

On Parameterized Complexity of Binary Networked Public Goods Game

Arnab Maiti

Indian Institute of Technology
Kharagpur, India
arnabmaiti@iitkgp.ac.in

Palash Dey

Indian Institute of Technology
Kharagpur, India
palash.dey@cse.iitkgp.ac.in

ABSTRACT

In the Binary Networked Public Goods (BNPG for short) game, every player needs to decide if she participates in a public project whose utility is shared equally by the community. We study the problem of deciding if there exists a pure strategy Nash equilibrium (PSNE) in such games. The problem is already known to be NP-complete. This casts doubt on predictive power of PSNE in BNPG games. We provide fine-grained analysis of this problem under the lens of parameterized complexity theory. We consider various natural graph parameters and show $W[1]$ -hardness, XP, and FPT results. Hence, our work significantly improves our understanding of BNPG games where PSNE serves as a reliable solution concept. We finally prove that some graph classes, for example path, cycle, bi-clique, and complete graph, always have a PSNE if the utility function of the players are same.

KEYWORDS

Fixed-Parameter Tractability; W-Hierarchy; Binary Networked; Public Goods Game; Pure Strategy Nash Equilibrium; Heterogeneous; Fully Homogeneous

ACM Reference Format:

Arnab Maiti and Palash Dey. 2022. On Parameterized Complexity of Binary Networked Public Goods Game. In *Proc. of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022), Online, May 9–13, 2022, IFAAMAS*, 9 pages.

1 INTRODUCTION

In a public goods game, players need to decide if they contribute in a public project and, if yes, then how much. The outcome of such public projects is typically shared equally by all the players. Public goods games are effective in modeling tension between individual cost vs community well beings [16, 24]. One of the well-explored variants of the above game is the *networked* public goods game where we assume a network structure on the players and the utilities of individual players depend on the action of them and their neighbors only [2].

An important class of networked public goods game is the *binary* networked public goods (BNPG for short) game where players only need to decide if they participate (play 1) in the public project or not (play 0) [10]. Although this seems restricted, such games are still powerful enough to model various important real world application scenarios. For some motivating examples, let us think of an air-borne virus pandemic like Covid-19 where individuals need to decide whether to wear a mask or not. While individuals may feel

uncomfortable while wearing a mask, the benefits of herd immunity, if achieved by a large fraction of population wearing a mask, will be shared by the entire community. Indeed, there are reports that a considerable fraction of population refuse to wear a mask during Covid-19 pandemic [3, 27]. Another important application is whether to report a crime or not. While individuals who report crimes may be at risk, the benefit of having lower crime rates will be enjoyed by the entire community. The general observation at many places is that crimes are often under-reported [21].

Computing a pure strategy Nash equilibrium (PSNE) in any game is a fundamental question. The concept of Nash equilibrium guides social planner to predict how players will act in a strategic setting and act accordingly. We know that the EXISTS-PSNE problem, where we are asked to decide if a BNPG game has a PSNE, is NP-complete [29]. In this work, we provide a comprehensive study of the parameterized complexity of the EXISTS-PSNE problem.

Related Work. The immediate predecessor of our work is [29] where the authors initiate the algorithmic question of EXISTS-PSNE. Our work broadly belongs to the field of graphical games where there is a graph structure on the players and a player's utility depends only on the actions of her neighbors [14]. A central question in graphical games is to find complexity of the problem of computing an equilibrium [8, 9, 11]. Network public goods games are a special case of graphical games where the utility of players depends only on the sum of the "efforts" put in by neighbors and the cost of her action. Many models of the network public goods game have been explored which are fine-tuned to different applications. Important examples of such applications include economics, research collaboration, social influence, etc. [4, 5, 25, 26]. The BNPG model is closely related to that proposed in Bramoullé et al. [2]. There are however two qualitative distinctions (a) Bramoullé et al. focus on the continuous investment model whereas BNPG model focuses on binary investment decisions and (b) Bramoullé et al. assume homogeneous concave utilities whereas BNPG model considers a more general setting. Supermodular network games [20] and best-shot games (which is actually a special case of BNPG game) [7], etc. [10, 17, 18] are other important variations of graphical games. In the model of Supermodular network games, each agent's payoff is a function of the aggregate action of its neighbors and it exhibits strategic complementarity. An important example of supermodular games on graphs are technology adoption games which have been studied in the social network literature [12, 15, 22].

Parameters. As EXISTS-PSNE is NP-complete [29], we provide a comprehensive study of the parameterized complexity of the EXISTS-PSNE problem w.r.t. the following parameters:

- *Maximum Degree:* Many applications of BNPG games involve human beings as nodes in the network. Due to

human cognitive limitation, such graphs often exhibit small maximum degree. With this motivation, we consider the maximum degree of the graph as our parameter.

- **Diameter:** Graphs which involve human beings as nodes tend to have a small diameter. Therefore, we consider the diameter of the graph as our parameter.
- **Distance from tree and complete graph:** Trees and complete graphs are important classes of graphs in the context of BNPG games. It is already known from previous work that EXISTS-PSNE is polynomial-time solvable for trees and complete graphs. Therefore the next natural question would be to check the tractability of those instances where the graphs are quite close to being a tree or a complete graph. For this purpose, we consider the parameters distance from tree, which is also known as circuit rank, and distance from complete graphs [Definition 4.13].
- **Treedepth and Treewidth:** We also consider treedepth and treewidth as parameters as they have often turned out to be useful parameters to obtain a fixed-parameter-tractable (FPT) algorithm for many classical problems for which it is known that the problem is polynomial-time solvable for trees.
- **Number of participating and non-participating players:** One may wish to know what are the equilibria during a pandemic like Covid-19 example where most and least people wear masks. For such scenarios, the number of participating (who play 1) and non-participating players (who play 0) are the natural parameters.

2 PRELIMINARIES

For a set X , we denote its power set by 2^X . We denote the set $\{1, \dots, n\}$ by $[n]$. For 2 sets X and Y , we denote the set of functions from X to Y by \mathcal{Y}^X .

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be an undirected graph with n vertices. An edge between $u, v \in \mathcal{V}$ is represented by $\{u, v\}$. In a graph \mathcal{G} , we denote the degree of any vertex v by $d(v)$. For a subset $\mathcal{U} \subseteq \mathcal{V}$ of vertices (respectively a subset $\mathcal{F} \subseteq \mathcal{E}$ of edges), we denote the subgraph induced by \mathcal{U} (respectively \mathcal{F}) by $\mathcal{G}[\mathcal{U}]$ (respectively $\mathcal{G}[\mathcal{F}]$). A Binary Networked Public Goods (BNPG for short) game can be defined on \mathcal{G} as follows. The set of players is \mathcal{V} . The strategy set of every player is $\{0, 1\}$. We denote the number of neighbors of w in \mathcal{G} who play 1 in the strategy profile $(x_v)_{v \in \mathcal{V}}$ by n_w ; that is, $n_w = |\{u \in \mathcal{V} : \{u, w\} \in \mathcal{E}, x_u = 1\}|$. For a strategy profile $(x_v)_{v \in \mathcal{V}} \in \{0, 1\}^{|\mathcal{V}|}$, the utility $U_w((x_v)_{v \in \mathcal{V}})$ of player (without abusing the notation much) $w \in \mathcal{V}$ is defined as follows.

$$U_w((x_v)_{v \in \mathcal{V}}) = U_w(x_w, n_w) = g_w(x_w + n_w) - c_w \cdot x_w$$

where $g_w : \mathbb{N} \cup \{0\} \rightarrow \mathbb{R}^+$ is a non-decreasing function in x and $c_w \in \mathbb{R}^+$ is a constant. We denote a BNPG game by $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$. For any number $n \in \mathbb{N} \cup \{0\}$ and function $g : \mathbb{N} \cup \{0\} \rightarrow \mathbb{R}^+$, we define $\Delta g(n) = g(n+1) - g(n)$. In general, every player $w \in \mathcal{V}$ has a different mapping function $g_w(\cdot)$ and hence we call this version of the game a *heterogeneous* BNPG game. If not mentioned otherwise, by BNPG game, we refer to a heterogeneous BNPG game. In this paper, we also study the following three special cases – (i) *homogeneous*: $g_w = g$ for all $w \in \mathcal{V}$, (ii) *fully homogeneous*: homogeneous and $c_w = c$ for

all $w \in \mathcal{V}$ and (iii) *strict*: for every player $w \in \mathcal{V}$, we have $U_w(x_w = 0, x_{-w}) \neq U_w(x_w = 1, x_{-w})$ for every strategy profile x_{-w} of other players. So a BNPG game is strict if and only if $\Delta g_w(k) \neq c_w, \forall w \in \mathcal{V}, \forall k \in \{0, 1, \dots, d(w)\}$

A strategy profile $(x_v)_{v \in \mathcal{V}}$ is called a *pure-strategy Nash Equilibrium (PSNE)* of a BNPG game if we have $U_v(x_v, x_{-v}) \geq U_v(x'_v, x_{-v}) \forall x'_v \in \{0, 1\}, \forall v \in \mathcal{V}$. We call the problem of deciding if there exists a PSNE in BNPG games as EXISTS-PSNE.

For a player w in a BNPG game $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$, we define her best response function $\beta_w : \{0, 1, \dots, n-1\} \rightarrow 2^{\{0,1\}} \setminus \{\emptyset\}$ as follows. For every $k \in \{0, 1, \dots, n-1\}$ and $a \in \{0, 1\}$, we have $a \in \beta_w(k)$ if and only if, for every strategy profile x_{-w} of players other than w where exactly k players in the neighborhood of w play 1, we have $U_w(x_w = a, x_{-w}) \geq U_w(x_w = a', x_{-w})$ for all $a' \in \{0, 1\}$. The following lemma proves that, for every function β_w , there is a function $g_w : \mathbb{N} \cup \{0\} \rightarrow \mathbb{R}^+$ and constant c_w such that β_w is the best response function.

LEMMA 2.1 (★). *Let $\beta : \{0, 1, \dots, n-1\} \rightarrow 2^{\{0,1\}} \setminus \{\emptyset\}$ be an arbitrary function. Then we can compute in polynomial (in n) time a function $g : \mathbb{N} \cup \{0\} \rightarrow \mathbb{R}^+$ and constant c such that β is the corresponding best response function.*

We call a function $f : \mathbb{N} \cup \{0\} \rightarrow \mathbb{R}^+$ *sub-additive* if $f(x+y) \leq f(x) + f(y)$ for every $x, y \in \mathbb{N} \cup \{0\}$ and *additive* if $f(x+y) = f(x) + f(y)$. We call a BNPG game $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ *sub-additive* (respectively *additive*) if g_v is sub-additive (respectively additive) for every $v \in \mathcal{V}$.

Parameterized Complexity. A parameterized problem is represented by the tuple (x, k) , where k is the parameter. *Fixed parameter tractability* (FPT) refers to the solvability of a given instance (x, k) in time $f(k) \cdot p(|x|)$, where p is a polynomial in the input size $|x|$ and f is an arbitrary computable function of k . We use the notation $\mathcal{O}^*(f(k))$ to denote $\mathcal{O}(f(k) \text{poly}(|x|))$. There is a hierarchy of complexity classes above FPT, such as W [1], W [2], para-NP, and showing that a parameterized problem is hard for one of these complexity classes would imply that the problem may not be fixed-parameter tractable. **XP** is the class of parameterized problems that can be solved in time $n^{f(k)}$, where k is the parameter, n is the input size and f is some computable function.

Definition 2.2. [6] A **tree decomposition** of a graph G is a pair $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$, where T is a tree whose every node t is assigned a vertex subset $X_t \subseteq V(G)$, called a bag, such that the following three conditions hold:

- (1) $\bigcup_{t \in V(T)} X_t = V(G)$. In other words, every vertex of G is in at least one bag.
- (2) For every $\{u, v\} \in E(G)$, there exists a node t of T such that bag X_t contains both u and v .
- (3) For every $u \in V(G)$, the set $T_u = \{t \in V(T) : u \in X_t\}$, i.e., the set of nodes whose corresponding bags contain u , induces a connected subtree of T .

Definition 2.3. [6] The **width** of tree decomposition $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ equals $\max_{t \in V(T)} |X_t| - 1$, that is, the maximum size of its bag minus 1. The **treewidth** of a graph G , denoted by $tw(G)$, is the minimum possible width of a tree decomposition of G .

Definition 2.4. [13] An **elimination forest** T of a graph $G = (V, E)$ is a rooted forest on the same vertex set V such that, for every edge $\{u, v\} \in E$, one of u and v is an ancestor of the other. The **depth** of T is the maximum number of vertices on a path from a root to a leaf in T . The **tree-depth** $td(G)$ of a graph G is the minimum depth among all possible elimination forests.

3 TECHNICAL CONTRIBUTIONS

Our main technical contributions in this paper are the hardness results. First we show that EXISTS-PSNE is para-NP-hard with respect to the maximum degree of the graph as parameter [Theorem 4.1]. We prove this by exhibiting a non-trivial reduction from (3, B2)-SAT. Next we show that EXISTS-PSNE is W[1]-hard parameterized by treedepth [Theorem 4.3]. We prove this by exhibiting a non-trivial reduction from GENERAL FACTOR. We also show an important reduction from heterogeneous game to fully homogeneous game which allows us to prove that the hardness results for maximum degree, treedepth, diameter hold even for fully homogeneous games [Theorems 4.7 and 4.8].

We complement the hardness result for treedepth by designing a non-trivial dynamic programming based XP algorithm parameterized by treewidth [Theorem 4.11]. Our XP algorithm also yields a fixed-parameter tractability for the combined parameter “treewidth+maximum degree”.

Lastly, using some standard techniques, we bridge the gap between tractability and intractability by showing (i) W[2]-hardness for the parameters- the number of participating (who play 1) and non-participating players (who play 0) [Theorems 4.5 and 4.6], (ii) fixed-parameter tractability for parameters like vertex-cover number [for strict games], circuit rank and distance from complete graphs [Theorems 4.12, 4.14 and 4.15] and (iii) existence of PSNE in Fully homogeneous games for important classes of graphs like path, complete graph, cycle, and bi-clique [Theorem 4.16].

4 RESULTS

We begin with presenting our results for EXISTS-PSNE. In the interest of space, we omit some proofs; they are marked \star . They are available in the full version [19].

4.1 Hardness Results

The EXISTS-PSNE problem is already known to be NP-complete [29]. We strengthen this result significantly in Theorem 4.1 by proving para-NP-hardness by the maximum degree and the number of different utility functions. We use the NP-complete problem (3, B2)-SAT to prove some of our hardness results [1]. The (3, B2)-SAT problem is the 3-SAT problem restricted to formulas in which each clause contains exactly three literals, and each variable occurs exactly twice positively and twice negatively.

THEOREM 4.1. EXISTS-PSNE is NP-complete for sub-additive strict BNPG games even if the underlying graph is 3-regular and the number of different utility functions is 2. In particular, EXISTS-PSNE parameterized by (maximum degree Δ , the number of different utility functions) is para-NP-hard even for sub-additive strict BNPG games.

PROOF. The EXISTS-PSNE problem clearly belongs to NP. To show its NP-hardness, we reduce from the (3, B2)-SAT problem. The

high-level idea of our proof is as follows. For every clause in (3, B2)-SAT instance, we create a vertex in the EXISTS-PSNE instance. Also, for every literal we create a vertex in the EXISTS-PSNE instance. We then add the set of edges and define the best-response functions in such a way that all the clause vertices play 1 in any PSNE and a set of literal vertices play 1 in a PSNE if and only if there is a satisfying assignment where the same set of literal vertices is assigned TRUE. We now present our construction formally.

Let $(\mathcal{X} = \{x_i : i \in [n]\}, \mathcal{C} = \{C_j : j \in [m]\})$ be an arbitrary instance of (3, B2)-SAT. We define a function $f : \{x_i, \bar{x}_i : i \in [n]\} \rightarrow \{a_i, \bar{a}_i : i \in [n]\}$ as $f(x_i) = a_i$ and $f(\bar{x}_i) = \bar{a}_i$ for $i \in [n]$ and consider the following instance $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ of EXISTS-PSNE.

$$\mathcal{V} = \{a_i, \bar{a}_i : i \in [n]\} \cup \{y_j : j \in [m]\}$$

$$\mathcal{E} = \{\{y_j, f(l_1^j)\}, \{y_j, f(l_2^j)\}, \{y_j, f(l_3^j)\} : C_j = (l_1^j \vee l_2^j \vee l_3^j), j \in [m]\} \cup \{\{a_i, \bar{a}_i\} : i \in [n]\}$$

We observe that the degree of every vertex in \mathcal{G} is 3. We now define $(g_v)_{v \in \mathcal{V}}$ and $(c_v)_{v \in \mathcal{V}}$. $\forall j \in [m]$, $c_{y_j} = 4$, $g_{y_j}(0) = 1000$, $g_{y_j}(1) = 1003$, $g_{y_j}(2) = 1008$, $g_{y_j}(3) = 1013$, $g_{y_j}(4) = 1018$. $\forall i \in [n]$, $c_{a_i} = c_{\bar{a}_i} = 4$, $g_{a_i}(0) = g_{\bar{a}_i}(0) = 1000$, $g_{a_i}(1) = g_{\bar{a}_i}(1) = 1005$, $g_{a_i}(2) = g_{\bar{a}_i}(2) = 1010$, $g_{a_i}(3) = g_{\bar{a}_i}(3) = 1015$, $g_{a_i}(4) = g_{\bar{a}_i}(4) = 1018$.

It follows from the definition that both the above functions are sub-additive. Also, one can easily verify that the above functions give the following best-response functions for the players.

$$\forall i \in [n], \beta_{a_i}(k) = \beta_{\bar{a}_i}(k) = \begin{cases} 1 & \text{if } k \leq 2 \\ 0 & \text{otherwise} \end{cases}$$

$$\forall j \in [m], \beta_{y_j}(k) = \begin{cases} 0 & \text{if } k = 0 \\ 1 & \text{otherwise} \end{cases}$$

From the best-response functions, it follows that the game is strict. We now claim that the above BNPG game has a PSNE if and only if the (3, B2)-SAT instance is a YES instance.

For the “if” part, suppose the (3, B2)-SAT instance is a YES instance. Let $h : \{x_i : i \in [n]\} \rightarrow \{\text{TRUE}, \text{FALSE}\}$ be a satisfying assignment of the (3, B2)-SAT instance. We consider the following strategy profile for the BNPG game.

- $\forall j \in [m], s(y_j) = 1$
- $\forall i \in [n], s(a_i) = 1$ if and only if $h(x_i) = \text{TRUE}$
- $\forall i \in [n], s(\bar{a}_i) = 0$ if and only if $h(x_i) = \text{TRUE}$

We observe that, since h is a satisfying assignment, the player y_j for every $j \in [m]$ has at least one neighbor who plays 1 and thus y_j does not have any incentive to deviate (from playing 1). For $i \in [n]$ such that $h(x_i) = \text{TRUE}$, the player a_i has at least one neighbor, namely \bar{a}_i , who plays 0 and thus a_i does not have any incentive to deviate (from playing 1); on the other hand the player \bar{a}_i has all her neighbor playing 1, and thus she is happy to play 0. Similarly, for $i \in [n]$ such that $h(x_i) = \text{FALSE}$, both the players a_i and \bar{a}_i have no incentive to deviate. This proves that the above strategy profile is a PSNE.

For the “only if” part, let $(s(a_i)_{i \in [n]}, s(\bar{a}_i)_{i \in [n]}, s(y_j)_{j \in [m]})$ be a PSNE for the BNPG game. We claim that $s(y_j) = 1$ for every $j \in [m]$. Suppose not, then there exists a $t \in [m]$ such that $s(y_t) = 0$. Let the literals in clause C_t be l_1^t, l_2^t, l_3^t . Then $s(f(l_i^t)) = 0, \forall i \in [3]$

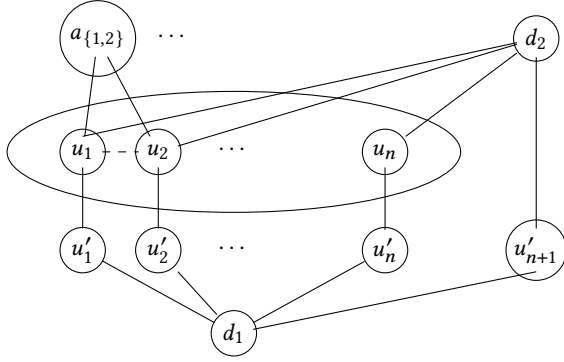


Figure 1: Graph \mathcal{H} in the proof of Theorem 4.3.

otherwise the player y_t will deviate from 0 and play 1. But then the player $f(l_1^t)$ will deviate to 1 as y_t plays 0 which is a contradiction. We now claim that we have $s(a_i) \neq s(\bar{a}_i)$ for every $i \in [n]$. Suppose not, then there exists a $\lambda \in [n]$ such that $s(a_\lambda) = s(\bar{a}_\lambda)$. If $s(a_\lambda) = s(\bar{a}_\lambda) = 1$, then both the players a_λ and \bar{a}_λ have incentive to deviate to 0. On the other hand, if $s(a_\lambda) = s(\bar{a}_\lambda) = 0$, then both the players a_λ and \bar{a}_λ have incentive to deviate to 1. This proves the claim. We now consider the assignment $h : \{x_i : i \in [n]\} \rightarrow \{\text{TRUE}, \text{FALSE}\}$ defined as $h(x_i) = \text{TRUE}$ if and only if $s(a_i) = 1$ for every $i \in [n]$. We claim that h is a satisfying assignment for the (3, B2)-SAT formula. Suppose not, then h does not satisfy a clause, say $C_\gamma, \gamma \in [m]$. Then the player y_γ has incentive to deviate to 0 as none of its neighbors play 1 which is a contradiction. \square

For the remainder of this subsection, we describe a game using the best response functions for the sake of simplicity of presentation. This suffices as due to Lemma 2.1, we can always compute the utility functions using the best response functions in polynomial time.

We next consider treedepth as parameter. Problems on graphs which are easy for trees are often fixed-parameter-tractable with respect to treedepth as parameter. We show that this is not the case for our problem. Towards that, we use the GENERAL FACTOR problem which is $W[1]$ -hard parameterized by treedepth [23].

Definition 4.2 (GENERAL FACTOR). Given a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ and a set $K(v) \subseteq \{0, \dots, d(v)\}$ for each $v \in \mathcal{V}$, compute if there exists a subset $\mathcal{F} \subseteq \mathcal{E}$ such that, for each vertex $v \in \mathcal{V}$, the number of edges in \mathcal{F} incident on v is an element of $K(v)$. We denote an arbitrary instance of this problem by $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (K(v))_{v \in \mathcal{V}})$.

THEOREM 4.3. EXISTS-PSNE for BNPG games is $W[1]$ -hard parameterized by treedepth.

PROOF. To prove $W[1]$ -hardness, we reduce from GENERAL FACTOR parameterized by treedepth to BNPG game.

Let $(\mathcal{G} = (\{v_i : i \in [n]\}, \mathcal{E}'), (K(v_i))_{i \in [n]})$ be an arbitrary instance of GENERAL FACTOR. The high level idea of our construction is as follows. For each vertex and edge in the graph \mathcal{G} associated with GENERAL FACTOR instance, we add a node in the graph \mathcal{H} (where the BNPG game is defined) associated with EXISTS-PSNE problem instance. On top of that we add some extra nodes and edges in \mathcal{H} and appropriately define the best response functions

of every player in \mathcal{H} so that a set of nodes in \mathcal{H} corresponding to a set \mathcal{F} of edges belonging to \mathcal{G} play 1 in a PSNE if and only if \mathcal{F} makes GENERAL FACTOR a yes instance. We now formally present our construction.

We consider a BNPG game on the following graph $\mathcal{H} = (\mathcal{V}, \mathcal{E})$. See Figure 1 for a pictorial representation of \mathcal{H} .

$$\mathcal{V} = \{u_i : i \in [n]\} \cup \{a_{\{i,j\}} : \{v_i, v_j\} \in \mathcal{E}'\}$$

$$\cup \{u'_i : i \in [n+1]\} \cup \{d_1, d_2\}$$

$$\mathcal{E} = \{\{u_i, a_{\{i,j\}}\}, \{u_j, a_{\{i,j\}}\} : \{v_i, v_j\} \in \mathcal{E}'\} \cup \{\{u_i, u'_i\} : i \in [n]\}$$

$$\cup \{\{d_1, u'_i\}, \{d_2, u_i\} : i \in [n]\} \cup \{\{d_1, u'_{n+1}\}, \{d_2, u'_{n+1}\}\}$$

Let the treedepth of \mathcal{G} be τ . Create a graph \mathcal{G}' by adding the vertices d_1, d_2 and the set of edges $\{\{d_1, v_i\}, \{d_2, v_i\} : i \in [n]\} \cup \{d_1, d_2\}$ to the graph \mathcal{G} . The treedepth of \mathcal{G}' is at most $\tau + 2$. We claim that the treedepth of \mathcal{H} is at most $\tau + 3$. To see this, we begin with an elimination tree of \mathcal{G}' and replace v_i with u_i for every $i \in [n]$. Let $\mathcal{S} = \{u'_i : i \in [n+1]\} \cup \{a_{\{i,j\}} : \{v_i, v_j\} \in \mathcal{E}'\}$. $\forall u' \in \mathcal{S}$, add an edge between u' and u in the elimination tree where u, v are neighbors of u' in \mathcal{H} and u is descendant of v in the elimination tree. This results in a valid elimination tree for \mathcal{H} and hence, the treedepth of \mathcal{H} is at most $\tau + 3$.

We now describe the best-response functions of the vertices in \mathcal{H} to complete the description of the BNPG game.

$$\forall i \in [n], \beta_{u_i}(k) = \begin{cases} 1 & \text{if } k-1 \in K(v_i) \\ 0 & \text{otherwise} \end{cases}$$

$$\forall i \in [n+1], \beta_{u'_i}(k) = \begin{cases} 1 & \text{if } k=2 \\ 0 & \text{otherwise} \end{cases}$$

$$\forall \{v_i, v_j\} \in \mathcal{E}', \beta_{a_{\{i,j\}}}(k) = \{0, 1\} \forall k \in \mathbb{N} \cup \{0\}$$

$$\beta_{d_1}(k) = \begin{cases} 1 & \text{if } k=0 \text{ or } k=n \\ 0 & \text{otherwise} \end{cases}, \beta_{d_2}(k) = \begin{cases} 1 & \text{if } k=0 \\ 0 & \text{otherwise} \end{cases}$$

We claim that the above BNPG game has a PSNE if and only if the GENERAL FACTOR instance is a YES instance.

For the “if” part, suppose the GENERAL FACTOR instance is a YES instance. Then there exists a subset $\mathcal{F} \subseteq \mathcal{E}'$ such that for all $i \in [n]$, the degree of v_i in $\mathcal{G}[\mathcal{F}]$ is an element of the set $K(v_i)$. We consider the strategy profile $\bar{x} = (x_v)_{v \in \mathcal{V}}$.

$$\forall i \in [n], x_{u_i} = x_{u'_i} = 1, x_{u'_{n+1}} = 0$$

$$\forall \{v_i, v_j\} \in \mathcal{E}', x_{a_{\{i,j\}}} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in \mathcal{F} \\ 0 & \text{otherwise} \end{cases}, x_{d_1} = 1, x_{d_2} = 0$$

Now we argue that \bar{x} is a PSNE for the BNPG game. Clearly no player $a_{\{i,j\}}, \{v_i, v_j\} \in \mathcal{E}'$ deviates as both 0 and 1 are her best-responses irrespective of the action of their neighbors. The player d_1 does not deviate as she has exactly n neighbors playing 1. The player $u'_i, i \in [n]$ does not deviate as she has exactly 2 neighbors playing 1. The player u'_{n+1} does not deviate as she has exactly 1 neighbor playing 1. The player d_2 does not deviate as she has at least 1 neighbors playing 1. Note that $\forall i \in [n]$, the number of neighbors of u_i playing 1 excluding u'_i and d_2 (which in this case is $n_{u_i} - 1$ as $x_{d_2} = 0, x_{u'_i} = 1$) is the same as the number of edges in \mathcal{F} which are incident on v_i in \mathcal{G} . Hence, $\forall i \in [n]$, the player u_i does not deviate as $(n_{u_i} - 1) \in K(v_i)$. Hence, \bar{x} is a PSNE.

For the “only if” part, let $\bar{x} = (x_v)_{v \in \mathcal{V}}$ be a PSNE of the BNPG game. We claim that we have $x_{d_1} = 1, x_{u_i} = x_{u'_i} = 1, \forall i \in [n], x_{u'_{n+1}} = 0, x_{d_2} = 0$. To prove this, we consider all cases for $(x_{u_i})_{i \in [n]}$.

- (1) Case $\forall i \in [n] x_{u_i} = 1$: We have $x_{d_2} = 0$ as $n_{d_2} > 0$ otherwise d_2 would deviate. This implies that $x_{u'_{n+1}} = 0$ since $n_{u'_{n+1}} \leq 1$ (as $x_{d_2} = 0$). Now we consider the following sub-cases (according to the values of x_{d_1} and $x_{u'_i}, i \in [n]$):
 - $(x_{d_1} = 1, \exists k \in [n]$ such that $x_{u'_k} = 0$). Here $x_{u'_k}$ will then deviate to 1 as $n_{u'_k} = 2$. Hence, it is not a PSNE.
 - $(x_{d_1} = 1, \forall i \in [n] x_{u'_i} = 1)$. This is exactly what we claim thus we have nothing to prove in this case.
 - $(x_{d_1} = 0, \exists k \in [n]$ such that $x_{u'_k} = 1)$. Here $x_{u'_k}$ will then deviate to 0 as $n_{u'_k} = 1$. Hence, it is not a PSNE.
 - $(x_{d_1} = 0, \forall i \in [n] x_{u'_i} = 0)$. The player d_1 will deviate to 1 as $n_{d_1} = 0$. Hence, it is not a PSNE.
- (2) Case $\exists k_1, k_2 \in [n]$ such that $x_{u_{k_1}} = 1$ and $x_{u_{k_2}} = 0$: We have $x_{d_2} = 0$ as $n_{d_2} > 0$ otherwise d_2 would deviate. This implies that $x_{u'_{n+1}} = 0$ since $n_{u'_{n+1}} \leq 1$ (as $x_{d_2} = 0$). Now we consider the following sub-cases (according to the values of x_{d_1} and $x_{u'_i}, i \in [n]$):
 - $(x_{d_1} = 1, \forall i \in [n] x_{u'_i} = 0)$. Here u'_{k_1} will deviate to 1 as $n_{u'_{k_1}} = 2$. So, it isn't a PSNE.
 - $(x_{d_1} = 1, \forall i \in [n] x_{u'_i} = 1)$. Here u'_{k_2} will deviate to 0 as $n_{u'_{k_2}} = 1$. So, it isn't a PSNE.
 - $(x_{d_1} = 1, \exists i, j \in [n]$ such that $x_{u'_i} = 1$ and $x_{u'_j} = 0$). Here d_1 will deviate to 0 as $0 < n_{d_1} < n$ (there are at least 2 neighbours of d_1 which play 0 and at least 1 neighbour of d_1 which plays 1). Hence, it is not a PSNE.
 - $(x_{d_1} = 0, \forall i \in [n] x_{u'_i} = 0)$. Here d_1 will deviate to 1 as $n_{d_1} = 0$. So, it isn't a PSNE.
 - $(x_{d_1} = 0, \exists i \in [n]$ such that $x_{u'_i} = 1)$. Here u'_i will deviate to 0 as $n_{u'_i} \leq 1$ and hence, it is not a PSNE.
- (3) Case $\forall i \in [n] x_{u_i} = 0$: For every $i \in [n]$, we must have $x_{u'_i} = 0$ so that u'_i doesn't deviate. We have the following sub-cases (according to the values of x_{d_1}, x_{d_2} and $x_{u'_{n+1}}$):
 - $(x_{d_1} = 0, x_{u'_{n+1}} = 0)$. Here d_1 deviates to 1 as $n_{d_1} = 0$ and hence, it is not a PSNE.
 - $(x_{d_1} = 0, x_{u'_{n+1}} = 1)$. Here u'_{n+1} deviates to 0 as $n_{u'_{n+1}} \leq 1$. So, it isn't a PSNE.
 - $(x_{d_1} = 1, x_{u'_{n+1}} = 0, x_{d_2} = 0)$. Here d_2 deviates to 1 as $n_{d_2} = 0$. So, it isn't a PSNE.
 - $(x_{d_1} = 1, x_{u'_{n+1}} = 0, x_{d_2} = 1)$. Here u'_{n+1} deviates to 1 as $n_{u'_{n+1}} = 2$ and hence, it is not a PSNE.
 - $(x_{d_1} = 1, x_{u'_{n+1}} = 1, x_{d_2} = 0)$. Here u'_{n+1} deviates to 0 as $n_{u'_{n+1}} = 1$ and hence, it is not a PSNE.
 - $(x_{d_1} = 1, x_{u'_{n+1}} = 1, x_{d_2} = 1)$. Here d_2 deviates to 0 as $n_{d_2} > 0$. So, it isn't a PSNE.

So if $\bar{x} = (x_v)_{v \in \mathcal{V}}$ is a PSNE of the BNPG game, then we have $x_{d_1} = 1, \forall i \in [n], x_{u_i} = 1, x_{u'_{n+1}} = 0, \forall i \in [n] x_{u_i} = 1, x_{d_2} = 0$. Now consider the set $\mathcal{F} = \{\{v_i, v_j\} : x_{a_{(i,j)}} = 1, \{v_i, v_j\} \in \mathcal{E}'\}$. Note that $\forall i \in [n]$, the number of neighbors of u_i playing 1 excluding u'_i and d_2 (which in this case is $n_{u_i} - 1$ as $x_{d_2} = 0, x_{u'_i} = 1$) is the

same as the number of edges in \mathcal{F} which are incident on v_i in \mathcal{G} . Since $\forall i, n_{u_i} - 1 \in K(v_i)$, the number of edges in \mathcal{F} incident on v_i in GENERAL FACTOR instance is an element of $K(v_i)$. Hence, the GENERAL FACTOR instance is a YES instance. \square

COROLLARY 4.4. *EXISTS-PSNE for BNPG games is W[1]-hard parameterized by treewidth and pathwidth.*

We next consider the diameter (d) of the graph as our parameter and prove para-NP-hardness in Observation 1. It follows immediately from the fact that the reduced instance in the NP-completeness proof of EXISTS-PSNE for BNPG games in [29] has diameter 2.

OBSERVATION 1. *EXISTS-PSNE for BNPG games is NP-complete even for graphs of diameter at most 2. In particular, the EXISTS-PSNE problem for BNPG games is para-NP-hard parameterized by diameter.*

We next consider a variant of EXISTS-PSNE where at most k_0 (respectively k_1) players are playing 0 (respectively 1) in the PSNE. We denote this variant as k_0 -EXISTS-PSNE (resp. k_1 -EXISTS-PSNE). Obviously there is a brute force XP algorithm which runs in time $O^*(n^{k_0})$ (respectively $O^*(n^{k_1})$). We show that k_0 -EXISTS-PSNE (resp. k_1 -EXISTS-PSNE) is W[2]-hard parameterized by k_0 (respectively k_1). For this, we reduce from the DOMINATING SET problem parameterized by the size of dominating set which is known to be W[2]-hard [6].

THEOREM 4.5 (★). *k_0 -EXISTS-PSNE for BNPG games is W[2]-hard parameterized by k_0 .*

THEOREM 4.6 (★). *k_1 -EXISTS-PSNE for BNPG games is W[2]-hard parameterized by k_1 even for fully homogeneous BNPG games.*

Till now we have mostly focused on heterogeneous BNPG games. We next consider fully homogeneous BNPG games and show the following by reducing from the EXISTS-PSNE problem on heterogeneous BNPG games.

THEOREM 4.7. *The following results hold even for fully homogeneous games.*

- (1) EXISTS-PSNE is NP-complete even if the diameter of the graph is at most 4.
- (2) EXISTS-PSNE is W[1]-hard with respect to the parameter treedepth of the graph.
- (3) k_0 -EXISTS-PSNE is W[2]-hard parameterized by k_0 .

PROOF. We first present a reduction from the EXISTS-PSNE problem on heterogeneous BNPG games to the EXISTS-PSNE problem on fully homogeneous BNPG games. Let $(\mathcal{G} = (\mathcal{V} = \{v_i : i \in [n]\}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ be any heterogeneous BNPG game. We now construct the graph $\mathcal{H} = (\mathcal{V}', \mathcal{E}')$ for the instance of the fully homogeneous BNPG game.

$$\mathcal{V}' = \{u_i : i \in [n]\} \cup \bigcup_{i \in [n]} \mathcal{V}_i,$$

$$\text{where } \mathcal{V}_i = \{a_j^i : j \in [2 + (i - 1)n]\}, \forall i \in [n]$$

$$\mathcal{E}' = \{\{u_i, u_j\} : \{v_i, v_j\} \in \mathcal{E}\} \cup \bigcup_{i \in [n]} \mathcal{E}_i,$$

$$\text{where } \mathcal{E}_i = \{\{a_j^i, u_i\} : j \in [2 + (i - 1)n]\}, \forall i \in [n]$$

Let us define $f(x) = \lfloor \frac{x-2}{n} \rfloor + 1$, $h(x) = x - 2 - (f(x) - 1)n$. We now define best-response strategies β for the fully homogeneous BNPG game on \mathcal{H} .

$$\beta(k) = \begin{cases} 1 & \text{if } k = 0 \text{ or } k = 1 \\ \{0, 1\} & \text{if } \Delta g_{v_{f(k)}}(h(k)) = c_{v_{f(k)}}, k > 1 \\ 1 & \text{if } \Delta g_{v_{f(k)}}(h(k)) > c_{v_{f(k)}}, k > 1 \\ 0 & \text{if } \Delta g_{v_{f(k)}}(h(k)) < c_{v_{f(k)}}, k > 1 \end{cases}$$

This finishes description of our fully homogeneous BNPG game on \mathcal{H} . We now claim that there exists a PSNE in the heterogeneous BNPG game on \mathcal{G} if and only if there exists a PSNE in the fully homogeneous BNPG game on \mathcal{H} .

For the “only if” part, let $x^* = (x_v^*)_{v \in \mathcal{V}}$ be a PSNE in the heterogeneous BNPG game on \mathcal{G} . We now consider the following strategy profile $\bar{y} = (y_v)_{v \in \mathcal{V}}$ for players in \mathcal{H} .

$$\forall i \in [n] y_{u_i} = x_{v_i}^*; y_w = 1 \text{ for other vertices } w$$

Clearly the players in $\cup_{i \in [n]} \mathcal{V}_i$ do not deviate as their degree is 1 and $\beta(0) = \beta(1) = 1$. In \bar{y} , we have $n_{u_i} = n_{v_i} + 2 + (i - 1)n \geq 2$ and $n_{v_i} \leq n - 1$ for every $i \in [n]$. If $x_{v_i}^* = 1$, then we have $\Delta g_{v_i}(n_{v_i}) \geq c_{v_i}$. We have $f(n_{u_i}) = i$ and $h(n_{u_i}) = n_{v_i}$. This implies that $\Delta g_{v_{f(n_{u_i})}}(h(n_{u_i})) \geq c_{v_{f(n_{u_i})}}$. So u_i does not deviate as 1 is the best-response. If $x_{v_i}^* = 0$, then we have $\Delta g_{v_i}(n_{v_i}) \leq c_{v_i}$. This implies that $\Delta g_{v_{f(n_{u_i})}}(h(n_{u_i})) \leq c_{v_{f(n_{u_i})}}$. So u_i does not deviate as 0 is the best-response. Hence, \bar{y} is a PSNE.

For the “if” part, suppose there exists a PSNE $(x_v^*)_{v \in \mathcal{V}}$ in the fully homogeneous BNPG game on \mathcal{H} . Clearly $x_v^* = 1$ for all $v \in \cup_{i \in [n]} \mathcal{V}_i$ as $n_v \leq 1$. Now we claim that the strategy profile $\bar{x} = (x_{v_i} = x_{u_i}^*)_{i \in [n]}$ forms a PSNE for the heterogeneous BNPG game on \mathcal{G} . We observe that if $x_{u_i}^* = 1$, then $\Delta g_{v_{f(n_{u_i})}}(h(n_{u_i})) \geq c_{v_{f(n_{u_i})}}$ for $i \in [n]$. This implies that $\Delta g_{v_i}(n_{v_i}) \geq c_{v_i}$. So $x_{v_i} = 1$ is the best-response for v_i in \mathcal{G} and hence, she does not deviate. Similarly, if $x_{u_i}^* = 0$, then $\Delta g_{v_{f(n_{u_i})}}(h(n_{u_i})) \leq c_{v_{f(n_{u_i})}}$. This implies that $\Delta g_{v_i}(n_{v_i}) \leq c_{v_i}$. So $x_{v_i} = 0$ is the best-response for $v_i \in \mathcal{V}$ and hence, it won't deviate. Hence, \bar{x} is a PSNE in the heterogeneous BNPG game on \mathcal{G} .

We now prove the three statements in the theorem as follows.

- (1) We observe that, if the diameter of \mathcal{G} is at most 2, then the diameter of \mathcal{H} is at most 4. Hence, the result follows from Observation 1.
- (2) We observe that the treedepth of \mathcal{H} is at most 1 more than the treedepth of \mathcal{G} . Hence, the result follows from Theorem 4.3.
- (3) We observe that there exists a PSNE where at most k players play 0 in the heterogeneous BNPG game on \mathcal{G} if and only if there exists a PSNE where at most k players play 0 in the fully homogeneous BNPG game on \mathcal{H} . Hence, the result follows from Theorem 4.5. \square

We next show that EXISTS-PSNE for fully homogeneous BNPG games is para-NP-hard parameterized by the maximum degree of the graph again by reducing from heterogeneous BNPG games.

THEOREM 4.8 (★). EXISTS-PSNE for fully homogeneous BNPG games is NP-complete even if the maximum degree Δ of the graph is at most 9.

4.2 XP Algorithm for the parameter treewidth

Our next result is an XP algorithm for the EXISTS-PSNE problem when parameterized by treewidth. Towards that, we introduce the notion of “feasible function” in Definition 4.9 and prove a related algorithmic result in Lemma 4.10.

Definition 4.9. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with maximum degree Δ . Let $f : V \rightarrow [\Delta] \cup \{0\}$ be a function where $V \subseteq \mathcal{V}$. We call a function f feasible if there exists a strategy profile S of all the players in \mathcal{G} such that for each $u \in V$, number of neighbours of u playing 1 in the strategy profile S is $f(u)$.

LEMMA 4.10 (★). Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with maximum degree Δ . Let $V \subseteq \mathcal{V}$. Then the set of all feasible functions $f : V \rightarrow [\Delta] \cup \{0\}$ can be computed in time $O^*(\Delta^{|V|})$.

We now present a $O^*(\Delta^{O(k)})$ time XP algorithm for EXISTS-PSNE where k is the treewidth of the input graph. Note that the running time of $O^*(\Delta^{O(k)})$ implies that EXISTS-PSNE is fixed-parameter tractable for the combined parameter “treewidth+maximum degree”.

THEOREM 4.11. Let \mathcal{G} be an n -vertex graph given together with its tree decomposition of treewidth at most k . Then there is an algorithm running in time $O^*(\Delta^{O(k)})$ for EXISTS-PSNE in BNPG game on \mathcal{G} where Δ is the maximum degree of graph \mathcal{G} .

PROOF SKETCH. Let $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ be any instance of EXISTS-PSNE for BNPG games. Let $(\beta_v(\cdot))_{v \in \mathcal{V}}$ be the set of the best response functions. Let $\mathcal{T} = (T, \{X_t\}_{t \in V(T)})$ be a nice tree decomposition of the input n -vertex graph \mathcal{G} that has width at most k . Let \mathcal{T} be rooted at some node r . For a node t of \mathcal{T} , let V_t be the union of all the bags present in the subtree of \mathcal{T} rooted at t , including X_t . We solve the EXISTS-PSNE problem using dynamic programming. Let $N_1(X_t)$ denote set of vertices in $\mathcal{V} \setminus V_t$ which is adjacent to at least one vertex in X_t . Let $N_2(X_t)$ denote set of vertices in $V_t \setminus X_t$ which is adjacent to at least one vertex in X_t . Let $c[t, (x_v)_{v \in X_t}, (d_v^1)_{v \in X_t}, (d_v^2)_{v \in X_t}] = 1$ (resp. 0) denote that there exists (resp. doesn't exist) a strategy profile S of all the players in \mathcal{G} such that for each $u \in X_t$, u plays x_u , number of neighbours of u in $N_1(X_t)$ (resp. $N_2(X_t)$) playing 1 is d_u^1 (resp. d_u^2) and none of the vertices in V_t deviate in the strategy profile S . Before we proceed, we would like to introduce some notations. Let V be a set of vertices and $S_1 = (x_v)_{v \in V}$, $S_2 = (x_v)_{v \in V \setminus \{w\}}$ be two tuples. Then $S_1 \setminus \{x_w\} := S_2$ and $S_2 \cup \{x_w\} := S_1$. Also, we denote an empty tuple by ϕ . Clearly $c[r, \phi, \phi, \phi]$ indicates whether there is a PSNE in \mathcal{G} or not. We now present the recursive equation to compute $c[t, (x_v)_{v \in X_t}, (d_v^1)_{v \in X_t}, (d_v^2)_{v \in X_t}]$ for various types of node in \mathcal{T} .

Leaf Node: For a leaf node t we have that $X_t = \phi$. Hence, $c[t, \phi, \phi, \phi] = 1$.

Join Node: For a join node t , let t_1, t_2 be its two children. Note that $X_t = X_{t_1} = X_{t_2}$.

Now we proceed to compute $c[t, (x_v)_{v \in X_t}, (d_v^1)_{v \in X_t}, (d_v^2)_{v \in X_t}]$. Let \mathcal{F} be a set of tuples $(d'_v)_{v \in X_t}$ such that there is a strategy profile S such that for each $v \in X_t$, its response is x_v , the number of neighbours in $N_1(x)$, $V_{t_1} \setminus X_{t_1}$ and $V_{t_2} \setminus X_{t_2}$ playing 1 is $d_v^1, d'_v, d_v^2 - d'_v$ respectively. Using Lemma 4.10 we can find the set \mathcal{F} in time $O^*(\Delta^k)$. Then $c[t, (x_v)_{v \in X_t}, (d_v^1)_{v \in X_t}, (d_v^2)_{v \in X_t}]$ is equal to the

following formula:

$$0 \vee \bigvee_{(d'_v)_{v \in X_t} \in \mathcal{F}} (c[t_1, (x_v)_{v \in X_t}, (d'_v + d''_v - d'_v)_{v \in X_t}, (d'_v)_{v \in X_t}]) \\ \wedge c[t_2, (x_v)_{v \in X_t}, (d'_v + d''_v)_{v \in X_t}, (d''_v - d'_v)_{v \in X_t}]$$

Introduce Node: Let t be an introduce node with a child t' such that $X_t = X_{t'} \cup \{u\}$ for some $u \notin X_{t'}$. Let $S' = (x_v)_{v \in X_t}$ be a strategy profile of vertices in X_t . Let n'_v denote the number of neighbours of v playing 1 in S' . Let $g : \mathcal{V} \times \mathcal{V} \rightarrow \{0, 1\}$ be a function such that $g(\{u, v\}) = 1$ if and only if $\{u, v\} \in \mathcal{E}$. We now proceed to compute $c[t, S', (d''_v)_{v \in X_t}, (d''_v)_{v \in X_t}]$. If there is no strategy profile S where $\forall v \in X_t$, the number of neighbours of v in $N_1(X_t)$ (resp. $N_2(X_t)$) playing 1 is d''_v (resp. d''_v), then clearly $c[t, S', (d''_v)_{v \in X_t}, (d''_v)_{v \in X_t}] = 0$. Due to Lemma 4.10, we can check the previous statement in $O^*(\Delta^k)$ by considering a bipartite subgraph of \mathcal{G} between X_t and $N_1(X_t)$ (or $N_2(X_t)$). Otherwise, we have the following:

$$c[t, S', (d''_v)_{v \in X_t}, (d''_v)_{v \in X_t}] = \begin{cases} 0 & \text{if } \exists v \in X_t, x_v \notin \beta_v(n'_v + d''_v + d''_v) \\ c[t', S' \setminus \{x_u\}, (d''_v + g(\{u, v\}))_{v \in X_{t'}}, (d''_v)_{v \in X_{t'}}] & \text{if } x_u = 1 \\ c[t', S' \setminus \{x_u\}, (d''_v)_{v \in X_{t'}}, (d''_v)_{v \in X_{t'}}] & \text{otherwise} \end{cases}$$

Forget Node: Let t be a forget node with a child t' such that $X_t = X_{t'} \setminus \{w\}$ for some $w \in X_{t'}$. Let $S_0 = (x_v)_{v \in X_t} \cup \{x_w = 0\}$, $S_1 = (x_v)_{v \in X_t} \cup \{x_w = 1\}$ be two strategy profiles of vertices in $X_{t'}$. Let $g : \mathcal{V} \times \mathcal{V} \rightarrow \{0, 1\}$ be a function such that $g(\{u, v\}) = 1$ if and only if $\{u, v\} \in \mathcal{E}$. We now have the following:

$$c[t, (x_v)_{v \in X_t}, (d''_v)_{v \in X_t}, (d''_v)_{v \in X_t}] = \bigvee_{\substack{d''_w, d''_v: 0 \leq d''_w, d''_v \leq \Delta}} (c[t', S_0, (d''_v)_{v \in X_{t'}}, (d''_v)_{v \in X_{t'}}] \\ \vee c[t', S_1, (d''_v)_{v \in X_{t'}}, (d''_v - g(\{v, w\}))_{v \in X_{t'}}])$$

Due to space constraints, we refer the reader to the full version of our paper for the proof of correctness of the above recursive equations. Now we consider the time complexity of our algorithm. Total number of cells in the dynamic programming table which we created is $O^*(\Delta^{O(k)})$. For each cell, we spend at most $O^*(\Delta^{O(k)})$ time if we are computing the table in a bottom up fashion. Hence, the running time is $O^*(\Delta^{O(k)})$. \square

4.3 Tractable Results

To conclude our fine-grained analysis of the EXISTS-PSNE problem, we bridge the gap between the tractability and intractability by showing some tractable results. Our first result is an FPT algorithm for EXISTS-PSNE for strict games when parameterized by the vertex cover number.

THEOREM 4.12 (★). *There is a $O^*(2^{vc(\mathcal{G})})$ time algorithm for EXISTS-PSNE for strict BNPG games where $vc(\mathcal{G})$ is the vertex cover number.*

Our next result shows that we can always find a PSNE for additive BNPG games in $O(n)$ time. This complements the intractable result for subadditive BNPG games.

OBSERVATION 2 (★). *There exists an $O(n)$ time algorithm to find a PSNE in an additive BNPG game.*

We next consider circuit rank and distance from complete graph as parameter. These parameters can be thought of distance from tractable instances (namely tree and complete graph). They are defined as follows.

Definition 4.13. Let the number of edges and number of vertices in a graph \mathcal{G} be m and n respectively. Then d_1 (circuit rank) is defined to be $m - n + c$ (c is the number of connected components in the graph) and d_2 (distance from complete graph) is defined to be $\frac{n(n-1)}{2} - m$. Note that circuit rank is not the same as feedback arc set.

Yu et al. presented an algorithm for EXISTS-PSNE on trees in [29]. It turns out that their algorithm can be appropriately modified to get the following observation.

OBSERVATION 3. [29] *Given a BNPG game on a tree $\mathcal{T} = (\mathcal{V}, \mathcal{E})$, a subset of vertices $\mathcal{U} \subseteq \mathcal{V}$ and a strategy profile $(x_u)_{u \in \mathcal{U}} \in \{0, 1\}^{\mathcal{U}}$, there is a polynomial time algorithm for deciding if there exists a PSNE $(y_v)_{v \in \mathcal{V}} \in \{0, 1\}^{\mathcal{V}}$ for the BNPG game such that $x_u = y_u$ for every $u \in \mathcal{U}$.*

Now by using the observation 3 as a subroutine, we exhibit an FPT algorithm for the parameter circuit rank.

THEOREM 4.14. *There is an algorithm running in time $O^*(4^{d_1})$ for EXISTS-PSNE in BNPG games where d_1 is the circuit rank of the input graph.*

PROOF. Let $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ be any instance of EXISTS-PSNE for BNPG games. Let the graph \mathcal{G} have c connected components namely, $\mathcal{G}_1 = (\mathcal{V}_1, \mathcal{E}_1), \dots, \mathcal{G}_c = (\mathcal{V}_c, \mathcal{E}_c)$. For every $i \in [c]$, we decide if there exists a PSNE in \mathcal{G}_i ; clearly there is a PSNE in \mathcal{G} if and only if there is a PSNE in \mathcal{G}_i for every $i \in [c]$. Hence, in the rest of the proof, we focus on the algorithm to decide the existence of a PSNE in \mathcal{G}_i . We compute a minimum spanning tree \mathcal{T}_i in the connected component \mathcal{G}_i . Let $\mathcal{E}'_i \subseteq \mathcal{E}_i$ be the set of edges which are not part of \mathcal{T}_i ; let $|\mathcal{E}'_i| = d'_i$ and $\mathcal{V}'_i = \{v^1_i, v^2_i, \dots, v^{\ell}_i\} \subseteq \mathcal{V}_i$ be the set of vertices which are endpoints of at least one edge in \mathcal{E}'_i . Of course, we have $|\mathcal{V}'_i| = \ell \leq 2d'_i$. For every tuple $t = (x'_v)_{v \in \mathcal{V}'_i} \in \{0, 1\}^{\ell}$, we do the following.

- (1) For each $v \in \mathcal{V}'_i$, let n'_v be the number of neighbours of v in $\mathcal{G}_i[\mathcal{E}'_i]$ (subgraph of \mathcal{G}_i containing the set of nodes \mathcal{V}'_i and the set of edges \mathcal{E}'_i) who play 1 in t . We now define g'_v for every player $v \in \mathcal{V}'_i$ as follows.

$$g'_v(k) = \begin{cases} g_v(k + n'_v) & \text{if } v \in \mathcal{V}'_i \\ g_v(k) & \text{otherwise} \end{cases}$$

- (2) We now decide if there exists a PSNE $(y_v)_{v \in \mathcal{V}'_i} \in \{0, 1\}^{\mathcal{V}'_i}$ in the BNPG game $(\mathcal{T}_i, (g'_v)_{v \in \mathcal{V}'_i}, (c_v)_{v \in \mathcal{V}'_i})$ such that $y_v = x'_v$ for every $v \in \mathcal{V}'_i$; this can be done in polynomial time due to Observation 3. If such a PSNE exists, then we output YES.

If we fail to find a PSNE for every choice of tuple t , then we output NO. The running time of the above algorithm (for \mathcal{G}_i) is $O^*(2^{|\mathcal{V}'_i|})$. Hence the overall running time of our algorithm is $O^*(\sum_{i=1}^c 2^{|\mathcal{V}'_i|}) \leq O^*(2^{2d_1}) = O^*(4^{d_1})$. We now argue

correctness of our algorithm. We observe that it is enough to argue correctness for one component.

In one direction, let $x^* = (x_v^*)_{v \in \mathcal{V}_i}$ be a PSNE in the BNPG game $(\mathcal{G}_i, (g_v)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$. We now claim that $(x_v^*)_{v \in \mathcal{V}_i}$ is also a PSNE in the BNPG game on $(\mathcal{T}_i, (g_v^t)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$ where $t = (x_v^*)_{v \in \mathcal{V}_i}$. Let $n_v^{\mathcal{G}_i}$ and $n_v^{\mathcal{T}_i}$ be the number of neighbors of $v \in \mathcal{V}_i$ in \mathcal{G}_i and \mathcal{T}_i respectively who play 1 in x^* . With n_v^t defined as above, we have $n_v^{\mathcal{G}_i} = n_v^{\mathcal{T}_i} + n_v^t$ for $v \in \mathcal{V}_i'$ and $n_v^{\mathcal{G}_i} = n_v^{\mathcal{T}_i}$ for $v \in \mathcal{V}_i \setminus \mathcal{V}_i'$. Hence, we have $\Delta g_v^t(n_v^{\mathcal{T}_i}) = \Delta g_v(n_v^{\mathcal{T}_i} + n_v^t) = \Delta g_v(n_v^{\mathcal{G}_i})$ for $v \in \mathcal{V}_i'$ and $\Delta g_v^t(n_v^{\mathcal{T}_i}) = \Delta g_v(n_v^{\mathcal{G}_i})$ for $v \in \mathcal{V}_i \setminus \mathcal{V}_i'$. If $x_v^* = 1$ where $v \in \mathcal{V}_i$, then $\Delta g_v(n_v^{\mathcal{G}_i}) \geq c_v$ and thus we have $\Delta g_v^t(n_v^{\mathcal{T}_i}) \geq c_v$. Hence, v does not deviate in \mathcal{T}_i . Similarly, if $x_v^* = 0$ where $v \in \mathcal{V}_i$, then $\Delta g_v(n_v^{\mathcal{G}_i}) \leq c_v$ and thus we have $\Delta g_v^t(n_v^{\mathcal{T}_i}) \leq c_v$. Hence, v does not deviate in \mathcal{T}_i . Hence $(x_v^*)_{v \in \mathcal{V}_i}$ is also a PSNE in BNPG game $(\mathcal{T}_i, (g_v^t)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$ where $t = (x_v^*)_{v \in \mathcal{V}_i}$ (which means our Algorithm returns YES).

In the other direction, let $(x_v^*)_{v \in \mathcal{V}_i}$ be the PSNE in BNPG game on $(\mathcal{T}_i, (g_v^t)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$ where $t = (x_v^*)_{v \in \mathcal{V}_i}$ (which means our Algorithm returns YES). We claim that $(x_v^*)_{v \in \mathcal{V}_i}$ is also a PSNE in BNPG game $(\mathcal{G}_i, (g_v)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$. If $x_v^* = 1$ for $v \in \mathcal{V}_i$, then $\Delta g_v^t(n_v^{\mathcal{T}_i}) \geq c_v$. This implies that $\Delta g_v(n_v^{\mathcal{G}_i}) \geq c_v$ and thus v does not deviate in \mathcal{G}_i . Similarly, if $x_v^* = 0$ for $v \in \mathcal{V}_i$, then $\Delta g_v^t(n_v^{\mathcal{T}_i}) \leq c_v$. This implies that $\Delta g_v(n_v^{\mathcal{G}_i}) \leq c_v$ and thus v does not deviate in \mathcal{G}_i . Hence $(x_v^*)_{v \in \mathcal{V}_i}$ is also a PSNE in BNPG game on $(\mathcal{G}_i, (g_v)_{v \in \mathcal{V}_i}, (c_v)_{v \in \mathcal{V}_i})$. \square

Yu et al. presented an algorithm for EXISTS-PSNE on complete graphs in [29]. It turns out that their algorithm can be appropriately modified to get the following observation.

OBSERVATION 4. [29] *Given a BNPG game on a complete graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, and an integer k , there is a polynomial time algorithm for deciding if there exists a PSNE where exactly k players play 1 and returns such a PSNE if it exists.*

Now by using the observation 4 as a subroutine, we exhibit an FPT algorithm for the parameter distance from complete graph.

THEOREM 4.15. *There is an algorithm running in time $O^*(4^{d_2})$ for EXISTS-PSNE in BNPG games where d_2 is the distance from complete graph.*

PROOF. Let $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$ be any instance of EXISTS-PSNE for BNPG games. If $d_2 \geq \frac{n}{2}$, then iterating over all possible strategy profiles takes time $O^*(2^n) \leq O^*(4^{d_2})$. So allow us to assume for the rest of the proof that $d_2 < \frac{n}{2}$. Let us define $\mathcal{V}' = \{u \in \mathcal{V} : \exists v \in \mathcal{V}, v \neq u, \{u, v\} \notin \mathcal{E}\}$; we have $|\mathcal{V}'| \leq 2d_2$.

For every strategy profile $y = (y_u)_{u \in \mathcal{V}'}$, we do the following. For each $v \in \mathcal{V} \setminus \mathcal{V}'$, let n_v' be the number of neighbors of v in \mathcal{V}' who play 1 in y . We now define $g_v^t(\ell) = g_v(\ell + n_v')$ for every $\ell \in \mathbb{N} \cup \{0\}$ and every player $v \in \mathcal{V} \setminus \mathcal{V}'$. For every $k \in \{0, \dots, |\mathcal{V} \setminus \mathcal{V}'|\}$, we decide (using the algorithm in Observation 4) if there exists a PSNE $x^k = (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'}$ in the BNPG game $(\mathcal{G}[\mathcal{V} \setminus \mathcal{V}'], (g_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'}, (c_v)_{v \in \mathcal{V} \setminus \mathcal{V}'})$ where exactly k players play 1. If x^k exists, then we output YES if $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$ forms a PSNE in the BNPG game $(\mathcal{G} = (\mathcal{V}, \mathcal{E}), (g_v)_{v \in \mathcal{V}}, (c_v)_{v \in \mathcal{V}})$.

If the above procedure fails to find a PSNE, then we output NO. The running time of the above algorithm is $O^*(2^{|\mathcal{V}'|}) \leq O^*(4^{d_2})$. We now argue correctness.

Clearly, if the algorithm outputs YES, then there exists a PSNE for the input game. On the other hand, if there exists a PSNE $((y_u)_{u \in \mathcal{V}'}, (x_v)_{v \in \mathcal{V} \setminus \mathcal{V}'}) \in \{0, 1\}^{\mathcal{V}}$ in the input game, then let us consider the iteration of our algorithm with the guess $(y_u)_{u \in \mathcal{V}'}$. Let the number of players playing 1 in $(x_v)_{v \in \mathcal{V} \setminus \mathcal{V}'}$ be k . If $x_v = 1$ where $v \in \mathcal{V} \setminus \mathcal{V}'$, then $\Delta g_v(n_v' + k - 1) \geq c_v$ and thus we have $\Delta g_v^k(k - 1) \geq c_v$. Similarly, if $x_v = 0$ where $v \in \mathcal{V} \setminus \mathcal{V}'$, then $\Delta g_v(n_v' + k) \leq c_v$ and thus we have $\Delta g_v^k(k) \leq c_v$. Hence, we observe that $(x_v)_{v \in \mathcal{V} \setminus \mathcal{V}'}$ forms a PSNE in the BNPG game $(\mathcal{G}[\mathcal{V} \setminus \mathcal{V}'], (g_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'}, (c_v)_{v \in \mathcal{V} \setminus \mathcal{V}'})$. Let $(x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'}$ be the PSNE of the BNPG game $(\mathcal{G}[\mathcal{V} \setminus \mathcal{V}'], (g_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'}, (c_v)_{v \in \mathcal{V} \setminus \mathcal{V}'})$ where exactly k players play 1 returned by the algorithm in Observation 4. We observe that every player in \mathcal{V}' has the same number of neighbors playing 1 in both the strategy profiles $((y_u)_{u \in \mathcal{V}'}, (x_v)_{v \in \mathcal{V} \setminus \mathcal{V}'})$ and $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$. So no player in \mathcal{V}' will deviate in the strategy profile $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$. If $x_v^k = 1$ where $v \in \mathcal{V} \setminus \mathcal{V}'$, then $\Delta g_v^k(k - 1) \geq c_v$ and thus we have $\Delta g_v(n_v' + k - 1) \geq c_v$. Hence, v does not deviate in the strategy profile $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$. Similarly, if $x_v^k = 0$ where $v \in \mathcal{V} \setminus \mathcal{V}'$, then $\Delta g_v^k(k) \leq c_v$ and thus we have $\Delta g_v(n_v' + k) \leq c_v$. Hence, v does not deviate in the strategy profile $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$. Hence, $((y_u)_{u \in \mathcal{V}'}, (x_v^k)_{v \in \mathcal{V} \setminus \mathcal{V}'})$ also forms a PSNE in the input BNPG game and thus the algorithm outputs YES. This concludes the correctness of our algorithm. \square

We finally show that a PSNE always exists for fully homogeneous BNPG games for some important graph classes and such a PSNE can be found in $O(n)$ time.

THEOREM 4.16 (★). *There is always a PSNE in a fully homogeneous BNPG game for paths, complete graphs, cycles, and bi-cliques. Moreover, we can find a PSNE in $O(n)$ time.*

5 CONCLUSION AND FUTURE WORK

We have studied parameterized complexity of the EXISTS-PSNE problem for the BNPG games with respect to various important graph parameters. We exhibited intractability w.r.t. the parameters like maximum degree, diameter, treedepth, number of players playing 1 and 0. We complemented this by showing FPT algorithms parameterized by circuit rank, treewidth+maximum degree, and the distance from complete graph. We also showed that PSNE always exists in a fully homogeneous BNPG game for paths, complete graphs, cycles and bi-cliques.

Our work leaves some important questions open. For example, can we show PPAD-Hardness for finding Nash Equilibrium in BNPG games. Another immediate research direction is to study if our algorithmic results could be extended to other types of more general public goods games. Another research direction could be to look at social welfare functions in the context of BNPG game. We can also consider BNPG games with altruism introduced in [28] and try to resolve its parameterized complexity.

REFERENCES

- [1] Piotr Berman, Marek Karpinski, and A Scott. 2004. *Approximation hardness of short symmetric instances of MAX-3SAT*. Technical Report.
- [2] Yann Bramoullé, Rachel Kranton, et al. 2007. Public goods in networks. *Journal of Economic Theory* 135, 1 (2007), 478–494.
- [3] Elisabeth Buchwald. 2020. Why do so many Americans refuse to wear face masks? Politics is part of it – but only part. <https://www.marketwatch.com/story/why-do-so-many-americans-refuse-to-wear-face-masks-it-may-have-nothing-to-do-with-politics-2020-06-16>.
- [4] Ronald S Burt. 1987. Social contagion and innovation: Cohesion versus structural equivalence. *American Journal of Sociology* 92, 6 (1987), 1287–1335.
- [5] Timothy G Conley and Christopher R Udry. 2010. Learning about a new technology: Pineapple in Ghana. *American economic review* 100, 1 (2010), 35–69.
- [6] Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michał Pilipczuk, and Saket Saurabh. 2015. *Parameterized Algorithms*. Springer.
- [7] Luca Dall’Asta, Paolo Pin, and Abolfazl Ramezani. 2011. Optimal equilibria of the best shot game. *Journal of Public Economic Theory* 13, 6 (2011), 885–901.
- [8] Constantinos Daskalakis, Paul W Goldberg, and Christos H Papadimitriou. 2009. The complexity of computing a Nash equilibrium. *SIAM J. Comput.* 39, 1 (2009), 195–259.
- [9] Edith Elkind, Leslie Ann Goldberg, and Paul Goldberg. 2006. Nash equilibria in graphical games on trees revisited. In *Proceedings of the 7th ACM Conference on Electronic Commerce*. 100–109.
- [10] Andrea Galeotti, Sanjeev Goyal, Matthew O Jackson, Fernando Vega-Redondo, and Leat Yariv. 2010. Network games. *The review of economic studies* 77, 1 (2010), 218–244.
- [11] Georg Gottlob, Gianluigi Greco, and Francesco Scarcello. 2005. Pure Nash equilibria: Hard and easy games. *Journal of Artificial Intelligence Research* 24 (2005), 357–406.
- [12] Nicole Immorlica, Jon Kleinberg, Mohammad Mahdian, and Tom Wexler. 2007. The role of compatibility in the diffusion of technologies through social networks. In *Proceedings of the 8th ACM conference on Electronic commerce*. 75–83.
- [13] Yoichi Iwata, Tomoaki Ogasawara, and Naoto Ohsaka. 2017. On the power of tree-depth for fully polynomial FPT algorithms. *arXiv preprint arXiv:1710.04376* (2017).
- [14] Michael Kearns. 2007. Graphical games. *Algorithmic game theory* 3 (2007), 159–180.
- [15] Jon Kleinberg. 2007. Cascading behavior in networks: Algorithmic and economic issues. *Algorithmic game theory* 24 (2007), 613–632.
- [16] Peter Kollock. 1998. Social dilemmas: The anatomy of cooperation. *Annual review of sociology* 24, 1 (1998), 183–214.
- [17] Zohar Komarovsky, Vadim Levit, Tal Grinshpoun, and Amnon Meisels. 2015. Efficient equilibria in a public goods game. In *2015 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*, Vol. 2. IEEE, 214–219.
- [18] Vadim Levit, Zohar Komarovsky, Tal Grinshpoun, and Amnon Meisels. 2018. Incentive-based search for efficient equilibria of the public goods game. *Artificial Intelligence* 262 (2018), 142–162.
- [19] Arnab Maiti and Palash Dey. 2020. On Parameterized Complexity of Binary Networked Public Goods Game. *arXiv preprint arXiv:2012.01880* (2020).
- [20] Vahideh H Manshadi and Ramesh Johari. 2009. Supermodular network games. In *2009 47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*. IEEE, 1369–1376.
- [21] Rachel E. Morgan and Jennifer L. Truman. 2017. *Criminal victimization*. Technical Report.
- [22] Stephen Morris. 2000. Contagion. *The Review of Economic Studies* 67, 1 (2000), 57–78.
- [23] Marko Samer and Stefan Szeider. 2011. Tractable cases of the extended global cardinality constraint. *Constraints* 16, 1 (2011), 1–24.
- [24] Francisco C Santos, Marta D Santos, and Jorge M Pacheco. 2008. Social diversity promotes the emergence of cooperation in public goods games. *Nature* 454, 7201 (2008), 213–216.
- [25] Thomas W Valente. 1995. *Network models of the diffusion of innovations*. Number 303.484 V3.
- [26] Thomas W Valente. 2005. Network models and methods for studying the diffusion of innovations. *Models and methods in social network analysis* 28 (2005), 98.
- [27] Tessa Wong. 2020. Coronavirus: Why some countries wear face masks and others don’t. <https://www.bbc.com/news/world-52015486>.
- [28] Sixie Yu, David Kempe, and Yevgeniy Vorobeychik. 2021. Altruism Design in Networked Public Goods Games. *arXiv preprint arXiv:2105.00505* (2021).
- [29] Sixie Yu, Kai Zhou, P Jeffrey Brantingham, and Yevgeniy Vorobeychik. 2020. Computing Equilibria in Binary Networked Public Goods Games. In *AAAI* 2310–2317.