Pure and Strong Nash Equilibrium Computation in Compactly Representable Aggregate Games

Jared Soundy[1] **Mohammad T. Irfan**[2] **Hau Chan**[3]

[1] The Beacom College of Computer & Cyber Sciences, Dakota State University, Madison, South Dakota, USA
[2] Department of Computer Science, Bowdoin College, Brunswick, Maine, USA
[3] School of Computing, University of Nebraska-Lincoln, Lincoln, Nebraska, USA

Abstract

Aggregate games model interdependent decision making when an agent's utility depends on their own choice and the aggregation of everyone's choices. We define a compactly representable subclass of aggregate games we call additive aggregate games, which encompasses popular games like congestion games, anonymous games, Schelling games, etc. We study computational questions on pure Nash equilibrium (PNE) and pure strong Nash equilibrium (SNE). We show that PNE existence is NP-complete for very simple cases of additive aggregate games. We devise an efficient algorithmic scheme for deciding the existence of a PNE and computing one (if it exists) for bounded aggregate space. We also give an approximation algorithm for a special type of additive aggregate games. For SNE, we show that SNE recognition is co-NP-complete and SNE existence is Σ_2^P-complete, even for simple types of additive aggregate games. For broad classes, we provide several novel and efficient aggregate-space algorithms for recognizing an SNE and deciding the existence of an SNE. Finally, we connect our results to several popular classes of games and show how our computational schemes can shed new light on these games.

1 INTRODUCTION

The conceptualization of aggregate games is often attributed to Nobel laureate Reinhard Selten, who studied aggregation in linear-quadratic models in a book written in German [Selten, 1970]. Aggregate games mark a paradigm shift in game theory by connecting individual choices to population-level aggregation of choices. This aggregation can wash away the identities of agents and produce an aggregate measure of their choices. This has considerable implications on representation as well as computation. Not surprisingly, aggregate games have been applied to voting [Kearns and Mansour, 2002], stock markets [Cummings et al., 2015], market production [Babichenko, 2013], and resource allocation [Martimort and Stole, 2012], among other applications.

Furthermore, aggregate games generalize many widely studied classes of games. As a result, computational advances on aggregate games may contribute to understanding large-scale models of transportation systems with the presence of autonomous vehicles [Wang et al., 2019], wireless and telecommunication networks [Altman et al., 2006, Yamamoto, 2015], smart grids [Fadlullah et al., 2011], etc. By examining equilibrium behaviors of agents in these systems, relevant policymakers or stakeholders can better design infrastructure (e.g., by modifying networks) that lead to better equilibrium outcomes (e.g., advising agents to take certain actions to increase network efficiency) [Zardini et al., 2021]. Similar causal inference questions have been studied in other compactly representable games, such as influence games on networks [Irfan, 2013, Irfan and Ortiz, 2014].

Since an agent's utility in an aggregate game is defined using an aggregate measure of everyone's choices, aggregate games showcase succinct representation, especially when the aggregate measure is additively decomposable across the agents. We name such games *additive aggregate games*. Let us first situate these games within the broad and growing classes of compactly representable games. We leave a more detailed exposition to the Appendix.

A widely studied class of compact games is *graphical games* from UAI'01 [Kearns et al., 2001]. We can define a graphical game on a complete graph to capture an aggregate game, but it loses the computational appeal. The additive decomposition in additive aggregate games may ring a bell with *polymatrix games* [Janovskaja, 1968]. However, neither additive aggregate games nor polymatrix games contain the other. Yet another widely applicable class of compact games is *action graph games* (AGGs) from UAI'04 [Bhat and Leyton-Brown, 2004]. AGGs contain aggregate games but only by having a complete action graph, thereby losing their algorithmic features [Jiang et al., 2011]. On the reverse side of containment, additive aggregate games contain many classes of compact games, such as *congestion games*, *anonymous games*, *Schelling games*, and *Cournot games*. We explore these connections in Section 5.

The succinct representation of additive aggregate games comes with its own computational challenges because the running time of an algorithm is evaluated based on its input size. As we will see, computational questions on aggregate games are notoriously hard, often residing above NP in the polynomial hierarchy. Our goal is to study these provably hard problems and design algorithms for them. We study two solution concepts: pure Nash equilibrium (PNE) and pure strong Nash equilibrium (SNE).

Technical Contributions

We define a new subclass of aggregate games that we call additive aggregate games. For PNE, we show that deciding the existence of a PNE is NP-complete for very special cases of additive aggregate games. We then devise an algorithm for PNE computation. The algorithm is polynomial-time for bounded aggregate space. We also show the existence of ϵ-approximate PNE in a special type of aggregate games and give an algorithm to compute it.

For SNE, we show that recognizing whether an action profile is an SNE is co-NP-complete and determining the existence of an SNE is Σ_2^P-complete, even for simple types of additive aggregate games. For large classes of aggregate games, we provide several novel and efficient aggregate-space algorithms for recognizing an SNE and determining the existence of an SNE.

We connect our results to popular classes of games like congestion games, anonymous games, Schelling games, and Cournot games and show how our approach sheds new light on these very well-studied games. We derive several new results through these connections, including the first algorithm for SNE computation in Schelling games.

Significance

Although there are known hardness results for subclasses of aggregate games like weighted congestion games and anonymous games, these results do not necessarily carry over to other subclasses of aggregate games. For example, we show that SNE existence and computation are co-NP-complete and Σ_2^P-complete, respectively, for additive aggregate games. In contrast, these are in P and NP-complete, respectively, for anonymous games. Therefore, our hardness results on very narrow subclasses of additive aggregate games provide new knowledge on the boundary of tractable computation.

Our algorithm for PNE computation explores the aggregate space systematically through the non-trivial construction of a mapping between the aggregate space and action profiles using multipartite graph-based dynamic programming.

Our contributions to SNE recognition and computation fill in some important gaps in the literature on aggregate games. For general instances of additive aggregate games, we are not aware of any work on the hardness of and algorithms for recognizing and computing SNE. We provide novel algorithms for addressing these provably hard problems – the first algorithms for these problems to our knowledge.

Finally, the significance of our work is not limited to aggregate games alone. The connection between additive aggregate games and various popular classes of games shows the application potential of this study.

Related Work

Aggregate games model scenarios where an agent's utility function depends on the agent's own actions and the aggregation of everyone's actions [Jensen, 2010, Acemoglu and Jensen, 2013, Corchón, 1994, Koshal et al., 2016, Martimort and Stole, 2011, Cornes and Hartley, 2012, Martimort and Stole, 2012]. Most of the early work on aggregate games arose from economics, while computational work has been gaining traction lately.

Barring sporadic results on the existence of PNE for specific types of aggregate games [Jensen, 2010, Martimort and Stole, 2012], PNE *computation* did not get much attention in the aggregate games literature. In fact, the existence and computation of PNE have been mostly studied for subclasses of aggregate games, such as congestion games [Fabrikant et al., 2004, Ackermann et al., 2008, Vöcking and Aachen, 2006], anonymous games [Blonski, 2000, Carmona and Podczeck, 2020], and Schelling games [Elkind et al., 2019, Echzell et al., 2019, Chan et al., 2020]. There also exist results on approximate mixed Nash equilibrium computation [Cummings et al., 2015, Kearns and Mansour, 2002, Babichenko, 2013]. In contrast, we give a comprehensive treatment of PNE existence and computation for additive aggregate games. Additionally, we connect our results to other classes of games.

Compared to PNE, SNE is a stronger solution concept that is immune to deviations by coalitions. As one may expect, there is very little in the literature on SNE computation for general aggregate games. The few SNE results focus on specific classes of games, mostly different variants of congestion games [Holzman and Law-Yone, 1997, Hoefer and Skopalik, 2013, Rozenfeld and Tennenholtz, 2006, Hayrapetyan et al., 2006, Epstein et al., 2007, Gourves and Monnot, 2009, Holzman and Law-Yone, 1997]. There are some results on anonymous games [Hoefer and Skopalik, 2013] and continuous games [Nessah and Tian, 2014]. There is also some work on strong Nash equilibria computation in mixed strategies [Gatti et al., 2013]. In contrast, we address two SNE-related problems here: recognition of an SNE and computation of an SNE (if it exists). We study the hardness of and algorithms for these two problems.

2 PRELIMINARIES

We start with some game-theoretic notation and then define aggregate games. In prior work, aggregate games have also been referred to as summarization games [Cummings et al., 2015, Kearns and Mansour, 2002].

Let $N = \{1, \cdots, n\}$ be a set of n agents in a game. Each agent $i \in N$ has a set A_i of actions and selects an action $a_i \in A_i$. Let $m = \max_{i \in N} |A_i|$ be the maximum number of actions of any agent. Let $\boldsymbol{A} = A_1 \times A_2 \times ... \times A_n$ be the set of action profiles of all agents where an action profile $\boldsymbol{a} = (a_1, a_2, \cdots, a_n) \in \boldsymbol{A}$ consists of an action for each agent. Given an action profile $\boldsymbol{a} = (a_i, \boldsymbol{a}_{-i}) \in \boldsymbol{A}$, we use $\boldsymbol{a}_{-i}$ to refer to the actions of all agents except agent i. Given a subset of agents $I \subseteq N$, we use $\boldsymbol{a}_I$ to refer to the actions of each agent in I and $\boldsymbol{a}_{-I}$ to refer to the actions of agents not in I. To model the inter-dependence among the agents, for each agent $i \in N$, there is a utility function $u_i : \boldsymbol{A} \rightarrow \mathbb{R}$ that maps each action profile to a real number.

Compared to general games, an agent's utility function in an aggregate game depends the agent's action and the aggregation of everyone's actions (including the actions different from the agent's own action). To capture this aggregation, we define an aggregator function $\phi : \boldsymbol{A} \rightarrow Y$ that maps each action profile to an aggregate measure in the space Y. Here, $Y \subset \mathbb{Z}^d_{\geq 0}$ is a bounded and countable (discrete) aggregate space of d-dimensional vectors of non-negative integers. We denote agent i's utility function as $\pi_i : A_i \times Y \rightarrow \mathbb{R}$, which maps i's action and an aggregate to a real number.

Definition 2.1 (Aggregate Game)**.** The tuple $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ defines an aggregate game. It consists of a set N of agents, a set A_i of actions for each agent i, and a utility function $\pi_i(a_i, \phi(\boldsymbol{a}))$ for each agent $i \in N$ and $\boldsymbol{a} \in \boldsymbol{A}$, where π_i is a function of i's actions in A_i and the aggregator function ϕ's outputs in Y.

We make the standard assumption that the utility functions and the aggregator function of an aggregate game are given implicitly via value oracles or parameterized functions [Rosenthal, 1973, Daskalakis and Papadimitriou, 2015]. This implies that they can be evaluated efficiently. We next define additive aggregate games.

Definition 2.2 (Additive Aggregate Game)**.** An additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ is an aggregate game where, for any $\boldsymbol{a} = (a_1, \cdots, a_n) \in \boldsymbol{A}$, the aggregator function is additively separable: $\phi(\boldsymbol{a}) = \phi_1(a_1) + \cdots + \phi_n(a_n)$ for some function $\phi_i : A_i \rightarrow Y$ for each $i \in N$.

Since the aggregator function is additively separable and addition is well-defined over Y, for any $\boldsymbol{a}_I$ and $y, y' \in Y$, $\phi(\boldsymbol{a}_I) \pm y \pm y'$ is also well-defined. Furthermore, we can evaluate $\phi_i(a_i)$ for any i and $a_i \in A_i$ in $O(1)$ time.

For the most part, we focus on additive aggregate games due to their ability to capture large classes of commonly studied games. Given an instance of an additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$, we are interested in pure Nash equilibrium (PNE) and pure strong Nash equilibrium (SNE) concepts, where agents choose their actions deterministically. Below, we define PNE using best responses and SNE using joint deviations.

Definition 2.3 (Best Response)**.** Given the actions of other agents $\boldsymbol{a}_{-i}$, an agent i's action $a_i^* \in A_i$ is a best response if and only if

$\pi_i(a_i^*, \phi(a_i^*, \boldsymbol{a}_{-i})) \geq \pi_i(a_i, \phi(a_i, \boldsymbol{a}_{-i}))$ for any $a_i \in A_i$.

Definition 2.4 (Pure Nash Equilibrium (PNE))**.** An action profile $\boldsymbol{a}^* \in \boldsymbol{A}$ is a pure Nash equilibrium if and only if for each agent $i \in N$, i's action a_i^* is a best response to $\boldsymbol{a}^*_{-i}$.

In a PNE, no agent has an incentive to unilaterally deviate to another action. The well-motivated SNE concept [Aumann, 1959] extends the PNE concept to joint deviations. That is, no group of agents of any size can deviate together such that each agent in the group can gain from the joint deviation.

Definition 2.5 (Pure Strong Nash Equilibrium (SNE))**.** An action profile $\boldsymbol{a}^* \in \boldsymbol{A}$ is a pure strong Nash equilibrium if and only if there is no group of agents $I \subseteq N$ such that, for some joint deviation $\boldsymbol{a}_I$ and each agent $i \in I$,

$\pi_i(a_i, \phi(\boldsymbol{a}_I, \boldsymbol{a}^*_{-I})) > \pi_i(a_i^*, \phi(\boldsymbol{a}^*_I, \boldsymbol{a}^*_{-I}))$.

When the group size is one, SNE coincides with PNE. We next define an ϵ-approximate PNE as an additive approximation. Notably, additive approximations are a lot more prevalent than multiplicative in game theory [Daskalakis et al., 2007, Daskalakis and Papadimitriou, 2015]. For it to make sense, payoffs must be scaled, usually between [0, 1].

Definition 2.6 (ϵ-approximate PNE)**.** An action profile $\boldsymbol{a}^* \in \boldsymbol{A}$ is an ϵ-approximate PNE if and only if for each player $i \in N$ and any $a_i \in A_i$, we have that $\pi_i(a_i^*, \phi(a_i^*, \boldsymbol{a}^*_{-i})) \geq \pi_i(a_i, \phi(a_i, \boldsymbol{a}^*_{-i})) - \epsilon$.

3 PNE COMPUTATION

We establish the hardness of PNE existence in very simple instances of additive aggregate games. We then complement the hardness results by providing an efficient aggregate-space algorithm for bounded aggregate space. We also investigate approximation algorithms in this section.

3.1 HARDNESS OF COMPUTING A PNE

We show that PNE existence is NP-complete even for very special classes of additive aggregate games. Since additive aggregate games are commonly studied in the literature in various forms (see Section 5), we first contextualize our hardness results within the array of known hardness results for related classes of games.

As we show in Section 5, weighted congestion games are a type of additive aggregate games. As a result, the known hardness results for weighted congestion games are applicable to additive aggregate games. For example, PNE existence in a weighted symmetric network congestion game is strongly NP-complete [Dunkel and Schulz, 2008]. Although a PNE is guaranteed to exist in asymmetric network congestion games, computing it is PLS-complete [Fabrikant et al., 2004] even when the cost function is linear [Ackermann et al., 2008]. However, the hardness results from the congestion games literature do not readily apply to subclasses of aggregate games that are not congestion games.

This brings us to anonymous games, which is a well-studied subclass of additive aggregate games but is neither a subclass nor a superclass of congestion games. Deciding the existence of a PNE in anonymous games is NP-complete, even when the number of actions is linear in the number of players and there is a constant number of payoffs [Brandt et al., 2009]. This result does apply to additive aggregate games. We next show strong NP-completeness of very special instances of aggregate games that are not necessarily anonymous. Our proof technique, which uses a reduction from 3-Partition (please see the Appendix), also differs substantially from [Brandt et al., 2009].

Theorem 3.1. *It is strongly NP-complete to decide PNE existence in additive aggregate games,*

even when the number of actions is linear in the number of agents, the dimension of the aggregate space is linear in the number of agents, and the utility function of each agent returns two integer values.

Furthermore, we use a reduction from the Partition problem to show that PNE existence in additive aggregate games remains hard even when the number of each agent's actions and the dimension of the aggregate space are constant.

Theorem 3.2. *It is NP-complete to decide PNE existence in additive 2-action aggregate games, even when the dimension of the aggregate space is constant and the utility function of each agent returns two integer values.*

3.2 ALGORITHMS FOR COMPUTING A PNE

We utilize the structure and parameters of additive aggregate games to design efficient general-purpose algorithms for determining the existence of a PNE. To design efficient algorithms for PNE computation in additive aggregate games, our hardness results suggest that we must impose additional conditions. For instance, a straightforward brute-force algorithm for any game is to check whether each action profile is a PNE of an aggregate game. The algorithm runs in exponential time in the number of players (i.e., $O(nm^{n+1})$ with n agents and at most m actions). For bounded n, this algorithm is rather efficient. However, even for $m = 2$, such an algorithm is not efficient.

Given the insights from our hardness results, a natural direction is to examine parameters that deal with the size of the aggregate space. To this end, we introduce a novel, general aggregate-space algorithm that is efficient when the size of the aggregate space is bounded.

Our algorithmic approach systematically explores the aggregate space and determines whether an aggregate is consistent with a PNE (i.e., for a given $y \in Y$, is there a PNE $\boldsymbol{a}^*$ such that $\phi(\boldsymbol{a}^*) = y$?) by solving three subproblems described below. The fundamental algorithmic approach was derived from a CSP formulation in the context of congestion games in [Irfan et al., 2024]. Here, we extend it to general additive aggregate games that encompass many classes of games beyond congestion games (see Section 5). In addition, we later consider SNE computation, which did not get any attention in [Irfan et al., 2024]. Also, instead of using CSPs, we use multipartite graphs for a cleaner presentation.

Given an additive aggregate game and an aggregate, the first and second problems below seek to determine new aggregates and best responses when an agent deviates to other actions using only aggregates without considering the actions of the other agents.

Problem 3.3 (Deviation)**.** Given an additive aggregate game instance $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$ and $y \in Y$, for each agent $i \in N$ and any $a_i, a_i' \in A_i$, compute $\texttt{deviate}(a_i, y, a_i') = \hat{y}$ which returns an aggregate $\hat{y} = (y - \phi_i(a_i)) + \phi_i(a_i')$ ($\in Y$ or $\notin Y$) when agent i deviates from a_i to a_i' under y.

Problem 3.4 (Aggregate Best Response)**.** Given an additive aggregate game $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$ and $y \in Y$, for each agent $i \in N$, find the set of best-response actions given aggregate y. That is,

$$BR_i(y) = \{a_i \in A_i \mid \pi_i(a_i, y) \geq \pi_i(a_i', \hat{y}) \text{ for each } a_i' \in A_i \text{ such that } \texttt{deviate}(a_i, y, a_i')) = \hat{y} \in Y\}. \quad (1)$$

As a foreword, our algorithm will ensure the compatibility between y and a_i. Also note how Problem 3.4 and Definition 2.3 differ in their respective definitions of best response. In Definition 2.3, a best response of an agent is defined with respect to the actions of the other agents, but in Problem 3.4, a best response is defined with respect to an aggregate.

The next problem roughly asks whether there is an action profile that maps to a given aggregate.

Problem 3.5 (Construction)**.** Given an additive aggregate game $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$, $y \in Y$, and a subset $\tilde{A}_i \subseteq A_i$ of actions for each agent $i \in N$, determine if there exists $\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i\in N} \tilde{A}_i$ such that $\phi(\tilde{\boldsymbol{a}}) = y$.

Solving the above three subproblems will allow us to verify whether there is a PNE that can be mapped to a given aggregate y. In particular, in Problem 3.5, instantiating $\tilde{A}_i$ with $BR_i(y)$ for each i from Problem 3.4, we wish to construct an action profile that maps to y. Such an action profile must be a PNE. We next give an aggregate-space algorithm for PNE existence in additive aggregate games.

Theorem 3.6. *Given an additive aggregate game, suppose that Problem 3.3, Problem 3.4, and Problem 3.5 can be solved in $O(\alpha)$, $O(nm^2\alpha)$, and $O(\beta)$, respectively. There is an $O(|Y|(nm^2\alpha + \beta))$ algorithm for determining the existence of a PNE and returning a PNE (if it exists) for additive aggregate games.*

Proof Sketch. For each $y \in Y$, perform the following steps.

Step 1. For each agent $i \in N$, we compute $BR_i(y)$ the set of aggregate best-response actions given y (Problem 3.4). We require $O(nm^2\alpha)$ time in this step.

Step 2. Given the $BR_i(y)$ for each agent i, determine if there exists $\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i\in N} BR_i(y)$ such that $\phi(\tilde{\boldsymbol{a}}) = y$. If there is such a profile $\tilde{\boldsymbol{a}}$, then $\tilde{\boldsymbol{a}}$ is a PNE. We require $O(\beta)$ time in this step. □

For additive aggregate games, Problem 3.3 can be solved in O(1) time because for any $a_i \in A_i$, $\phi(a_i)$ amounts to a table lookup. Therefore, Problem 3.4 can be computed in $O(m^2)$ time for each agent. Unfortunately, Problem 3.5 is strongly NP-complete. Please see the Appendix for the proof.

Theorem 3.7. *It is strongly NP-complete to determine the existence of an action profile that maps to a given aggregate from given sets of actions of each agent in additive aggregate games (Problem 3.5).*

Despite the above hardness result, we next present an efficient graph-based dynamic programming algorithm that is polynomial in the size of the aggregate space Y for additive aggregate games.

Theorem 3.8. *There is an $O(|Y|nm)$ constructive algorithm for determining the existence of an action profile that maps to a given aggregate from given sets of actions of each agent in additive aggregate games (Problem 3.5).*

Proof. We iteratively construct an $(n + 1)$-partite graph $G = (Y_0, Y_1, Y_2, ..., Y_n, E)$, where $Y_0 = \{(0, 0, ..., 0)\}$ and $Y_i \subseteq Y$ for $i = 1, ..., n$, and edges in E only form between Y_{i-1} and Y_i for $i = 1, ..., n$. For $i = 1, ..., n$, $y_{i-1} \in Y_{i-1}$, and $y_i \in Y_i$, (y_{i-1}, y_i) is an edge if and only if (a) either $i = 1$ and $y_0 = (0, ..., 0)$, or $i > 1$ and there is an edge (y_{i-2}, y_{i-1}) for some $y_{i-2} \in Y_{i-2}$ and (b) $y_i = y_{i-1} + \phi_i(\tilde{a}_i)$ for some $\tilde{a}_i \in \tilde{A}_i$. There is an action profile that corresponds to the given aggregate $y \in Y$ if and only if there is a path from $y_0 \in Y_0$ to $y \in Y_n$. The pseudocode is given in Algorithm 1.

The running time of Algorithm 1 is $O(|Y|nm)$ because there are n iterations and at each iteration, we do $O(|Y|m)$ amount of work. If there is a path from $y_0 \in Y_0$ to y, then there must be an action profile that can be mapped or summed to y by the graph construction. The existence of such a path can be verified by checking whether $y \in Y_n$. If we find $y \in Y_n$, we can compute an exact action profile by tracing the path from y back to y_0 using the edge information. By construction of the graph, if there are multiple edges to a node y_i from the previous layer $(i - 1)$, any one can be chosen arbitrarily without disrupting a path back to y_0. □

Algorithm 1 Determining the existence of an action profile that maps to a given aggregate from given sets of actions of each agent in additive aggregate games (Problem 3.5)

Input: Additive aggregate game $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$, $y \in Y$, and a subset $\tilde{A}_i \subseteq A_i$ of action for each agent $i \in N$
Output: If there exists $\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i\in N} \tilde{A}_i$ such that $\phi(\tilde{\boldsymbol{a}}) = y$.

Let $G = (Y_0, ..., Y_n, E)$ be $(n + 1)$-partite graph where $Y_0 = \{(0, 0, \cdots, 0)\}$ and $Y_i = \{\}$ for $i \in \{1, 2, \cdots, n\}$
for $i \in \{1, 2, \cdots, n\}$ **do**
  for $y_{i-1} \in Y_{i-1}$ **do**
    for $\tilde{a}_i \in \tilde{A}_i$ **do**
      Let $y_i = y_{i-1} + \phi_i(\tilde{a}_i)$
      $Y_i = Y_i \cup \{y_i\}$
      $E = E \cup \{(y_{i-1}, y_i)\}$
return True [if $y \in Y_n$] or False [otherwise]

Theorems 3.6 and 3.8 lead to the following result.

Corollary 3.9. *There is an $O(|Y|(nm^2 + |Y|nm))$ algorithm for determining the existence of a PNE and returning a PNE (if it exists) for additive aggregate games.*

3.3 APPROXIMATION ALGORITHM FOR PNE

Against the backdrop of hardness results for very special types of additive aggregate games, we have some positive results on the approximation algorithms front. We give an approximation algorithm for a class of games that sits between additive aggregate games and anonymous games. We call this class *weighted anonymous games*.

In a weighted anonymous game, each agent $i \in N$ has a weight $w \in \mathbb{R}^+$. For each agent i, the payoff function is defined as $\rho_i : A \times Y \to \mathbb{R}$, where A is the common set of actions and Y is the space of all m-dimensional vectors representing the total weights of the agents selecting each action under an action profile. In contrast, in an anonymous game, vectors in Y consist of the *number* of agents selecting each action (see Section 5.2). We use the following result and the algorithm therein.

Theorem 3.10. *([Daskalakis and Papadimitriou, 2015]) Any λ-Lipschitz anonymous game with payoffs in $[0, 1]$ has an $O(m\lambda)$-approximate PNE.*

In the above theorem, for a real-valued $\lambda > 0$, an anonymous game is λ-Lipschitz if and only if for any agent i, action $a_i \in A_i$ and m-dimensional vectors $x, y \in Y$, $|\rho_i(a_i, x) - \rho_i(a_i, y)| \leq \lambda \cdot ||x - y||_{L_1}$. We have the following result for *weighted* anonymous games.

Theorem 3.11. *Any λ-Lipschitz weighted anonymous game with payoffs in $[0, 1]$ has an $O(mw\lambda)$-approximate PNE.*

Proof Sketch. Given a λ-Lipschitz weighted anonymous game, we construct an anonymous game instance and derive the approximation factor using Theorem 3.10. □

4 SNE COMPUTATION

In this section, we investigate two problems on pure strong Nash equilibrium (SNE): recognizing whether an action profile is an SNE and computing an SNE (if it exists).

4.1 HARDNESS OF RECOGNIZING AN SNE

While PNE recognition is easy, for SNE recognition, we need to ensure that no coalition of agents has any incentive to deviate jointly to other actions given the actions of the other agents. Therefore, the standard brute-force method would require considering all possible coalitions and their actions, yielding a time complexity of $O(\sum_{i=1}^{n} \binom{n}{i} m^i) = O((m+1)^n)$. We show below that recognizing whether a given action profile is an SNE is co-NP-complete for special types of additive aggregate games. The reduction is from graphical games. Please see the Appendix for details.

Theorem 4.1. *It is co-NP-complete to recognize whether a given action profile is an SNE for an additive aggregate game with a constant number of actions for each player.*

To put the above hardness result in the context of known results, the SNE recognition problem is polynomial-time solvable for anonymous games, a subclass of additive aggregate games [Hoefer and Skopalik, 2013].

4.2 ALGORITHMS FOR RECOGNIZING AN SNE

Even in additive aggregate games with a constant number of actions for each agent, the standard brute-force method has a time complexity exponential in the number of agents. To overcome the computational challenge, we examine parameters that deal with the size of the aggregate space and present another aggregate-space algorithm to determine whether a given action profile is an SNE. The algorithm is efficient when the size of the aggregate space is bounded.

Our approach solves a key subproblem stated below: Given an action profile and an arbitrary aggregate, is there a coalition of agents who have incentives to jointly deviate to some other actions that result in the given aggregate?

Problem 4.2 (Profitable Coalition Deviation)**.** Given an additive aggregate game $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$, $\boldsymbol{a} \in \boldsymbol{A}$, and $y' \in Y$, determine if there exists a coalition $I \subseteq N$ of agents such that, for some joint deviation $\boldsymbol{a}'_I \neq \boldsymbol{a}_I$, $\phi(\boldsymbol{a}'_I, \boldsymbol{a}_{-I}) = y'$ and for each agent $i \in I$,

$$\pi_i(a'_i, \phi(\boldsymbol{a}'_I, \boldsymbol{a}_{-I})) > \pi_i(a_i, \phi(\boldsymbol{a}_I, \boldsymbol{a}_{-I})).$$

For a given action profile $\boldsymbol{a} \in \boldsymbol{A}$, we can use Problem 4.2 to check whether there exists a profitable coalition deviation for each $y \in Y$. If the answer is no for all $y \in Y$, then $\boldsymbol{a}$ is an SNE. Otherwise, $\boldsymbol{a}$ is not an SNE. Thus, we have the following result.

Theorem 4.3. *Given an additive aggregate game, suppose that Problem 4.2 can be solved in $O(\gamma)$. There is an $O(|Y|\gamma)$ algorithm for recognizing whether a given action profile is an SNE in additive aggregate games.*

We now present an efficient graph-based dynamic programming algorithm (Algorithm 2) for Problem 4.2 that is polynomial in the size of Y.

In Algorithm 2, given $\boldsymbol{a} \in \boldsymbol{A}$ and $y' \in Y$, there are two cases: (1) $\phi(\boldsymbol{a}) \neq y'$ and (2) $\phi(\boldsymbol{a}) = y'$. In the first case, we iteratively construct an $(n + 1)$-partite graph $G = (Y_0, Y_1, Y_2, ..., Y_n, E)$, where $Y_0 = \{(0, 0, ..., 0)\}$ and $Y_i \subseteq Y$ for $i = 1, ..., n$, and the edges in E only form between Y_{i-1} and Y_i for $i = 1, ..., n$. For $i = 1, ..., n$, an edge $(y_{i-1}, y_i) \in E$ if and only if (a) either $i = 1$ and $y_0 = (0, ..., 0)$, or $i > 1$ and there is an edge (y_{i-2}, y_{i-1}) for some $y_{i-2} \in Y_{i-2}$, and (b) $y_i = y_{i-1} + \phi_i(a'_i)$ for some $a'_i \in A_i^D(y') \subseteq A_i$ where $A_i^D(y') = \{a'_i \in A_i \setminus \{a_i\} \mid \pi_i(a_i, \phi(\boldsymbol{a})) < \pi_i(a'_i, y')\} \cup \{a_i\}$ is the set of actions of agent i resulting in a profitable deviation (under y') from a_i, plus agent i's original action a_i (indicating no deviation). There is a profitable coalition deviation with action profiles that map to y' if and only if there is a path from $y_0 \in Y_0$ to $y' \in Y_n$.

In the second case (i.e., $\phi(\boldsymbol{a}) = y'$), some agents can still deviate jointly to achieve higher utilities while having the same, original aggregate y'. To check for this, we run the above procedure n times with the following change. At each run $i = 1, 2, ..., n$, we force agent i to deviate by only considering $A_i^D(y')$ without allowing agent i to take the original action a_i. Therefore, if there is a profitable deviation coalition, it must contain one of the agents, and one of the runs will produce a graph that has a path from $y_0 \in Y_0$ to $y' \in Y_n$.

Algorithm 2 Determining if there exists a coalition of agents that can jointly deviate from a given action profile to some action profiles that map to a given aggregate (Problem 4.2)

Input: Additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$, action profile $\boldsymbol{a} \in \boldsymbol{A}$, and $y' \in Y$
Output: If there exists a coalition $I \subseteq N$ of agents such that, for some joint deviation $\boldsymbol{a}'_I \neq \boldsymbol{a}_I$, $\phi(\boldsymbol{a}'_I, \boldsymbol{a}_{-I}) = y'$, and each agent $i \in I$ obtains a higher utility.

Procedure A: $\phi(\boldsymbol{a}) \neq y'$

1: Let $G = (Y_0, ..., Y_n, E)$ be $(n+1)$-partite graph where $Y_0 = \{(0, 0, \cdots, 0)\}$ and $Y_i = \{\}$ for $i \in \{1, 2, \cdots, n\}$
2: **for** $i \in \{1, ..., n\}$ **do**
3: $\quad A_i^D(y') = \{a'_i \in A_i \setminus \{a_i\} \mid \pi_i(a_i, \phi(\boldsymbol{a})) < \pi_i(a'_i, y')\} \cup \{a_i\}$
4: $\quad$ **for** $y_{i-1} \in Y_{i-1}$ **do**
5: $\quad\quad$ **for** $a'_i \in A_i^D(y')$ **do**
6: $\quad\quad\quad$ Let $y_i = y_{i-1} + \phi_i(a'_i)$
7: $\quad\quad\quad$ $Y_i = Y_i \cup \{y_i\}$
8: $\quad\quad\quad$ $E = E \cup \{(y_{i-1}, y_i)\}$
9: **return** True [if $y' \in Y_n$] or False [otherwise]

Procedure B: $\phi(\boldsymbol{a}) = y'$

Run Procedure A n times. For each run, $i = 1, ..., n$, remove a_i in line 3. Return True if one of the runs is True. Otherwise, return False.

Algorithm 2 leads us to the following result. Please see the Appendix for the proof.

Theorem 4.4. *There is an $O(|Y|n^2m)$ constructive algorithm for determining if there exists a coalition of agents that can jointly deviate from a given action profile to some action profiles that map to a given aggregate (Problem 4.2).*

Combining Theorem 4.3 and Theorem 4.4 for Problem 4.2, we obtain the following result for additive aggregate games. The result implies that verifying an SNE is polynomial-time solvable for bounded $|Y|$.

Corollary 4.5. *There is an $O(|Y|^2n^2m)$ algorithm for determining whether a given action profile is an SNE in additive aggregate games.*

4.3 HARDNESS OF COMPUTING AN SNE

Given that the SNE recognition problem is already a provably hard problem (Theorem 4.1), determining the existence of an SNE is likely to be hard. As we show below, the SNE existence problem is indeed Σ_2^P-complete, which is at a higher level in the polynomial hierarchy than NP (for reference, NP $= \Sigma_1^P$ [Arora and Barak, 2009]).[1] The following theorem uses a reduction from graphical games.

Theorem 4.6. *It is Σ_2^P-complete to determine the existence of an SNE in additive aggregate games, even when agents have a constant number of actions.*

Contrast the above Σ_2^P-completeness result for additive aggregate games with the known NP-completeness result for anonymous games [Hoefer and Skopalik, 2013].

4.4 ALGORITHMS FOR COMPUTING AN SNE

The most straightforward brute-force algorithm for computing an SNE is to enumerate all of the action profiles and verify whether each of the action profiles is an SNE. There are $O(m^n)$ action profiles. Checking each action profile for SNE takes $O((m+1)^n)$ without using our algorithm (Theorem 4.4 and Corollary 4.5). The same checking takes $O(|Y|^2n^2m)$ using our algorithm. Therefore, the runtime can be $O(m^n(m+1)^n)$ or $O(m^n|Y|^2n^2m)$.

Given the Σ_2^P-completeness of additive aggregate games even with a bounded number of actions, it is provably hard to devise any reasonably efficient algorithms. Therefore, we next consider additional properties and develop efficient algorithms for bounded aggregate space.

4.4.1 Symmetric Additive Aggregate Games

Given an additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$, it is *symmetric* if and only if for each agent $i \in N$, $A_i = A$, and $\pi_i = \pi$, and $\phi_i = \phi_0$. Let the number of dimensions of $Y \subseteq \mathbb{Z}_{>0}^d$ be d. We define the *support* $(\phi_0(a)) = \{k \in \{1, ..., d\} \mid \phi_0(a)_k > 0\}$ to be the set of dimensions that action $a \in A$ contributes to. The following result allows us to reduce the factor of m^n in the brute-force approach to SNE existence.

Theorem 4.7. *Suppose that a symmetric additive aggregate game has the properties that $d \geq |A|$ and $support(\phi_0(a)) \cap support(\phi_0(a')) = \emptyset$ for all distinct $a, a' \in A$. There is an $O(|Y|^3n^2m)$ algorithm for determining the existence of an SNE and returning an SNE (if it exists).*

[1] Intuitively, NP asks the question, "Does there exist an action profile $\boldsymbol{a} \in \boldsymbol{A}$ that meets some condition verifiable in polynomial time?" In contrast, Σ_2^P asks the question, "Does there exist an action profile $\boldsymbol{a} \in \boldsymbol{A}$ such that *for all* possible coalitions $I \subseteq N$, some condition verifiable in polynomial time does not hold?"

Proof Sketch. For $y \in Y$, if y is feasible, we construct an action profile consistent with it and apply Theorem 4.4. □

4.4.2 Non-Increasing Additive Aggregate Games

Next, we consider additive aggregate games with non-increasing utility functions, where an agent's utility is only affected by the elements of aggregate that they affect (a property that appears in subclasses of aggregate games, such as congestion games). Given an additive aggregate game $(N, \{A_i, \pi_i\}_{i\in N}, \phi)$, we consider the dimension of $Y \subseteq \mathbb{Z}^d_{\geq 0}$ to be $d = |A|$, $A = A_i = \{1, 2, ..., d\}$ for all $i \in N$. This game is non-increasing if and only if for any $i \in N$ and any $a_i \in A_i$, $\pi_i(a_i, y') \leq \pi_i(a_i, y)$ whenever $y \leq y' \in Y$ (i.e., $y_j \leq y'_j$ for $j = 1, ..., d$). We have the following result (details and pseudocode are in the Appendix).

Theorem 4.8. *Suppose that a non-increasing additive aggregate game has the properties that $d = |A|$, for any $a_i \in A = \{1, ..., d\}$ and $i \in N$, support$(\phi_i(a_i)) = \{a_i\}$, and $\pi_i(a_i, y) = \pi_i(a_i, y')$ for any $y, y' \in Y$ in which $y_{a_i} = y'_{a_i}$. There is an $O(|Y|(nm^2 + |Y|nm + |Y|n))$ constructive algorithm for SNE existence.*

4.4.3 Significance of SNE Results

As mentioned in Section 1, there is little to no result in the literature on SNE for aggregate games. There are some results for singleton unweighted congestion games [Hoefer and Skopalik, 2013] and variants [Holzman and Law-Yone, 1997, Rozenfeld and Tennenholtz, 2006]. Notably, our model is more general than singleton unweighted congestion game because we capture potentially heterogeneous weights of an agent i for each action a_i using $\phi_i(a_i)$. Furthermore, SNE existence is guaranteed for singleton unweighted congestion games, which is not the case for us. Lastly, the co-NP-completeness of SNE recognition and Σ_2^P-completeness of SNE existence (even for a constant number of actions) provide new insights into these hard problems.

5 CONNECTION TO OTHER GAMES

We establish connections between additive aggregate games and various popular classes of games. We leave a brief discussion on Cournot games to the Appendix.

5.1 CONGESTION GAMES

A congestion game $(N, R, \{A_i, \rho_i\}_{i\in N}, \{c_r\}_{r\in R})$ consists of a set $N = \{1, \ldots, n\}$ of n agents and a set of resources R, a set of actions $A_i \subseteq 2^R \setminus \{\emptyset\}$ for each agent i, a cost function $c_r : \mathbb{N} \to \mathbb{R}$ for each resource r, and a cost function $\rho_i : A_1 \times ... \times A_n \to \mathbb{R}$ for each agent i defined as follows. Given an action profile $\boldsymbol{a}$, let $\boldsymbol{x}_r(\boldsymbol{a})$ be the number of agents selecting resource r under $\boldsymbol{a}$. We define $\rho_i(\boldsymbol{a}) = \sum_{r\in a_i} c_r(\boldsymbol{x}_r(\boldsymbol{a}))$. That is, the cost of agent i is the sum of the costs of the resources selected by i under $\boldsymbol{a}$.

Congestion games are a special class of additive aggregate games with $|Y| = n^{|R|}$. To see this, given any congestion game $(N, R, \{A_i, \rho_i\}_{i\in N}, \{c_r\}_{r\in R})$, we construct an additive aggregate game with the same set of agents N and the same set of actions A_i for each agent i. We define each agent i's aggregator function as $\phi_i(a_i) \equiv \sum_{r\in a_i} e_r$, where e_k is an $|R|$-dimensional unit vector of all zeros except a 1 at the k-th place. Therefore, for any action profile $\boldsymbol{a}$, $\phi(\boldsymbol{a}) = \phi_1(a_1) + \phi_2(a_2) + \cdots + \phi_n(a_n) = \boldsymbol{x}(\boldsymbol{a})$. Finally, the utility function of agent i in the additive aggregate game is defined as $\pi_i(a_i, \phi(\boldsymbol{a})) \equiv -\sum_{r\in a_i} c_r(\phi_r(\boldsymbol{a})) = -\rho_i(\boldsymbol{a})$.[2]

To connect the computational results in this paper to congestion games, the aggregate space Y in congestion games is an $|R|$-dimensional vector representing the number of agents selecting each resource. Therefore, $|Y| = n^{|R|}$. We, therefore, get the following main results from Corollaries 3.9 and 4.5, respectively. Below, m is the maximum number of actions, which can be much smaller than $2^{|R|}$.

Theorem 5.1. *There is an $O(n^{|R|}(nm^2 + n^{|R|}nm))$ algorithm for computing a PNE for congestion games. The algorithm runs in polynomial time for bounded number of resources.*

Theorem 5.2. *There is an $O(n^{2|R|}n^2m)$ algorithm for recognizing whether a given action profile is an SNE in congestion games. The algorithm runs in polynomial time for bounded number of resources.*

Similarly, variants of congestion games (e.g., weighted, singleton, etc.) can be shown to be special types of additive aggregate games. Please see [Irfan et al., 2024] for a detailed exposition of congestion games.

5.2 ANONYMOUS GAMES

An anonymous game $(N, A, \{\rho_i\}_{i\in N})$ consists of a set $N = \{1, \ldots, n\}$ of n agents and a common set of m actions $A = \{1, \ldots, m\}$ for all agents, and a payoff function $\rho_i : A \times Y \to \mathbb{R}$ for each agent i, where Y is the space of all m-dimensional vectors representing the number of agents selecting each action under an action profile.

Anonymous games are not a subclass of congestion games. Nor are congestion games a subclass of anonymous games. Interestingly, a special type of congestion game with singleton resources is a special type of anonymous game.

Anonymous games, however, are a subclass of additive aggregate games. Given an anonymous game $(N, A, \{\rho_i\}_{i\in N})$,

[2]The negative sign translates costs to payoffs to keep the solution concepts the same.

we construct an additive aggregate game with *symmetric actions*. It has a set of agents N, a set of actions A for each agent, and the same aggregate space Y mentioned in the previous paragraph. For the additive aggregate game, we define $\phi_i(a_i) \equiv e_{a_i}$, where e_k is an m-dimensional unit vector with a 1 only at the k-th place. Therefore, for any action profile $\boldsymbol{a}$, the aggregator function $\phi(\boldsymbol{a}) = \phi_1(a_1) + \phi_2(a_2) + \cdots + \phi_n(a_n)$ computes an m-dimensional vector representing the number of agents selecting each action under $\boldsymbol{a}$. The utility function of each agent i in the additive aggregate game is defined as $\pi_i(a_i, \phi(\boldsymbol{a})) \equiv \rho(a_i, \phi(\boldsymbol{a}))$.

In this paper, we have shown SNE recognition and computation problems are co-NP-complete and Σ_2^P-complete for additive aggregate games, respectively. *In contrast, these problems are in P and NP-complete, respectively, for anonymous games* [Hoefer and Skopalik, 2013]. Also, the size of the aggregate space $|Y| = n^m$ for anonymous games gives an intuition into why the literature often assumes the number of actions m to be constant in anonymous games [Hoefer and Skopalik, 2013].

5.3 SCHELLING GAMES

Thomas Schelling had famously introduced a dynamic model of segregation to capture social phenomena like residential segregation by race [Schelling, 1969, 1971]. In Elkind et al.'s Schelling game (SG) [Elkind et al., 2019], there is a set N of n agents. Each agent is one of $k \geq 2$ types. There is an undirected graph $G = (V, E)$ where $V = \{1, \ldots, m\}$ is the set of m location choices for each agent ($m > n$). Each location can hold at most one agent. Given an action profile $\boldsymbol{a} = (a_1, \ldots, a_n)$ denoting the location choices of the n agents, each agent i's utility is defined as $\rho_i(\boldsymbol{a}) = \frac{f_i(\boldsymbol{a})}{f_i(\boldsymbol{a})+e_i(\boldsymbol{a})}$. Here, $f_i(\boldsymbol{a})$ represents the number of agents in i's neighborhood in G who are of the same type as i and $e_i(\boldsymbol{a})$ denotes the number of neighboring agents of a different type.

We show that SGs are a subclass of additive aggregate games. Given an SG, we construct a *symmetric* additive aggregate game having a set of n agents with an associated type t_i for each agent i and a common set V of m actions (i.e., location choices). We define an $(m \times k)$-dimensional aggregate space Y representing the number of agents of each type at each location. For an action profile $\boldsymbol{a}$, the aggregator function $\phi(\boldsymbol{a}) = \phi_1(a_1) + \phi_2(a_2) + \cdots + \phi_n(a_n)$, where $\phi_i(a_i)$ is an $(m \times k)$-dimensional unit vector having a 1 only at index (a_i, t_i). The utility function of agent i in the additive aggregate game is $\pi_i(a_i, \phi(\boldsymbol{a})) \equiv \frac{\phi_{a_i,t_i}(\boldsymbol{a})}{\sum_{v \in N_{a_i}} \sum_{t' \neq t_i} \phi_{v,t'}(\boldsymbol{a})} = \rho_i(\boldsymbol{a})$. Here, N_v denotes the set of neighbors of node v in G. Here, $|Y| \leq n^{mk}$. Therefore, we obtain the following results.

Theorem 5.3. *There is an $O(n^{mk}(nm^2 + n^{mk+1}m))$ algorithm for checking if there exists a PNE (and computing a PNE if exists) for Schelling games. The algorithm runs in polynomial time for bounded number of locations and types.*

The above result should be put in the context of NP-completeness results for very special types of SGs [Elkind et al., 2019]. Our algorithm provides new insight into computing PNE in these provably hard games.

Theorem 5.4. *There is an $O(n^{2mk+2}m)$ algorithm for recognizing whether a given action profile is an SNE in SGs. The algorithm is polynomial time for bounded number of locations and types.*

Above is the first SNE result on SGs to our knowledge. This highlights the broad applicability of our technical results. Furthermore, there have been many recent studies on SGs [Echzell et al., 2019, Chauhan et al., 2018, Agarwal et al., 2020, Chan et al., 2020, Kanellopoulos et al., 2020, Kreisel et al., 2021, Bilò et al., 2022], most of which are additive aggregate games. Our algorithmic scheme gives a new way of approaching these games.

6 CONCLUSION

This paper contributes to the study of equilibrium computation in aggregate games. We have shown the hardness of PNE computation as well as SNE recognition and computation in additive aggregate games–a class of games we have defined. Notably, the known hardness results for subclasses of additive aggregate games like anonymous games do not imply hardness of other subclasses. In fact, we have shown that SNE recognition for very special subclassses of additive aggregate games is co-NP-complete, whereas the problem is polynomial-time solvable for anonymous games. Therefore, this study contributes to our knowledge of tractability for a widely applicable class of games.

On the algorithmic front, we have devised a polynomial-time algorithm for PNE computation in additive aggregate games with bounded aggregate space. We have extended ϵ-approximate PNE computation for λ-Lipschitz anonymous games to a broader class of games we call weighted anonymous games. We have also presented algorithms for SNE recognition and computation. The broad range of connections to well-studied classes of games makes this study particularly appealing for future applications.

This study has led to some interesting open problems, particularly in the space of games between additive aggregate games and anonymous games. Extending our definition of weighted anonymous games (see Section 3.3) and studying equilibrium computation in such games will further enhance our understanding of this space of games. Applying our computational approach to aggregate games in a network setting [Garg and Jaakkola, 2017] is another interesting future direction.

Acknowledgements

We thank the reviewers for their encouraging words and helpful suggestions. JS thanks the South Dakota Board of Regents for support from the Artificial Intelligence and Machine Learning grant. MTI is grateful to the National Science Foundation for support from Award IIS-1910203. HC is supported by the National Institute of General Medical Sciences of the National Institutes of Health (P20GM130461), the Rural Drug Addiction Research Center at the University of Nebraska-Lincoln, and the National Science Foundation under grant IIS-2302999 and IIS-2414554. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

Daron Acemoglu and Martin Kaae Jensen. Aggregate comparative statics. *Games and Economic Behavior*, 81: 27–49, 2013.

Heiner Ackermann, Heiko Röglin, and Berthold Vöcking. On the impact of combinatorial structure on congestion games. *Journal of the ACM (JACM)*, 55(6):1–22, 2008.

Aishwarya Agarwal, Edith Elkind, Jiarui Gan, and Alexandros A Voudouris. Swap stability in schelling games on graphs. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, 2020.

Eitan Altman, Thomas Boulogne, Rachid El-Azouzi, Tania Jiménez, and Laura Wynter. A survey on networking games in telecommunications. *Computers & Operations Research*, 33(2):286–311, 2006.

Sanjeev Arora and Boaz Barak. *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.

Robert J Aumann. Acceptable points in general cooperative n-person games. *Contributions to the Theory of Games*, 4:287–324, 1959.

Yakov Babichenko. Best-Reply Dynamic in Large Aggregative Games. SSRN Scholarly Paper ID 2210080, Social Science Research Network, Rochester, NY, January 2013. URL `https://papers.ssrn.com/abstract=2210080`.

Navin A. R. Bhat and Kevin Leyton-Brown. Computing nash equilibria of action-graph games. In *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, UAI '04, page 35–42, Arlington, Virginia, USA, 2004. AUAI Press. ISBN 0974903906.

Davide Bilò, Vittorio Bilò, Pascal Lenzner, and Louise Molitor. Topological influence and locality in swap schelling games. *Autonomous Agents and Multi-Agent Systems*, 36 (2):47, 2022.

Matthias Blonski. Characterization of pure strategy equilibria in finite anonymous games. *Journal of Mathematical Economics*, 34(2):225–233, 2000.

Felix Brandt, Felix Fischer, and Markus Holzer. Symmetries and the complexity of pure nash equilibrium. *Journal of Computer and System Sciences*, 75(3):163–177, 2009. ISSN 0022-0000. doi: https://doi.org/10.1016/j.jcss.2008.09.001. URL `https://www.sciencedirect.com/science/article/pii/S0022000008000846`.

Guilherme Carmona and Konrad Podczeck. Pure strategy nash equilibria of large finite-player games and their relationship to non-atomic games. *Journal of Economic Theory*, 187:105015, 2020.

Hau Chan, Mohammad T. Irfan, and Cuong Viet Than. Schelling models with localized social influence: A game-theoretic framework. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '20, page 240–248, Richland, SC, 2020. International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9781450375184.

Ankit Chauhan, Pascal Lenzner, and Louise Molitor. Schelling segregation with strategic agents. In *International Symposium on Algorithmic Game Theory*, pages 137–149. Springer, 2018.

Luis C Corchón. Comparative statics for aggregative games the strong concavity case. *Mathematical Social Sciences*, 28(3):151–165, 1994.

Richard Cornes and Roger Hartley. Fully aggregative games. *Economics Letters*, 116(3):631–633, 2012.

Rachel Cummings, Michael Kearns, Aaron Roth, and Zhiwei Steven Wu. Privacy and Truthful Equilibrium Selection for Aggregative Games. In Evangelos Markakis and Guido Schäfer, editors, *Web and Internet Economics*, Lecture Notes in Computer Science, pages 286–299, Berlin, Heidelberg, 2015. Springer. ISBN 978-3-662-48995-6. doi: 10.1007/978-3-662-48995-6_21.

Constantinos Daskalakis and Christos H Papadimitriou. Approximate nash equilibria in anonymous games. *Journal of Economic Theory*, 156:207–245, 2015.

Constantinos Daskalakis, Aranyak Mehta, and Christos Papadimitriou. Progress in approximate nash equilibria. In *Proceedings of the 8th ACM Conference on Electronic Commerce*, pages 355–358, 2007.

Juliane Dunkel and Andreas S Schulz. On the Complexity of Pure-Strategy Nash Equilibria in Congestion and Local-Effect Games. *Mathematics of Operations Research*, page 19, 2008.

Hagen Echzell, Tobias Friedrich, Pascal Lenzner, Louise Molitor, Marcus Pappik, Friedrich Schöne, Fabian Sommer, and David Stangl. Convergence and hardness of strategic schelling segregation. In *Web and Internet Economics (WINE)*, 2019.

Edith Elkind, Jiarui Gan, Ayumi Igarashi, Warut Suksompong, and Alexandros A. Voudouris. Schelling games on graphs. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*, pages 266–272. International Joint Conferences on Artificial Intelligence Organization, 7 2019. doi: 10.24963/ijcai.2019/38. URL `https://doi.org/10.24963/ijcai.2019/38`.

Amir Epstein, Michal Feldman, and Yishay Mansour. Strong equilibrium in cost sharing connection games. In *Proceedings of the 8th ACM conference on Electronic commerce*, pages 84–92, 2007.

Alex Fabrikant, Christos Papadimitriou, and Kunal Talwar. The complexity of pure Nash equilibria. In *Proceedings of the thirty-sixth annual ACM symposium on Theory of computing - STOC '04*, page 604, Chicago, IL, USA, 2004. ACM Press. ISBN 978-1-58113-852-8. doi: 10.1145/1007352.1007445. URL `http://portal.acm.org/citation.cfm?doid=1007352.1007445`.

Zubair Md Fadlullah, Yousuke Nozaki, Akira Takeuchi, and Nei Kato. A survey of game theoretic approaches in smart grid. In *2011 International Conference on Wireless Communications and Signal Processing (WCSP)*, pages 1–4. IEEE, 2011.

Michael R Garey and David S. Johnson. Complexity results for multiprocessor scheduling under resource constraints. *SIAM journal on Computing*, 4(4):397–411, 1975.

Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1979.

Vikas Garg and Tommi Jaakkola. Local aggregative games. *Advances in Neural Information Processing Systems*, 30, 2017.

Nicola Gatti, Marco Rocco, and Tuomas Sandholm. Strong nash equilibrium is in smoothed p. In *Proceedings of the 17th AAAI Conference on Late-Breaking Developments in the Field of Artificial Intelligence*, pages 29–31, 2013.

Georg Gottlob, Gianluigi Greco, and Francesco Scarcello. Pure nash equilibria: Hard and easy games. *Journal of Artificial Intelligence Research*, 24:357–406, 2005.

Laurent Gourves and Jérôme Monnot. On strong equilibria in the max cut game. In *International Workshop on Internet and Network Economics*, pages 608–615. Springer, 2009.

Ara Hayrapetyan, Éva Tardos, and Tom Wexler. The effect of collusion in congestion games. In *Proceedings of the thirty-eighth annual ACM symposium on Theory of computing*, pages 89–98, 2006.

Martin Hoefer and Alexander Skopalik. On the complexity of pareto-optimal nash and strong equilibria. *Theory of computing systems*, 53(3):441–453, 2013.

Ron Holzman and Nissan Law-Yone. Strong equilibrium in congestion games. *Games and economic behavior*, 21 (1-2):85–101, 1997.

Mohammad T. Irfan and Luis E. Ortiz. On influence, stable behavior, and the most influential individuals in networks: A game-theoretic approach. *Artificial Intelligence*, 215:79–119, 2014. ISSN 00043702. doi: 10.1016/j.artint.2014.06.004. URL `http://www.sciencedirect.com/science/article/pii/S0004370214000812`.

Mohammad T. Irfan, Hau Chan, and Jared Soundy. Computing nash equilibria in multidimensional congestion games. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems*, pages 2309–2311, 2024.

Mohammad Tanvir Irfan. *Causal Strategic Inference in Social and Economic Networks*. PhD thesis, Stony Brook University, Department of Computer Science, August 2013. `http://mtirfan.com/papers/Mohammad_Tanvir_Irfan_Dissertation.pdf`.

Elena Janovskaja. Equilibrium points in polymatrix games. *Lithuanian Mathematical Journal*, 8(2):381–384, 1968.

Martin Kaae Jensen. Aggregative games and best-reply potentials. *Economic theory*, 43(1):45–66, 2010.

Albert Xin Jiang, Kevin Leyton-Brown, and Navin AR Bhat. Action-graph games. *Games and Economic Behavior*, 71 (1):141–173, 2011.

Panagiotis Kanellopoulos, Maria Kyropoulou, and Alexandros A Voudouris. Modified schelling games. In *Algorithmic Game Theory: 13th International Symposium, SAGT 2020, Augsburg, Germany, September 16–18, 2020, Proceedings 13*, pages 241–256. Springer, 2020.

Michael Kearns and Yishay Mansour. Efficient nash computation in large population games with bounded influence. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, UAI'02, page 259–266, San Francisco, CA, USA, 2002. Morgan Kaufmann Publishers Inc. ISBN 1558608974.

Michael Kearns, Michael L. Littman, and Satinder Singh. Graphical models for game theory. In *Proceedings of* *the Seventeenth Conference on Uncertainty in Artificial Intelligence*, UAI'01, page 253–260, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc. ISBN 1558608001.

Jayash Koshal, Angelia Nedić, and Uday V Shanbhag. Distributed algorithms for aggregative games on graphs. *Operations Research*, 64(3):680–704, 2016.

Luca Kreisel, Niclas Boehmer, Vincent Froese, and Rolf Niedermeier. Equilibria in schelling games: Computational hardness and robustness. *arXiv preprint arXiv:2105.06561*, 2021.

David Martimort and Lars Stole. Aggregate representations of aggregate games. Working paper, Munich Personal RePEc Archive (MPRA), 2011. URL https://mpra.ub.uni-muenchen.de/32871/.

David Martimort and Lars Stole. Representing equilibrium aggregates in aggregate games with applications to common agency. *Games and Economic Behavior*, 76(2): 753–772, 2012.

Rabia Nessah and Guoqiang Tian. On the existence of strong nash equilibria. *Journal of Mathematical Analysis and Applications*, 414(2):871–885, 2014. ISSN 0022-247X. doi: https://doi.org/10.1016/j.jmaa.2014.01.030. URL https://www.sciencedirect.com/science/article/pii/S0022247X14000377.

Luis E Ortiz and Michael Kearns. Nash propagation for loopy graphical games. *Advances in neural information processing systems*, 15, 2002.

Robert W. Rosenthal. A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, 2(1):65–67, December 1973. ISSN 0020-7276, 1432-1270. doi: 10.1007/BF01737559. URL http://link.springer.com/10.1007/BF01737559.

Ola Rozenfeld and Moshe Tennenholtz. Strong and correlated strong equilibria in monotone congestion games. In *Internet and Network Economics: Second International Workshop, WINE 2006, Patras, Greece, December 15-17, 2006. Proceedings 2*, pages 74–86. Springer, 2006.

Thomas C Schelling. Models of segregation. *The American Economic Review*, 59(2):488–493, 1969.

Thomas C. Schelling. Dynamic models of segregation. *Journal of Mathematical Sociology*, 1(2):143 – 186, 1971. ISSN 0022250X.

Reinhard Selten. Preispolitik der mehrproduktenunternehmung in der statischen theorie. *Ökonometrie und Unternehmensforschung/Econometrics and Operations Research*, 1970.

Berthold Vöcking and R Aachen. Congestion games: Optimization in competition. In *ACiD*, pages 9–20, 2006.

Jian Wang, Srinivas Peeta, and Xiaozheng He. Multiclass traffic assignment model for mixed traffic flow of human-driven vehicles and connected and autonomous vehicles. *Transportation Research Part B: Methodological*, 126: 139–168, 2019.

Koji Yamamoto. A comprehensive survey of potential game approaches to wireless networks. *IEICE Transactions on Communications*, 98(9):1804–1823, 2015.

Gioele Zardini, Nicolas Lanzetti, Laura Guerrini, Emilio Frazzoli, and Florian Dörfler. Game theory to study interactions between mobility stakeholders. In *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, pages 2054–2061. IEEE, 2021.

Pure and Strong Nash Equilibrium Computation in Compactly Representable Aggregate Games (Supplementary Material)

Jared Soundy[1] **Mohammad T. Irfan**[2] **Hau Chan**[3]

[1] The Beacom College of Computer & Cyber Sciences, Dakota State University, Madison, South Dakota, USA
[2] Department of Computer Science, Bowdoin College, Brunswick, Maine, USA
[3] School of Computing, University of Nebraska-Lincoln, Lincoln, Nebraska, USA

1 INTRODUCTION

COMPACT REPRESENTATION IN CONTEXT

We situate additive aggregate games within the broad and growing classes of compactly representable games. We begin with an exercise on representation size. In a general aggregate game with n players and m actions, given everyone's choices of actions $\boldsymbol{a} = (a_1, \cdots, a_n)$, the utility $\pi_i(a_i, \phi(\boldsymbol{a}))$ of agent i depends on a_i and an aggregator function $\phi(\boldsymbol{a})$ that is common for all agents. In an additive aggregate game, $\phi(\boldsymbol{a}) = \phi_1(a_1) + \phi_2(a_2) + \cdots + \phi_n(a_n)$. Representing $\phi_i(a_i)$ for all i and a_i takes $O(nm)$ space in tabular form or $O(n)$ space in parametric functional form, leading to the same representation size for the game. We make the standard assumption that the utility functions are given implicitly via value oracles or parameterized functions [Rosenthal, 1973, Daskalakis and Papadimitriou, 2015].

One of the most widely studied classes of compact games is *graphical games* from UAI'01 [Kearns et al., 2001]. In graphical games, the utility of an agent depends on their own action and the actions chosen by their neighbors. Aggregate games do not contain graphical games, even if the underlying graph is complete. On the other hand, although graphical games do not have an aggregator function, it is possible to define a graphical game on a complete graph with a special type of utility function that captures the aggregate-game utility function. However, doing so loses the computational appeal of graphical games (e.g., TreeNash [Kearns et al., 2001] and NashProp [Ortiz and Kearns, 2002]).

The additive decomposition used in additive aggregate games may ring a bell with *polymatrix games* [Janovskaja, 1968], where agent i's utility $u_i(\boldsymbol{a})$ is the sum of partial utilities $u_{ij}(a_i, a_j)$ from other agents j. Since the aggregator function $\phi(\boldsymbol{a})$ is the same for all players, additive aggregate games do not contain polymatrix games. More interestingly, the additively decomposable utility of polymatrix games cannot represent the aggregate game utility function of the shape $\pi_i(a_i, \phi(\boldsymbol{a}))$, even when $\phi(\boldsymbol{a})$ is additively decomposable. So, polymatrix games do not contain additive aggregate games either.

Yet another widely applicable class of compact games is *action graph games* (AGGs) from UAI'04 [Bhat and Leyton-Brown, 2004]. In an AGG, there is a graph among the actions. The utility of an agent is a function of their action and the number of agents choosing the neighboring actions. AGGs do contain aggregate games but only by having a complete action graph and distinct actions for agents, thereby losing the algorithmic features [Jiang et al., 2011].

Additive aggregate games do contain many classes of compact games, such as congestion games, anonymous games, Schelling games, and Cournot games. We explore the computational implications of this containment in Section 5.

3 PNE COMPUTATION

Theorem 3.1. *It is strongly NP-complete to decide PNE existence in additive aggregate games,*

even when the number of actions is linear in the number of agents, the dimension of the aggregate space is linear in the number of agents, and the utility function of each agent returns two integer values.

Proof. First, it is not hard to see that we can verify whether a given action profile is a PNE in polynomial time. Therefore, the problem is in NP.

Next, we show that the problem is strongly NP-hard by reducing from the 3-Partition problem, which is known to be strongly NP-complete [Garey and Johnson, 1975].

In an instance of the 3-Partition problem, we are given a multiset $X = \{x_1, ..., x_{n_{3P}}\}$ of n_{3P} integers. The 3-Partition problem seeks to determine whether X can be partitioned into $m = \frac{n_{3P}}{3}$ sets, each set of size 3, such that each set has the same sum $T = \frac{\sum_{x \in X} x}{m}$.[1] We further assume that each integer in X is strictly between $T/4$ and $T/2$.

We reduce the 3-Partition instance to an instance of additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ as follows:

- Let $n = n_{3P} + 2$ agents;
- For agent $i \in N \setminus \{n_{3P}+1, n_{3P}+2\}$, let $A_i = \{1, 2, \cdots, m\}$. For agent $i \in \{n_{3P}+1, n_{3P}+2\}$, let $A_i = \{m+1, m+2\}$;
- Let $Y \subset \{0, 1, .., mT\}^{m+2}$ where each aggregate $y \in Y$ is an $(m+2)$-dimensional vector of $m+2$ integers;
- Let the aggregator function be additively separable $\phi(a_1, \cdots, a_n) = \phi_1(a_1) + \cdots + \phi_n(a_n)$ for some function $\phi_i : A_i \to Y$ for each $i \in N$ where
  - For agent $i \in N \setminus \{n_{3P}+1, n_{3P}+2\}$, $\phi_i(a_i) = x_i e_{a_i}$ where e_k is an $(m+2)$-dimensional unit vector of all zeros except the kth dimension;
  - For agent $i \in \{n_{3P}+1, n_{3P}+2\}$, $\phi_i(a_i) = e_{a_i}$;
- For agent $i \in N \setminus \{n_{3P}+1, n_{3P}+2\}$, for each $a_i \in A_i$, and $y \in Y$, we define

$$\pi_i(a_i, y) = \begin{cases} 1 & y \in \{(T, T, \cdots, T, 1, 1), \\ & (T, T, \cdots, T, 2, 0), \\ & (T, T, \cdots, T, 0, 2)\} \\ -1 & \text{otherwise.} \end{cases}$$

- For agent $i = n_{3P} + 1$, $a_i \in A_i$, $y \in Y$, and $X_1, ..., X_m$ not all equal to T, we define[2]

$$\pi_{n_{3P}+1}(a_i, y) = \begin{cases} 1 & y \in \{(T, T, \cdots, T, 1, 1), \\ & (T, T, \cdots, T, 2, 0), \\ & (T, T, \cdots, T, 0, 2)\} \\ 1 & y \in \{(X_1, X_2, \cdots, X_m, 2, 0), \\ & (X_1, X_2, \cdots, X_m, 0, 2)\} \\ -1 & y = (X_1, X_2, \cdots, X_m, 1, 1) \end{cases}$$

- For agent $i = n_{3P} + 2$, $a_i \in A_i$, $y \in Y$, and $X_1, ..., X_m$ not all equal to T, we define

$$\pi_{n_{3P}+2}(a_i, y) = \begin{cases} 1 & y \in \{(T, T, \cdots, T, 1, 1), \\ & (T, T, \cdots, T, 2, 0), \\ & (T, T, \cdots, T, 0, 2)\} \\ 1 & y = (X_1, X_2, \cdots, X_m, 1, 1) \\ -1 & y \in \{(X_1, X_2, \cdots, X_m, 2, 0), \\ & (X_1, X_2, \cdots, X_m, 0, 2)\} \end{cases}$$

The utility of each agent i can be defined implicitly based on π_i as in Definition 2.1, allowing the reduction to take place in polynomial time. Given the above construction, we show the following implications to complete our proof.

[1] Without loss of generality, we assume n_{3P} to be a multiple of 3 and $\sum_{x \in X}$ to be a multiple of m.

[2] As mentioned in the Preliminaries section, π_i's are specified as parameterized functions (i.e., by $m+2$ variables corresponding to the dimension of Y and the agent's action), as opposed to a matrix or table.

3-Partition Problem solution $\implies$ PNE. Given a solution to the 3-Partition instance, we show that the solution can be mapped to a PNE of the constructed aggregate game instance. In such a solution, we have m sets, $S_1, S_2, ..., S_m$, each of size 3 and sum to T. To construct an action profile, for each S_j, $j = 1, ..., m$, we let $a_i = j$ be the action of agent i corresponding to $x_i \in S_j$. For agent $i \in \{n_{3P} + 1, n_{3P} + 2\}$, we set $a_i \in \{m + 1, m + 2\}$ to be any one of the two actions. It is not hard to see that such an action profile is a PNE as the first m dimensions of y of the action profile sum to T (due to the definition of ϕ where each agent i contributes x_i to a_ith entry only) and each agent $i \in N$ receives a utility of 1 (any unilateral deviation will not obtain utility higher than 1) by our construction.

PNE $\implies$ 3-Partition Problem solution. Given a PNE, we show that it can be mapped to a solution of the 3-Partition instance. Let $\boldsymbol{a}^* = (a_1^*, ..., a_n^*)$ be a PNE of the aggregate game instance. First, note that, in any PNE, the first m dimensions of aggregate $\phi(\boldsymbol{a}^*)$ must all have the value of T. Otherwise, the last two agents will create a situation in the game in which a PNE will not exist (i.e., the last two agents have exclusively two distinct actions from other agents contributing to only the last two dimensions of the aggregate and induce a subgame without any PNE unless the first mth dimensions have the value of T). We can construct a solution to the 3-Partition instance by defining the m sets, $S_1, S_2, ..., S_m$, to be $S_j = \{x_i \mid a_i^* = j, i \in N \setminus \{n_{3P+1}, n_{3P+2}\}\}$ for $j = 1, ..., m$. By construction, each S_j sums up to T as agent i with action j contributes x_i to only the jth dimension (due to the construction of ϕ) and has the size of 3 (as each x_i is in between $T/4$ and $T/2$). □

Theorem 3.2. *It is NP-complete to decide PNE existence in additive 2-action aggregate games, even when*

the dimension of the aggregate space is constant

and the utility function of each agent returns two integer values.

Proof. It is not hard to see that we can verify whether a given action profile is a PNE in polynomial time. Therefore, the problem is in NP.

We now show that the problem is NP-hard by reducing from the Partition problem, which is known to be NP-complete [Garey and Johnson, 1979].

Given a multiset $X = \{x_1, ..., x_{n_P}\}$ of n_P distinct positive integers, the Partition problem asks whether it is possible to partition X into sets S_1 and S_2 such that $\sum_{x \in S_1} x = \sum_{x \in S_2} x$. Without loss of generality, we assume $\sum_{x \in X} x = 2T$.

We reduce the Partition instance to an instance of additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ as follows:

- Let $n = n_P + 2$ agents;
- For agent $i \in N \setminus \{n_P + 1, n_P + 2\}$, let $A_i = \{1, 2\}$. For agent $i \in \{n_P + 1, n_P + 2\}$, let $A_i = \{3, 4\}$;
- Let $Y \subset \{0, 1, .., 2T\}^4$ where each aggregate $y \in Y$ is a 4-dimensional vector of 4 integers;
- Let aggregator function be additively separable $\phi(a_1, \cdots, a_n) = \phi_1(a_1) + \cdots + \phi_n(a_n)$ for some function $\phi_i : A_i \to Y$ for each $i \in N$ where
  - For agent $i \in N \setminus \{n_P + 1, n_P + 2\}$, $\phi_i(a_i) = x_i e_{a_i}$ where e_k is a 4-dimensional unit vector of all zero except the kth dimension;
  - For agent $i \in \{n_P + 1, n_P + 2\}$, $\phi_i(a_i) = e_{a_i}$;
- For agent $i \in N \setminus \{n_P + 1, n_P + 2\}$, for each $a_i \in A_i$, and $y \in Y$, we define

$$\pi_i(a_i, y) = \begin{cases} 1 & y \in \{(T, T, 1, 1), \\ & (T, T, 2, 0), (T, T, 0, 2)\} \\ -1 & \text{otherwise.} \end{cases}$$

- For agent $i = n_P + 1$, $a_i \in A_i$, $y \in Y$, and X_1 and X_2 not all equal to T, we define[3]

$$\pi_{n_P+1}(a_i, y) = \begin{cases} 1 & y \in \{(T,T,1,1), \\ & (T,T,2,0),(T,T,0,2)\} \\ 1 & y \in \{(X_1,X_2,2,0), \\ & (X_1,X_2,0,2)\} \\ -1 & y = (X_1,X_2,1,1) \end{cases}$$

- For agent $i = n_P + 2$, $a_i \in A_i$, $y \in Y$, and X_1 and X_2 not all equal to T, we define

$$\pi_{n_P+2}(a_i, y) = \begin{cases} 1 & y \in \{(T,T,1,1), \\ & (T,T,2,0),(T,T,0,2)\} \\ 1 & y = (X_1,X_2,1,1) \\ -1 & y \in \{(X_1,X_2,2,0), \\ & (X_1,X_2,0,2)\} \end{cases}$$

The utility of each agent i can be defined implicitly based on π_i as in Definition 2.1. Given the above construction, we show the following implications to complete our proof.

Partition Problem solution $\implies$ PNE. Given a solution to the Partition instance, we show that the solution can be mapped to a PNE of the constructed aggregate game instance. In such a solution, we will have two sets, S_1 and S_2, that each sum to T. We construct an action profile as follows. For each S_j, $j \in \{1, 2\}$, we let $a_i = j$ be the action of agent i corresponding to $x_i \in S_j$. For agent $i \in \{n_P + 1, n_P + 2\}$, we set $a_i \in \{3, 4\}$ to be any one of the two actions.

Due to the definition of ϕ where each agent i contributes x_i only to the a_ith dimension, this action profile ensures that the first two dimensions of the aggregate y sum to T. Therefore, all agents receive the maximum utility of 1, and no agent can increase their utility by deviating unilaterally. Hence, the action profile is a PNE.

PNE $\implies$ Partition Problem solution. Now, we show that a PNE can be mapped to a solution of the Partition instance. Let $\boldsymbol{a}^* = (a_1^*, \cdots, a_n^*)$ be a PNE of the aggregate game instance. First, observe that in any PNE, the first two dimensions of the aggregate $\phi(\boldsymbol{a}^*)$ must both be equal to T. If not, the last two agents create a situation where no PNE will exist. We can construct a solution to the Partition problem instance as follows. We define two sets, S_1 and S_2, where $S_j = \{x_i \mid a_i^* = j, i \in N\}$ for $j \in \{1, 2\}$. Each S_j sums up to T as agent i with action j contributes x_i to only the jth dimension. $\square$

Theorem 3.6. *Given an additive aggregate game, suppose that Problem 3.3, Problem 3.4, and Problem 3.5 can be solved in $O(\alpha)$, $O(nm^2\alpha)$, and $O(\beta)$, respectively. There is an $O(|Y|(nm^2\alpha + \beta))$ algorithm for determining the existence of a PNE and returning a PNE (if it exists) for additive aggregate games.*

Proof. The algorithm starts by considering each $y \in Y$. For each $y \in Y$, we would like to check to see if y corresponds to an action profile $\boldsymbol{a}^* \in \boldsymbol{A}$ that is a PNE. We decompose the algorithm into the following two steps.

Step 1. For each agent $i \in N$, we compute $BR_i(y)$ the set of aggregate best-response actions given y (Problem 3.4).

This says that, under the aggregate $y \in Y$, for each $a_i \in BR_i(y)$, agent i would have no incentive to deviate another actions. It is not hard to see that, since there are at most m actions for each agent and `deviate` (Problem 3.3) can be computed in $O(\alpha)$, constructing the set of aggregate best-response actions for each agent takes $O(m^2\alpha)$. Since there are n agents, we require $O(nm^2\alpha)$ in this step.

Step 2. Given the $BR_i(y)$ for each agent i, we then check to see if we can use the sets of aggregate best-response actions to construct an action profile in which it yields the aggregate y (Problem 3.5). That is, determine if there exists

[3] As mentioned in Section 2, π_i's are specified as parameterized functions (i.e., by four variables corresponding to the dimension of Y and the agent's action), as opposed to a matrix or table.

$\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i \in N} BR_i(y)$ such that $\phi(\tilde{\boldsymbol{a}}) = y$. If there is such a profile $\tilde{\boldsymbol{a}}$, then $\tilde{\boldsymbol{a}}$ is a PNE. Indeed, we can verify that, for each $i \in N$, we have, for any $a'_i \in A_i \setminus \{\tilde{a}_i\}$,

$$\begin{aligned}
&\pi_i(\tilde{a}_i, \phi(\tilde{a}_i, \tilde{\boldsymbol{a}}_{-i})) = \pi_i(\tilde{a}_i, y) \\
&\geq \pi_i(a'_i, \texttt{deviate}(\tilde{a}_i, y, a'_i)) \\
&= \pi_i(a'_i, (y - \phi_i(\tilde{a}_i)) + \phi_i(a'_i)) \\
&= \pi_i(a'_i, (\sum_{j \in N \setminus \{i\}} \phi_j(\tilde{a}_j) + \phi_i(a'_i)) = \pi_i(a'_i, \phi(a'_i, \tilde{\boldsymbol{a}}_{-i})),
\end{aligned}$$

where the first equality is by the fact that $\phi(\tilde{\boldsymbol{a}}) = y$, the first inequality is by the definition of best-response action, the second equality is by definition, and the third and fourth equalities are by using the additive separable property.

Because we consider each $y \in Y$ and each $\boldsymbol{a} \in A$ is mapped to some $y \in Y$ under ϕ, we can determine if there is a PNE. The total running time is $O(|Y|(nm^2\alpha + \beta))$. □

Theorem 3.7. *It is strongly NP-complete to determine the existence of an action profile that maps to a given aggregate from given sets of actions of each agent in additive aggregate games (Problem 3.5).*

Proof. First, we note that, because the aggregator function is additive, for a given action profile $\boldsymbol{a} = (a_1, ..., a_n)$, we have that $\phi(\boldsymbol{a}) = \sum_{i \in N} \phi_i(a_i)$. Therefore, we can easily check whether a potential solution $\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i \in N} \tilde{A}_i$ can be mapped to a given y efficiently. Therefore, the problem is in NP.

To show that the problem is strongly NP-complete, we reduce from the 3-Partition problem, which is known to be strongly NP-complete [Garey and Johnson, 1975].

In an instance of the 3-Partition problem, we are given a multiset $X = \{x_1, ..., x_{n_{3P}}\}$ of n_{3P} integers. The 3-Partition problem seeks to determine whether X can be partitioned into $m = \frac{n_{3P}}{3}$ sets, each set of size 3, such that each set has the same sum $T = \frac{\sum_{x \in X} x}{m}$.[4] We further assume that each integer in X is strictly between $T/4$ and $T/2$.

We reduce the 3-Partition instance to an instance of Problem 3.5 with additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ and input y as follows:

- Let $n = n_{3P}$;
- For each agent $i \in N$, let $A_i = \tilde{A}_i = \{1, ..., m\}$;
- Let $Y \subseteq \{0, 1, ..., mT\}^m$ and $y = (T, ..., T)$ an m-dimensional vector with value of T for each dimension.
- Let the aggregator function be additively separable $\phi(\tilde{a}_1, \cdots, \tilde{a}_n) = \phi_1(\tilde{a}_1) + \cdots + \phi_n(\tilde{a}_n)$ for some function $\phi_i : \tilde{A}_i \to Y$ for each $i \in N$ where
  - For agent $i \in N$, $\phi_i(a_i) = x_i e_{a_i}$ where e_k is an (m)-dimensional unit vector of all zero except the kth dimension;

The utility function (i.e., π_i) of each agent can be defined arbitrarily (as they are not crucial for Problem 3.5).

3-Partition Problem solution $\implies$ Problem 3.5 solution.

Given a solution to the 3-Partition instance, we show that the solution can be mapped to a solution of the constructed Problem 3.5 instance. In such a solution, we have m sets, $S_1, S_2, ..., S_m$, each of size 3 and sum to T. To construct an action profile, for each S_j, $j = 1, ..., m$, we let $a_i = j$ be the action of agent i corresponding to $x_i \in S_j$ to be j. It is not hard to see that such an action profile is a solution to the Problem 3.5 instance as the m dimensions of y of the action profile sum to T (due to the definition of ϕ where each agent i contributes x_i to a_ith entry only) by our construction.

Problem 3.5 solution $\implies$ 3-Partition Problem solution. Given a solution of the Problem 3.5 instance, we show that it can be mapped to a solution of the 3-Partition instance. Let $\tilde{\boldsymbol{a}} = (\tilde{a}_1, ..., \tilde{a}_n) \in \prod_{i \in N} \tilde{A}_i$ be a solution such that $\phi(\tilde{\boldsymbol{a}}) = \sum_{i \in N} \phi_i(\tilde{a}_i) = y = (T, ..., T)$. We can construct a solution to the 3-Partition instance by defining the m sets,

[4] Without loss of generality, we assume n_{3P} to be a multiple of 3 and $\sum_{x \in X}$ to be a multiple of m.

$S_1, S_2, ..., S_m$, to be $S_j = \{x_i \mid \tilde{a}_i = j, i \in N\}$ for $j = 1, ..., m$. By construction, each S_j sums up to T as agent i with action j contributes x_i to only the jth dimension (due to the construction of ϕ) and has the size of 3 (as each x_i is in between $T/4$ and $T/2$). □

Theorem 3.11. *Any λ-Lipschitz weighted anonymous game with payoffs in $[0, 1]$ has an $O(mw\lambda)$-approximate PNE.*

Proof. Given a λ-Lipschitz weighted anonymous game instance with aggregate space Y, payoff function ρ_i for any $i \in N$ and any $y \in Y$, we first construct an anonymous games with aggregate space Y' and payoff functions ρ'_i as follows: define $\rho'_i(a_i, y/w) \equiv \rho_i(a_i, y)/w$. Therefore, whenever $y \in Y$, $y/w \in Y'$. The newly constructed anonymous game is λ-Lipschitz because for any $x', y' \in Y'$, $|\rho'_i(a_i, x') - \rho'_i(a_i, y')| = |\rho_i(a_i, wx') - \rho_i(a_i, wy')|/w \leq \lambda \cdot ||wx' - wy'||_{L_1}/w = \lambda \cdot ||x' - y'||_{L_1}$.

Theorem 3.10 gives us an $O(m\lambda)$-approximate PNE for the anonymous game instance. We can translate it to an $O(mw\lambda)$-approximate NE for the corresponding weighted anonymous game instance using Definition 2.6

$\rho_i(a_i, y) = w\rho'_i(a_i, y/w)$. □

4 SNE COMPUTATION

Theorem 4.1. *It is co-NP-complete to recognize whether a given action profile is an SNE for an additive aggregate game with a constant number of actions for each player.*

Proof. To show that the problem is in co-NP, we first argue that determining whether a given action profile is not an SNE is in NP. For a given action profile $\boldsymbol{a}$ to be not an SNE, we consider a "no" certificate. Such a no certificate is specified by another given action profile $\boldsymbol{a'} \neq \boldsymbol{a}$, capturing the deviation of a coalition of one or more players. We can verify in polynomial time whether the coalition of agents deviating from $\boldsymbol{a}$ to $\boldsymbol{a'}$ are incentivized to do so. That is, for each agent i such that $a_i \neq a'_i$, we can verify in polynomial time whether $\pi_i(a'_i, \phi(\boldsymbol{a'})) > \pi_i(a_i, \phi(\boldsymbol{a}))$.

Next, we show that the problem is co-NP-hard by reducing the problem of determining whether a given action profile is an SNE in graphical games, which is shown to be co-NP-complete even when each agent has at most three neighbors and a fixed number of actions [Gottlob et al., 2005].

In a graphical game instance $G = (V, E)$, $\{\bar{A}_i, \bar{u}_i\}_{i \in V}$, we have a set $V = \{1, ..., \bar{n}\}$ of $\bar{n}$ agents, a set E of edges between the agents in V, a set $\bar{A}_i$ of actions for each agent i, and a utility function $\bar{u}_i : \bar{A}_{N(i) \cup \{i\}} \to \mathbb{R}$ for each agent i that maps the actions of i's neighbors $N(i)$ in G and i to some real numbers. We consider instances where $|N(i)| \leq 3$ and $|A_i| \leq C$ (for a constant C) for each agent i. Given a graphical game, it is co-NP-complete to recognize whether an action profile $\bar{\boldsymbol{a}}' \in \bar{\boldsymbol{A}}$ is an SNE.

We reduce the above problem instance to an instance of the corresponding problem for additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ and input action profile $\boldsymbol{a}$ as follows:

- Let $n = \bar{n}$;
- For each $i \in N$, let $A_i = \bar{A}_i$. We relabel the actions as $A_i = \{1, 2, .., |\bar{A}_i|\}$ via an arbitrary one-to-one and onto mapping function $f_i(\bar{a}_i) = a_i$ for $\bar{a}_i \in \bar{A}_i$ and $a_i \in A_i$;
- Let $\boldsymbol{a}' = \bar{\boldsymbol{a}}'$;
- Let $Y = A_1 \times ... \times A_n$ where each aggregate $y \in Y$ is an n-dimensional vector of n integers;
- Let aggregator function be additively separable $\phi(a_1, \cdots, a_n) = \phi_1(a_1) + \cdots + \phi_n(a_n)$ for some function $\phi_i : A_i \to Y$ for each $i \in N$ where
  - For agent $i \in N$, $\phi_i(a_i) = a_i e_i$ where e_k is a n-dimensional unit vector of all zero except the kth dimension;
- For agent $i \in N$, for each $a_i \in A_i$, and $y \in Y$, we define

$$\pi_i(a_i, y) = \bar{u}_i(f_i^{-1}(a_i), (f_j^{-1}(y_j))_{j \in N(i)})$$

  where f_i^{-1} is the inverse of f_i.

The utility of each agent i can be defined implicitly based on π_i as in Definition 2.1. While the representation of the graphical games is explicit (where utility is represented using tables or matrices), the construction takes polynomial time because the number of actions for each agent, the number of neighbors is bounded by 3, and π_i's are specified as parameterized functions. Moreover, the utility functions of agents of the two games are equivalent. That is, for any agent i and $\boldsymbol{a} = (a_i, \boldsymbol{a}_{-i}) \in \boldsymbol{A}$, $\pi_i(a_i, \phi(\boldsymbol{a})) = \pi_i(a_i, y) = \bar{u}_i(f_i^{-1}(a_i), (f_j^{-1}(y_j))_{j \in N(i)}) = \bar{u}_i(\bar{a}_i, \bar{\boldsymbol{a}}_{N(i)})$ where f is one-to-one and onto. Therefore, $\boldsymbol{a}'$ is an SNE in the aggregate game instance if and only if $\bar{\boldsymbol{a}}'$ is an SNE in the graphical game instance. □

Theorem 4.4. *There is an $O(|Y|n^2m)$ constructive algorithm for determining if there exists a coalition of agents that can jointly deviate from a given action profile to some action profiles that map to a given aggregate (Problem 4.2).*

Proof. The running times of Procedure A and Procedure B of Algorithm 2 are $O(|Y|nm)$ and $O(|Y|n^2m)$, respectively. (Algorithm 2 is presented in the main text.)

In the case of $\phi(\boldsymbol{a}) \neq y'$, if there is a path from $y_0 \in Y_0$ to $y' \in Y$, then there must be an action profile that can be mapped or summed to y' by the graph construction in which each agent i either takes their original action a_i or another action in $A_i^D(y')$ that can obtain a strictly better utility under y'. Because $\phi(\boldsymbol{a}) \neq y'$, if $y' \in Y_n$, it must be the case that not all agents have retained their original actions, and some agents have benefited by deviating.

In the case of $\phi(\boldsymbol{a}) = y'$, at each run $i = 1, ..., n$, we force agent i to deviate. If there is a path from $y_0 \in Y_0$ to $y' \in Y_n$, then it must be the case that there is a profitable deviation for agent i and possibly for other agents. Because we force each agent to deviate at least once, if there is a profitable deviation coalition, one of the runs must produce a graph that has a path from $y_0 \in Y$ to $y' \in Y_n$.

An exact action profile deviating from the input action profile can be extracted in polynomial time via the standard method of tracing back. □

Theorem 4.6. *It is Σ_2^P-complete to determine the existence of an SNE in additive aggregate games, even when agents have a constant number of actions.*

Proof. To show Σ_2^P membership, we first note that, given an action profile, we can recognize whether it is an SNE (which is shown to be in co-NP in Theorem 4.1). Therefore, the problem is in Σ_2^P.

Following the construction of Theorem 4.1, we can show that the problem is Σ_2^P-hard by reducing the problem of determining the existence of an SNE in graphical games, which is shown to be Σ_2^P-complete even when each agent has at most three neighbors and a fixed number of actions [Gottlob et al., 2005]. □

Theorem 4.7. *Suppose that a symmetric additive aggregate game has the properties that $d \geq |A|$ and $support(\phi_0(a)) \cap support(\phi_0(a')) = \emptyset$ for all distinct $a, a' \in A$. There is an $O(|Y|^3n^2m)$ algorithm for determining the existence of an SNE and returning an SNE (if it exists).*

Proof. For an aggregate $y \in Y$, we can check whether y is feasible by (a) computing the number of agents for each action and (b) checking if the total number of agents of all actions is n. If y is feasible and because the game is symmetric, we can create an arbitrary action profile of the agents that is consistent with y. Given the action profile and each $y' \in Y$, we can run Algorithm 2 to determine whether the action profile is an SNE. Since generating action profiles takes $O(|Y|)$ time and iterating over $y' \in Y$ takes $O(|Y|)$ time, Algorithm 2 will be run $O(|Y|^2)$ times. Therefore, we get the running time using Theorem 4.4. □

4.4.2 Non-Increasing Additive Aggregate Games

We first provide some useful results on non-increasing additive aggregate games that satisfy some natural properties.

Appendix Lemma 4.1. *Suppose that a non-increasing additive aggregate game has the properties that $d = |A|$, for any $a_i \in A = \{1, ..., d\}$ and $i \in N$, support$(\phi_i(a_i)) = \{a_i\}$, and $\pi_i(a_i, y) = \pi_i(a_i, y')$ for any $y, y' \in Y$ in which $y_{a_i} = y'_{a_i}$. The game has a PNE $\boldsymbol{a}$ with $\phi(\boldsymbol{a}) = y$ if and only if the game has an SNE $\boldsymbol{a}^*$ with $\phi(\boldsymbol{a}^*) = y$ for $y \in Y$.*

Proof. Suppose the game has an SNE $\boldsymbol{a}^*$ with $\phi(\boldsymbol{a}^*) = y$ for $y \in Y$. It follows from the definition of SNE, the action profile $\boldsymbol{a} = \boldsymbol{a}^*$ is also a PNE with $\phi(\boldsymbol{a}) = y$.

Suppose the game has a PNE $\boldsymbol{a}$ with $\phi(\boldsymbol{a}) = y = (y_1, y_2, ..., y_d)$ for $y \in Y$. We argue that any profitable coalition deviation from $\boldsymbol{a}$ must result in another action profile $\boldsymbol{a}'$ that has the same aggregate y (i.e., $\phi(\boldsymbol{a}') = y$). For the sake of contradiction, suppose that $\phi(\boldsymbol{a}') = y' = (y'_1, ..., y'_d) \neq y = (y_1, ..., y_d)$. It follows that there must be a dimension $j = \{1, \cdots, d\}$ in which $y'_j > y_j$. Consider an agent i that deviates to action $a'_i = j$ from $a_i \neq j$. It follows that

$$\begin{aligned}
&\pi_i(a_i, \phi(a_i, \boldsymbol{a}_{-i})) \\
&= \pi_i(a_i, (y_j, y_{-j})) \\
&\geq \pi_i(j, (\bar{y}_j, \bar{y}_{-j})) \\
&\geq \pi_i(j, (y'_j, \bar{y}_{-j})) \\
&= \pi_i(j, y')
\end{aligned}$$

where the first inequality is by the definition of a PNE, the third equality is because $\phi(j, \boldsymbol{a}_{-i}) = \bar{y} = (\bar{y}_1, ..., \bar{y}_d)$, the second inequality is because (a) $y'_j \geq \bar{y}_j \geq y_j$ because dimension j at $\boldsymbol{a}'$ has at least the contribution of i and dimension j at $\boldsymbol{a}$ does not have the contribution of i and (b) π_i is non-increasing depending on the value of j, and the last equality is because π_i depends on j only. There is a contradiction; i cannot improve their utility by deviating to j. Therefore, any profitable coalition deviation resulting in $\boldsymbol{a}'$ must have the same aggregate y.

Hence, when we have a PNE, we have an SNE (with possibly a different action profile) with the same aggregate.

□

The above lemma provides us with the following results regarding recognizing and determining the existence of an SNE in the considered non-increasing additive aggregate games.

More specifically, for recognizing an SNE, we are able to reduce the complexity by a factor of $|Y|$.

Appendix Theorem 4.2. *Suppose that a non-increasing additive aggregate game has the properties that $d = |A|$, for any $a_i \in A = \{1, ..., d\}$ and $i \in N$, support$(\phi_i(a_i)) = \{a_i\}$, and $\pi_i(a_i, y) = \pi_i(a_i, y')$ for any $y, y' \in Y$ in which $y_{a_i} = y'_{a_i}$. There is an $O(|Y|n^2m)$ algorithm for recognizing whether a given action profile is an SNE.*

Proof. From Appendix Lemma 4.1, we can first check to see if the given action profile is a PNE in polynomial time. If the action profile is a PNE, we can run Algorithm 2 with the action profile and its aggregate as input. □

For determining the existence of an SNE, because of Appendix Lemma 4.1, we can use our algorithmic results in Corollary 3.9 (which combines the results of Theorem 3.6 and Theorem 3.8) to determine the existence of a PNE in $O(|Y|(nm^2+|Y|nm))$. If there isn't a PNE, then we can conclude that there isn't an SNE. If there is a PNE, then we can run Algorithm 3 to turn the PNE into an SNE.

Let $\boldsymbol{a}$ be the input PNE and $\phi(\boldsymbol{a}) = y'$. Algorithm 3 takes $\boldsymbol{a}$ and y' as input and output an SNE. Algorithm 3 iteratively constructs an $(n+1)$-partite *weighted* graph $G = (Y_0, Y_1, ..., Y_n, E, w)$ where $Y_0\{(0, 0, ..., 0)\}$ and $Y_i \subseteq Y$ for $i = 1, ..., n$, edges form only between Y_{i-1} and Y_i for $i = 1, ..., n$, and edge weight $w : E \to \mathbb{R}$ is defined based on the utilities of the agents.

For $i = 1, ..., n$, an edge $(y_{i-1}, y_i) \in E$ with weight $w((y_{i-1}, y_i)) = \pi_i(a'_i, y')$, $y_{i-1} \in Y_{i-1}$ and $y_i \in Y_i$, if and only if (a) $y_0 = (0, ..., 0)$ or there is an edge (y_{i-2}, y_{i-1}) for some $y_{i-2} \in Y_{i-2}$ and (b) $y_i = y_{i-1} + \phi_i(a'_i)$ for some

Algorithm 3 Computing an SNE from a PNE in non-increasing additive aggregate games satisfying properties in Theorem 4.8

Input: Non-increasing additive aggregate game $(N, \{A_i, \pi_i\}_{i \in N}, \phi)$ has the properties of Theorem 4.8 and PNE $\boldsymbol{a} \in \boldsymbol{A}$, and $\phi(\boldsymbol{a}) = y'$
Output: an SNE $\boldsymbol{a}^*$

```
Let G = (Y_0, ..., Y_n, E, w) be (n+1)-partite weighted graph where Y_0 = {(0, 0, ···, 0)} and Y_i = {} for i ∈ {1, 2, ···, n}
for i ∈ {1, ..., n} do
    A_i^D(y') = {a'_i ∈ A_i \ {a_i} | π_i(a_i, φ(a)) < π_i(a'_i, y')} ∪ {a_i}
    for y_{i-1} ∈ Y_{i-1} do
        for a'_i ∈ A_i^D(y') do
            Let y_i = y_{i-1} + φ_i(a'_i)
            Y_i = Y_i ∪ {y_i}
            E = E ∪ {(y_{i-1}, y_i)}
            w((y_{i-1}, y_i)) = π_i(a'_i, φ(a))
return a* [extract from a maximum weighted path from y_0 ∈ Y_0 to y' ∈ Y_n]
```

$a'_i \in A_i^D(y') \subseteq A_i$ where $A_i^D(y') = \{a'_i \in A_i \setminus \{a_i\} \mid \pi_i(a_i, \phi(\boldsymbol{a})) < \pi_i(a'_i, y')\} \cup \{a_i\}$ is the set of actions that agent i can take resulting in a profitable deviation from a_i to some possible action profiles that map to aggregate y' and agent i's original action a_i (indicating no deviation). Given the n-partite weighted graph G, a maximum weighed path from $y_0 \in Y_0$ to $y \in Y_n$ can be used to form an SNE.

Theorem 4.8. *Suppose that a non-increasing additive aggregate game has the properties that $d = |A|$, for any $a_i \in A = \{1, ..., d\}$ and $i \in N$, support$(\phi_i(a_i)) = \{a_i\}$, and $\pi_i(a_i, y) = \pi_i(a_i, y')$ for any $y, y' \in Y$ in which $y_{a_i} = y'_{a_i}$. There is an $O(|Y|(nm^2 + |Y|nm + |Y|n))$ constructive algorithm for SNE existence.*

Proof. Algorithm 3 requires a PNE (if it exists) which can be checked (and returned if it exists) in $O(|Y|(nm^2 + |Y|nm))$ as discussed above. If there is a PNE, say $\boldsymbol{a}$ with aggregate $\phi(\boldsymbol{a}) = y'$, then we use it as input to Algorithm 3. Algorithm 3 takes $O(|Y|nm)$ plus the time for computing a maximum weighted path from $y_0 \in Y_0$ to $y' \in Y$. Since the graph is a directed acyclic graph, a maximum weighted path can be computed in time linear in the size of the graph; i.e., $O(n|Y| + n|Y|^2)$ or $O(n|Y|^2)$.

To show that the returned $\boldsymbol{a}^*$ is indeed an SNE, we first note that any path from $y_0 \in Y_0$ to $y' \in Y_n$ is formed by some action profile $\boldsymbol{a'}$ that maps to the aggregate y'. We now argue that a maximum weighted path will provide $\boldsymbol{a}^*$ that is SNE.

For the sake of contradiction, suppose that $\boldsymbol{a}^*$ is not an SNE and there is a profitable deviation from $\boldsymbol{a}^*$ to $\boldsymbol{a}'$. It follows that we have $\pi_i(a'_i, \phi(\boldsymbol{a}')) \geq \pi_i(a^*_i, \phi(\boldsymbol{a}^*)) \geq \pi_i(a_i, \phi(\boldsymbol{a}))$ for each agent $i \in N$, and strict inequalities hold for some agents. Since action profiles $\boldsymbol{a}$, $\boldsymbol{a}'$, and $\boldsymbol{a}^*$ all map to y', we have that $\pi_i(a'_i, y') \geq \pi_i(a^*_i, y') \geq \pi_i(a_i, y')$ for each $i \in N$, and strict inequalities hold for some agent. Therefore, we have that $\sum_{i \in N} \pi_i(a'_i, y') > \sum_{i \in N} \pi_i(a^*_i, y')$. This is a contradiction to the fact that $\boldsymbol{a}^*$ is derived from a maximum weighted path from $y_0 \in Y_0$ to $y' \in Y_n$.

□

5 CONNECTION TO OTHER GAMES

COURNOT GAMES

A Cournot game $(N, \{C_i, \pi_i\}_{i \in N}, D)$ is one of the oldest games in game theory, going back to Antoine Augustin Cournot (1801 - 1877). In fact, Cournot games motivated one of the initial inquiries into aggregate games [Martimort and Stole, 2012]. In this game, each firm $i \in N$ decides to produce $a_i \in \mathbb{Z}_{>0}$ goods at a cost of $C_i(a_i)$. The total goods produced is $\phi(\boldsymbol{a}) = a_1 + \cdots + a_n$. The market price of the goods, denoted by $D(\phi(\boldsymbol{a})) \in \mathbb{R}$, is determined by the aggregated total production of all good by all firms. Each firm's profit is determined by how many goods that firm produced, how much it cost the firm to produce those goods, and the market price of the goods. Firm i's profit is $\pi_i(a_i, \phi(\boldsymbol{a})) = D(\phi(\boldsymbol{a}))a_i - C_i(a_i)$. Clearly, Cournot is a type of additive aggregate game.