyclic order).

Theorem 1 combined with Theorem 4 already give us the outcome for multidimensional grids, but we can also prove it with a symmetry strategy:

**Theorem 3 (\*)**. *CLOSED GEODETIC GAME has outcome  $\mathcal{N}$  for a multidimensional grid if and only if all its dimensions are odd.*

Paths are another graph class where the outcome can be derived from a symmetry strategy. Going further, we hereby give a full characterization of the Sprague-Grundy values of paths (which complements the possible computation using the algorithm for trees from Araujo *et al.* [2]). Note that, while the value is expected, the proof itself is more involved.

**Theorem 4 (\*)**. *For any positive integer  $n$ ,  $\mathcal{G}(P_n) = n \bmod 2$ .*

*Proof (Sketch)*. If  $n$  is even, the second player has an easy symmetry strategy. If  $n$  is odd, the first player wins by picking the middle vertex. We then prove that no option of  $P_n$  has Sprague-Grundy value 1, by showing that all of those options have themselves an option with Sprague-Grundy value 1. In order to do this, we prove that  $\mathcal{G}(P_{4m+2}, \{u_{2m+1}\}) = 1$ , and exhibit such options for each option of  $P_n$ .  $\square$

## 5 Sprague-Grundy values of block graphs and cacti

In their paper [2, Algorithm 2], Araujo *et al.* presented a linear-time algorithm for computing the Sprague-Grundy value of a given tree for CLOSED GEODETIC GAME. In order to do this, they use dynamic programming with subtrees that decrease in size at each step: for any given selected vertex  $u$ , each of the subtrees rooted at a neighbour of  $u$  are completely independent in CLOSED GEODETIC GAME, since by Lemma 2 selecting a vertex in a subtree will never impact any of the other subtrees. The base case happens when the subtree is either empty or a single vertex. Hence, it is possible to define a recursive algorithm computing, for each vertex of the tree, the Sprague-Grundy value of the subtrees rooted at each of its neighbours, and then apply the mex operation to the resulting set. Since each subtree is independent, and the intermediate values can be stored, the algorithm returns the result in linear-time.

In this section, we extend this idea to other tree-like graph classes: block graphs and cacti. Indeed, the building blocks of those graphs are simple graphs for which we can easily compute the Sprague-Grundy values: complete graphs and cycles. We will present polynomial-time algorithms for those classes, adapted from the trees' algorithm in [2]. Note that the building blocks have specific properties, and are close to the class of *geodetic graphs*, that is, graphs for which there is a unique shortest path between any pair of vertices (this class includes trees, block graphs, and cacti with only odd cycles); the only non-geodetic class is cacti with even cycles, for which the shortest paths between two vertices will share common articulation points and thus are almost as constrained.

## 5.1 Block graphs

A graph is a *block graph* if every biconnected component is a clique. Informally, we can see a block graph as a tree-like structure in which cliques are connected to each other by one of their vertices.

**Theorem 5.** *There is a linear-time algorithm computing the Sprague-Grundy value of a block graph for CLOSED GEODETIC GAME.*

*Proof.* The algorithm is based on dynamic programming. First, for a given block graph  $G$  and one of its vertices  $u$ , we present how to recursively compute  $\mathcal{G}(G, \{u\})$ , that is, we consider that the first player selects  $u$ . We will show that some clique vertices are equivalent for this first move, and that we can store the intermediary results to reuse them for other vertices (thus avoiding polynomial-time when applying the algorithm for all the non-equivalent vertices of  $G$ ). Note that the base case is when the graph is a complete graph, since we can compute its Sprague-Grundy value using Proposition 1 (each selected vertex within a complete graph flips the Sprague-Grundy value between 0 and 1).

There are two types of vertices in  $G$ : those that are articulation points between several cliques, and those that are not. First, assume that the selected vertex  $u$  is an articulation point, then we can compute  $\mathcal{G}(G, \{u\})$  by using Lemma 2 and store the intermediate values for future computations.

Assume now that the selected vertex  $u$  is not an articulation point, hence  $u$  is a vertex in a given clique  $A$  of  $G$ . Note that selecting any other non-articulation point of  $A$  is equivalent to selecting  $u$ . Now, consider the components that are attached to  $A$  in  $G$  (that is, they share one vertex with  $A$ ), denote them by  $B_1, \dots, B_k$  and the articulation point between  $A$  and  $B_i$  by  $v_i$ . Note that selecting any other non-articulation point in  $A$  will not affect any of the  $B_i$ 's, and that selecting  $v_i$  can only affect  $B_i$ . Hence, we can decompose the game into almost-independent subgraphs:  $A \setminus \{v_1, \dots, v_k\}$  with the vertex  $u$  selected, and each component containing  $B_i$  with a leaf attached to  $v_i$ , which is selected too. This is depicted on Figure 5. Note that this decomposition works since  $A$  is a clique, and so all its vertices are at distance 1 from each other.

Now, we need to see how we can decompose the graph with one selected vertex  $u$  when another vertex  $v$  is selected. Again, if  $v$  is an articulation point, we can use Lemma 2. If  $v$  is also in  $A$ , then we do not need to decompose further. So, assume that  $v$  is in another clique  $B$  and is not an articulation point. We can apply the same principle of decomposition, after covering the vertices in the shortest paths between  $u$  and  $v$ . Note that if  $A$  and  $B$  are not adjacent cliques, then let  $C$  be a clique with two vertices on the shortest path between  $u$  and  $v$ , two vertices of  $C$  are covered and the other ones will have no impact on  $A$  or  $B$  for the rest of the game. This decomposition is depicted on Figure 6.

Note that both decompositions have the property that a subgraph that is not a clique has at most one selected vertex. This covers all possible cases, and thus the algorithm will further decompose the input block graph until it is a disjoint sum of cliques, for which the Sprague-Grundy value can be computed in constant time. By using the nim-sum and the mex, we can then go back and compute

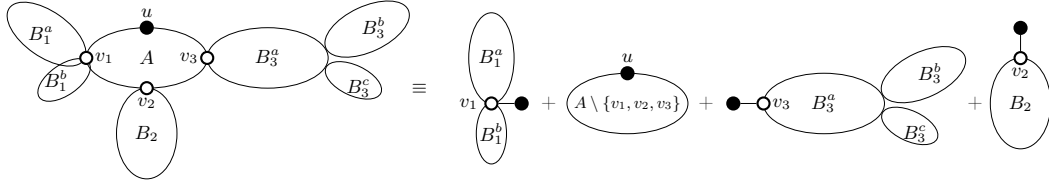


Fig. 5: Selecting a vertex in a clique in a block graph allows us to decompose the graph as follows (here, each  $B_i^x$  is a clique).

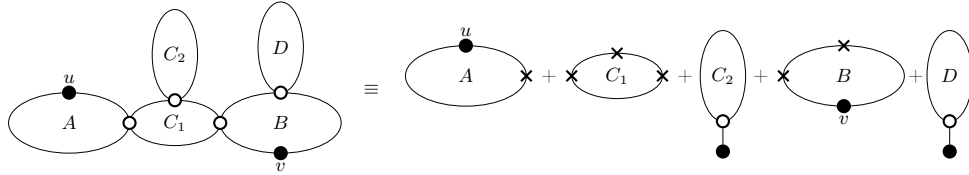


Fig. 6: Selecting a second vertex in a block graph allows us to decompose it further.

the Sprague-Grundy value of the block graph. The algorithm ends in linear time for a given vertex, and allows us to store the intermediary results for future computations. In fact, the number of block components being linear, any future computation will see similar decompositions, and thus reuse the intermediary results, hence allowing the algorithm to remain linear-time when considering all possible first moves.  $\square$

## 5.2 Cacti

A *cactus* (plural: cacti) is a graph where every edge is in at most one simple cycle. Alternately, cacti have a tree-like structure in which cycles and trees are connected to each other by one of their vertices.

**Theorem 6 (\*)**. *There is a quadratic-time algorithm computing the Sprague-Grundy value of a cactus for CLOSED GEODETIC GAME.*

*Proof (Sketch)*. The algorithm is also based on dynamic programming. We identify three types of cacti with either one or two selected vertices, and show that any move in either type yields a graph that can be decomposed (thanks to Lemma 2) into a disjoint sum of subgraphs, each of one of those three types. We use the algorithm in [2], Theorems 2 and 4 and Proposition 4 as base cases.  $\square$

## 6 Future research

In this paper, we explored CLOSED GEODETIC GAME in three directions: general results such as graph operations (Cartesian product) and decomposition

through articulation points, characterization of outcomes and Sprague-Grundy values, and polynomial-time algorithms based on dynamic programming for constrained graph classes. However, many more general graph classes remain open, algorithmically as well as structurally.

A first interesting research direction would be to fully characterize the graphs for which CLOSED GEODETIC GAME is a SHE LOVES ME-SHE LOVES ME NOT game, or those that are close to such a situation. We already exhibited complete graphs and stars as SHE LOVES ME-SHE LOVES ME NOT games in Section 3, but there might be other very structured classes with many simplicial vertices that are in the same situation. Note that the unicity of minimum-size geodetic sets and the structural backbone of the graph are not sufficient conditions for this property.

Another idea would be to further the work initiated in Section 4 and to define a type of graph symmetry adapted to CLOSED GEODETIC GAME, that would allow us to obtain the outcomes of other structured graph classes. Expansions of cycles could be good candidates.

As explained in Section 5, the dynamic programming framework we used for block graphs and cacti requires strong constraints for shortest paths between vertices, which do not hold for more general graph classes such as outerplanar graphs. A research direction would be to try to express weaker constraints allowing a dynamic programming framework based on decomposition into independent subcomponents to apply for other classes. Outerplanar graphs are a natural candidate, since the geodetic set problem is polynomial for them but NP-hard for planar graphs.

Finally, other graph products than Cartesian (studied in Section 2) could be explored, such as the tensor, strong, lexicographical, modular or rooted products. However, those products may be more involved: for example, the outcome of the tensor product of two graphs is not necessarily a function of the outcomes of the two graphs. Some further conditions on the structure of the graphs or their product may be needed to obtain similar results.

**Acknowledgements.** The authors would like to thank the universities' libraries, without which the seminal paper [7] wouldn't be obtainable; as well as Sagnik Sen for his advice and tremendous help.

## References

1. Albert, M., Nowakowski, R., Wolfe, D.: Lessons in play: an introduction to combinatorial game theory. AK Peters/CRC Press (2019)
2. Araújo, S.N., Brito, J.M., Folz, R., de Freitas, R., Sampaio, R.M.: Graph convexity impartial games: Complexity and winning strategies. Theoretical Computer Science **998**, 114534 (2024)
3. Atici, M.: Computational complexity of geodetic set. International journal of computer mathematics **79**(5), 587–591 (2002)
4. Benesh, B.J., Ernst, D.C., Meyer, M., Salmon, S.K., Sieben, N.: Impartial geodetic building games on graphs. International Journal of Game Theory pp. 1–34 (2025)

5. Berlekamp, E.R., Conway, J.H., Guy, R.K.: Winning ways for your mathematical plays. AK Peters/CRC Press (2001–2004)
6. Buckley, F., Harary, F., Quintas, L.: Extremal results on the geodetic number of a graph. *Scientia* **2A**, 17–26 (1988)
7. Buckley, F., Harary, F.: Closed geodetic games for graphs. *Congr. Numer* **47**, 131–138 (1985)
8. Buckley, F., Harary, F.: Geodetic games for graphs. *Quaestiones Mathematicae* **8**(4), 321–334 (1985)
9. Buckley, F., Harary, F.: Distance in graphs. New York: Addison and Wesley (1990)
10. Chakraborty, D., Das, S., Foucaud, F., Gahlawat, H., Lajou, D., Roy, B.: Algorithms and complexity for geodetic sets on planar and chordal graphs. In: 31st International Symposium on Algorithms and Computation (ISAAC 2020). Schloss Dagstuhl-Leibniz-Zentrum für Informatik (2020)
11. Chakraborty, D., Gahlawat, H., Roy, B.: Algorithms and complexity for geodetic sets on partial grids. *Theoretical Computer Science* **979**, 114217 (2023)
12. Conway, J.H.: On numbers and games. CRC Press (2000)
13. Dailly, A., Gahlawat, H., Myint, Z.M.: The closed geodetic game: algorithms and strategies. arXiv preprint arXiv:2409.20505 (2024)
14. Ekim, T., Erey, A., Heggernes, P., van't Hof, P., Meister, D.: Computing minimum geodetic sets of proper interval graphs. In: Latin American Symposium on Theoretical Informatics. pp. 279–290. Springer (2012)
15. Fraenkel, A.S., Harary, F.: Geodetic contraction games on graphs. *International Journal of Game Theory* **18**, 327–338 (1989)
16. Grundy, P.M.: Mathematics and games. *Eureka* **2**, 6–8 (1939)
17. Harary, F.: Convexity in graphs: achievement and avoidance games. In: North-Holland Mathematics Studies, vol. 87, p. 323. Elsevier (1984)
18. Harary, F., Loukakis, E., Tsouros, C.: The geodetic number of a graph. *Mathematical and Computer Modelling* **17**(11), 89–95 (1993)
19. Harary, F., Nieminen, J.: Convexity in graphs. *Journal of Differential Geometry* **16**(2), 185–190 (1981)
20. Haynes, T.W., Henning, M., Tiller, C.A.: Geodetic achievement and avoidance games for graphs. *Quaestiones Mathematicae* **26**(4), 389–397 (2003)
21. Mezzini, M.: Polynomial time algorithm for computing a minimum geodetic set in outerplanar graphs. *Theoretical Computer Science* **745**, 63–74 (2018)
22. Nečásková, M.: A note on the achievement geodetic games. *Quaestiones Math.* **12**, 115–119 (1988)
23. Sieben, N.: Impartial hypergraph games. *The Electronic Journal of Combinatorics* pp. P2–13 (2023)
24. Siegel, A.N.: Combinatorial game theory, vol. 146. American Mathematical Soc. (2013)
25. Sprague, R.: Über mathematische kampfspiele. *Tohoku Mathematical Journal, First Series* **41**, 438–444 (1935)