Journal of Artificial Intelligence Research 82 (2025) 1711-1742 Submitted 09/2024; published 03/2025

Existence, Computation and Efficiency of Nash Stable Outcomes in Hedonic Skill Games

Laurent Gourvès LAURENT.GOURVES@DAUPHINE.FR
Université Paris-Dauphine, Université PSL, CNRS,
LAMSADE, 75016, Paris, France

Gianpiero Monaco GIANPIERO.MONACO@UNICH.IT
University of Chieti-Pescara, Italy

Abstract

This article deals with hedonic skill games, a non-transferable utility counterpart of coalitional skill games which model collaboration among entities through the abstract notions of tasks and the skills required to complete them.

In the weighted tasks setting, we show that deciding whether an instance of the game admits a Nash stable outcome is NP-complete. We then characterize the instances admitting a Nash stable outcome. This characterization relies on the fact that every agent holds (resp., every task requires) either a single skill or more than one skill. For these instances, the complexity of computing a Nash stable outcome is determined, together with the possibility that natural dynamics converge to a Nash stable outcome from any initial configuration. Our study is completed with a thorough analysis of the price of anarchy of instances always admitting a Nash stable outcome.

1. Introduction

Game theory plays an important role in artificial intelligence because it makes it possible to analyze the incentives of the self-interested entities of some multi-agent systems. Indeed, in a cooperative game, agents organize themselves into coalitions receiving a joint payment determined by a characteristic function. This function offers great freedom to model a wide variety of multi-agent systems. Moreover, cooperative game theory provides various solution concepts on how to divide the payment of a coalition among its members, and also measures of the power of the participants. However, it is crucial for the operability of the system that computing a coalition's payment can be done efficiently. Unfortunately, some cooperative games are intractable because their characteristic function is not concisely representable, or its manipulation necessitates prohibitive calculations. Therefore, we must restrict ourselves to tractable games, at the possible cost of a loss of expressiveness concerning the systems that they model.

For this purpose, a family of transferable utility games called *coalitional skill games* (CSGs, for short) was introduced and studied. CSGs model collaboration among entities and are based on the abstract notions of tasks and the skills required to complete them. Introduced by Bachrach and Rosenschein (2008) (see also Bachrach, Parkes, & Rosenschein, 2013), CSGs are highly expressive cooperative games. Weighted Task Skill Games (WTSGs) are special CSGs which allow a succinct representation, yet preserving a good level of expressiveness (Bachrach & Rosenschein, 2008; Bachrach et al., 2013).

In WTSGs, we are given a set of agents, each one possessing a set of skills, and a set of positive weighted tasks, each one requiring a set of skills in order to be performed. An outcome for these games is a *coalition structure*, which is a partition of the agents into coalitions. A coalition can perform a task only if its members cover the set of required skills for the task. The worth of a coalition is defined as the sum of the weights of the tasks that the coalition can perform.

WTSGs have applications in many real life situations including oil extraction, voting, knowledge sharing, robots and human rescuers, and multi-sensor networks (Bachrach, Meir, Jung, & Kohli, 2010; Bachrach et al., 2013).

So far, WTSGs are viewed from the angle of cooperative game theory with transfers of utility. Typical fundamental questions in this field are the following: How to distribute the worth of collaboration that all participants find acceptable? Who are the powerful members of a coalition? Which coalition structure induces the maximum total value?[1] Classical solution concepts and power indices of cooperative game theory have been studied in WTSGs such as the core, the Shapley value, and the Banzhaf power index (Bachrach et al., 2013).

This work considers WTSGs from the perspective of *hedonic games* where utility is *not* transferable. In the present article, the utility derived by the agents of every coalition corresponds to the fair division of the total weight of the tasks performed within that coalition. Our motivation is to capture and study situations where every agent can freely and unilaterally decide which coalition she wants to be part of, given that every agent's utility depends on the members of her coalition, and is independent of the other coalitions. Therefore, central to our work is the solution concept of Nash stability. This approach leads to new questions and challenges regarding WTSGs: Does the system admit a Nash stable outcome? How difficult is the computation of such a state? Do the agents naturally converge to a Nash stable state? How bad can the social welfare of Nash stable states be?

The study of hedonic WTSGs (hedonic skill games, for short) is relevant in voting situations where the agents are candidates, the skill set consists of opinions on societal issues, and the tasks represent some segments of the electorate that are sensitive to different opinions. The candidates can freely join or depart from political coalitions, with the aim of maximizing their influence on the electorate. Hedonic skill games also model the situation where some volunteers (agents) decide to join some charity organizations (coalitions) in order to contribute to some humanitarian activities (tasks). Those activities require some skills (e.g., medicine, logistics, translation, education, etc.) held by some agents. The volunteers are free to decide which organization to join, with the objective of being as helpful as possible.

To the best of our knowledge, Nash stability and its associated unilateral deviations were not considered before for CSGs. Moreover, we emphasize that Bachrach et al. (2010) mentioned in their conclusion that further game theoretic analysis is appropriate for the setting where agents are selfish and only care about their own utility.

---

[1] CSGs are not super-additive since the union of two disjoint coalitions has a value possibly smaller than sum of the values of the separate coalitions. The grand coalition, containing all the participants, does not necessarily maximizes the value.

2. Related Work

The game studied in this article falls into the family of *coalitional formation games* where the outcome is a coalition structure, i.e., a partition of the set of agents (Hajdjuková, 2006). Each agent has a preference relation over the coalition structures. In most cases, agents only care about their own coalition in the coalition structure, without being affected by how the participants of other coalitions are partitioned. Such coalition formation games are said to be *hedonic*, a property shared by the game studied in this article. Introduced by Drèze and Greenberg (1980), and later popularized by Banerjee, Konishi, and Sönmez (2001), Bogomolnaia and Jackson (2002), Cechlárová and Romero-Medina (2001), hedonic games are among the most important game-theoretic approaches to the study of coalition formation problems. A comprehensive survey and a book chapter on hedonic games have been written by Woeginger (2013), and Aziz and Savani (2016), respectively.

Hedonic games generalize well studied problems such as *stable matchings* where the agents form pairs on the basis of their preferences (Gale & Shapley, 1962; Roth & Sotomayor, 1990). An extension of hedonic games are *generalised group activity selection games* where we are given a set of activities and a set of agents. Each agent has to be assigned to one activity and agent's preferences bear both on the activity she will be assigned to, and on the set of agents who will participate in the same activity. Specific scenarios of generalised group activity selection games have also been studied (Bilò, Fanelli, Flammini, Monaco, & Moscardelli, 2019; Darmann, Elkind, Kurz, Lang, Schauer, & Woeginger, 2018; Igarashi, Peters, & Elkind, 2017).

Our work is mainly connected to articles by Bachrach and Rosenschein (2008), Bachrach et al. (2010, 2013). Coalitional skill games (CSGs) were introduced by Bachrach and Rosenschein (2008) (see also Bachrach et al., 2013) and mainly studied from the point of view of transferable utility cooperative games (Elkind & Rothe, 2016). The work by Bachrach and Rosenschein (2008), Bachrach et al. (2013) analyzes the expressiveness of CSGs and presents several restrictions that permit a concise yet good representation for these games. The authors also consider the computational complexity of several natural problems in WTSGs and other interesting subclasses such as Task Count Skill Games (TCSGs) where each task has weight 1, and Single Task Skill Games (STSGs) where there is only one task. Specifically, they study the complexity of the following issues: testing if an agent is a dummy or veto agent,[2] computing the core and core-related solution concepts, and computing power indices such as the Shapley value and Banzhaf's power index.

The work by Bachrach et al. (2010) considers the computational complexity of computing optimal coalition structures in CSGs (see also Aziz & de Keijzer, 2011 for this approach in numerous coalitional games including CSGs) and show that the problem is hard even for very restricted classes of games, i.e., STSGs and WTSGs where tasks require at most two skills and skills are owned by at most two agents. On the positive side, they present a fixed parameter tractable algorithm for instances whose underlying structure has a bounded tree-width. Another positive result is the existence of a polynomial-time algorithm that

---

[2] An agent is dummy if his addition to a coalition never increases the value of that coalition. An agent is veto if every winning coalition (whose value is above a given threshold, or is able to perform the only task of the game) contains that specific agent.

computes a socially optimal coalition structure for WTSGs with a fixed number of skills (Aziz & de Keijzer, 2011).

Note that in the definition of WTSGs, tasks have a positive weight (Bachrach & Rosenschein, 2008; Bachrach et al., 2013), making this class a proper subclass of CSGs. Bachrach, Parkes, and Rosenschein mention that the case where negative weights are allowed is possible but controversial as agents may freely avoid to perform tasks that reduce their payoff (Bachrach et al., 2013). Nevertheless, if negative weights are allowed (and enforced), any cooperative game can be cast as a WTSG (Bachrach et al., 2010, Theorem 1), demonstrating the wide expressiveness of WTSGs, but this result is obtained with a number of tasks that is exponential in the number of agents. Since our motivation is to consider operable systems, we will focus on succinctly representable WTSGs where the tasks have positive weights.

A related model is the *Coalitional Resource Games* (CRGs) where agents wish to achieve various goals and are endowed with certain amounts of resources required to achieve these goals. CSGs and CRGs are related since performing a task in CSGs requires a coalition to have a certain set of skills, and achieving a goal in CRGs requires certain resources. However, there are important differences between CSGs and CRGs (see Section 5.4 of Bachrach et al., 2013 for a detailed comparison). Wooldridge and Dunne (2006) investigate the computational complexity of a number of natural decision problems for CRGs.

Another related topic is the *Task Oriented Team Formation problem.* We are given a task defined as a set of skills and agents possessing subsets of skills. For each pair of agents there is a cost that has to be paid if the two agents are selected in the team. The objective is to find a subset of agents that minimizes the overall cost subject to that every skill in the task is covered. To cover a skill, it is sufficient that one team member possesses that skill. Lappas, Liu, and Terzi (2009) show the hardness of this problem. Moreover, Okimoto, Schwind, Clement, Ribeiro, Inoue, and Marquis (2015) consider the robust version of the problem and define a team k-robust (for a non-negative integer k) if it can be subject to the removal of any k agents and still complies with the skill requirements for the task.

Let us also mention the hedonic game by Saad, Han, Basar, Debbah, and Hjorungnes (2011) where some wireless agents have to service a given set of entities by collecting and transmitting their data. Every such action is called a task. The agents form a coalition structure before executing the tasks. Though this model deals with tasks and coalition structures, it significantly differs from our problem because there is only one skill (data management), and each task is assigned to a single coalition.

Finally, a further related topic is the hedonic expertise games (Caskurlu, Kizilkaya, & Ozen, 2024) where we are given a global set of skills and a set of agents where each agent possesses a level of expertise in each of these skills. Agents form coalitions of limited cardinality and agents belonging to the same coalition have the same utility which is defined as the sum of the maximum expertise for each skill among the agents of the coalition.

3. Model

In this section we provide the formal definitions of the considered model.

tasks	skills	weight
t_1	$\{s_1, s_2\}$	6
t_2	$\{s_2, s_3\}$	4

agents	skills
1	$\{s_1, s_2\}$
2	$\{s_1, s_3\}$
3	$\{s_1\}$

state	coalition structure	performed tasks	utilities u_1	u_2	u_3	social welfare SW
σ	$(\{1, 2, 3\})$	t_1 and t_2 performed by $\{1, 2, 3\}$	6	3	1	10
σ'	$(\{2\}, \{1, 3\})$	t_1 performed by $\{1, 3\}$, no task performed by $\{2\}$	9/2	0	3/2	6

Table 1: Instance depicted in Example 1.

3.1 Setting

A *hedonic skill game* is composed of a set of agents $\mathcal{N} = \{1, \dots, n\}$, a set of tasks $\mathcal{T} = \{t_1, \dots, t_m\}$, and a set of skills $\mathcal{S} = \{s_1, \dots, s_k\}$. Each task t_j has a positive weight $w(t_j)$ and requires a non-empty set of skills $S(t_j) \subseteq \mathcal{S}$ in order to be executed. Every agent $\ell \in \mathcal{N}$ has a non-empty skill set $S(\ell) \subseteq \mathcal{S}$.

Let us illustrate the model with a small running example summarized in Table 1.

Example 1. *Suppose there are three skills $\{s_1, s_2, s_3\}$, two tasks $\{t_1, t_2\}$ of weights $w(t_1) = 6$ and $w(t_2) = 4$ requiring skills $S(t_1) = \{s_1, s_2\}$ and $S(t_2) = \{s_2, s_3\}$, and a set of three agents $\mathcal{N} = \{1, 2, 3\}$ having skills $S(1) = \{s_1, s_2\}$, $S(2) = \{s_1, s_3\}$, and $S(3) = \{s_1\}$.*

3.2 States and Coalition Structures

Agents decide which coalition they want to be part of. In a state σ of the hedonic skill game, each coordinate σ_ℓ indicates the coalition that agent ℓ has decided to join. *Mixed strategies*, where agents' decisions are probability distributions over multiple coalitions, are not considered in this paper. Instead, the decisions are said to be *pure*, indicating that every agent selects exactly one coalition. As is standard in game theory, $\sigma_{-\ell}$ denotes the state σ from which σ_ℓ is removed, and $(\sigma_{-\ell}, C)$ refers to the state σ where σ_ℓ is replaced by C.

Every state σ induces a *coalition structure* $(C_1, \dots, C_h)$ such that each C_i is the non-empty subset of agents who selected C_i. A coalition structure is a partition of $\mathcal{N}$, i.e., $\emptyset \neq C_i \subseteq \mathcal{N}$ for all i, $\cup_{i=1}^{h} C_i = \mathcal{N}$, and $C_i \cap C_{i'} = \emptyset$ for all (i, i') satisfying $1 \leq i < i' \leq h$. The number of coalitions of $(C_1, \dots, C_h)$ is h. In this article, we interchangeably use the notion of state of the game and its induced coalition structure, as they both determine the necessary information about the agents' joint decisions.

Every coalition C_i holds the set of skills $S(C_i) := \cup_{\ell \in C_i} S(\ell)$, which is the union of the skills of its members. A coalition C_i is able to perform a task t_j if its members have all the needed skills, i.e., when $S(t_j) \subseteq S(C_i)$.

In the model, a task t can be performed by more than one coalition if several coalitions contain agents who possess the skills that t requires. However, a coalition executes a task at most once.

Example 1 (continued). *Under the state σ where the three agents are in the same coalition, the coalition structure is $(\{1, 2, 3\})$, and the tasks t_1 and t_2 are both performed by the unique coalition. Consider another state σ' where agent 2 is alone in a coalition while agents 1 and 3 are together in another coalition. Then, the coalition structure associated with σ' is $(\{2\}, \{1, 3\})$, and only the task t_1 can be performed by $\{1, 3\}$.*

In this article, $T(C_i)$ denotes the subset of tasks that C_i can perform.

$$T(C_i) := \{t \in \mathcal{T} \mid S(t) \subseteq S(C_i)\}$$

3.3 Utilities and Social Welfare

Let us now define the utility of the agents. Due to the hedonic nature of our skill game, the utility of every agent depends on the agents that are in her coalition, and is independent on how the agents in other coalitions are grouped together. Moreover, utilities are non-transferable.

In the model, the weight $w(t)$ of every task t that a coalition C can perform is seen as a monetary reward distributed among C's members having at least one skill within $S(t)$. Concretely, for every skill $s \in S(t)$, every agent $\ell \in C$ such that $s \in S(\ell)$ receives a reward of value

$$\frac{w(t)}{|S(t)| \cdot |\{a \in C \mid s \in S(a)\}|}.$$

Hence, the weight of a task t requiring $|S(t)|$ different skills is cut into $|S(t)|$ equal parts of value $w(t)/|S(t)|$. Then, each of these parts is divided by the number of members who possess the corresponding skill, namely $|\{a \in C \mid s \in S(a)\}|$ where $s \in S(t) \cap S(\ell)$.

The total sum of the rewards that an agent receives constitutes her utility.[3]

Definition 1. *The utility of agent ℓ belonging to coalition C is equal to*

$$\sum_{t \in T(C)} \sum_{s \in S(t) \cap S(\ell)} \frac{w(t)}{|S(t)| \cdot |\{a \in C \mid s \in S(a)\}|}.$$

Example 1 (continued). *Under the state σ, corresponding to the coalition structure $(\{1, 2, 3\})$, agent 1 has utility 6, agent 2 has utility 3, and agent 3 has utility 1. Under the state σ', corresponding to the coalition structure $(\{2\}, \{1, 3\})$, agent 1 has utility $9/2$, agent 2 has utility 0, and agent 3 has utility $3/2$.*

The way utilities are defined is such that: (*i*) only agents who contribute to the execution of a task receive a positive reward from this task (no free riding), (*ii*) all agents who contribute to the execution of a task through the same skill receive the same reward (fairness), and (*iii*) the sum of the agents' utilities within a coalition C is always equal to the total weight of the tasks performed by C (efficiency). More generally, the sum over $\mathcal{N}$ of

---

[3] See Anshelevich, Dasgupta, Kleinberg, Tardos, Wexler, and Roughgarden (2008) for a similar equal-division mechanism in a different context.

the agents' utilities is equal to the total weight of the tasks performed in the whole coalition structure.[4] One can also observe that Definition 1 is additive with respect to the rewards (implying computability and monotonicity because weights are positive), and anonymous (labels, by which members are identified, are irrelevant).

As a notation, u_ℓ denotes agent ℓ's utility function. The social welfare associated with a state σ, denoted by $SW(\sigma)$, is defined as the sum of the agents' utilities (a.k.a. utilitarian social welfare). From the above discussion, $SW(\sigma)$ is also equal to the total weight of all the executed tasks. Thus, if σ induces the coalition structure $(C_1, \ldots, C_h)$, then it holds that

$$SW(\sigma) = \sum_{\ell \in \mathcal{N}} u_\ell(\sigma) = \sum_{i=1}^{h} \sum_{t \in T(C_i)} w(t). \tag{1}$$

A *social optimum* refers to a state σ^* for which $SW(\sigma^*)$ is maximum.

In Example 1, the social welfare is 10 under the coalition structure $(\{1,2,3\})$ associated with σ. It is 6 under the coalition structure $(\{2\},\{1,3\})$ associated with σ'.

3.4 Equilibrium

Our objective is to study hedonic skill games through their equilibria (if they exist). A (pure) *Nash stable* outcome (a.k.a. Nash equilibrium in pure strategy) is a situation where no agent can deviate from her current coalition and strictly increase her utility (i.e., no agent possesses a better response).[5] Thereafter, we exclusively deal with pure states and omit the word *pure*.

Definition 2. *A state σ is a Nash stable outcome if $u_\ell(\sigma) \geq u_\ell(\sigma_{-\ell}, C)$ holds for all agent ℓ and coalition C.*

In Example 1, state σ is Nash stable but σ' is not Nash stable because agent 2 (resp., agent 1) would increase her utility by joining the coalition $\{1,3\}$ (resp., coalition $\{2\}$). Let us introduce another example.

Example 2. *Suppose there are two skills $\{s_1, s_2\}$, a single task t_1 of weight $w(t_1) = 2$ requiring skills $S(t_1) = \{s_1, s_2\}$, and two agents having skills $S(1) = \{s_1, s_2\}$ and $S(2) = \{s_2\}$ (cf. Table 2). If the two agents are in the same coalition, then the task t_1 can be performed: agent 1 has utility $3/2$ whereas agent 2 has utility $1/2$. If agent 2 is alone, then the task cannot be performed in her coalition, and agent 2's utility is 0. If agent 1 is alone, then the task can be performed in her coalition, and agent 1's utility is 2.*

Example 2 disproves the guaranteed existence of a Nash stable outcome, even if the number of possible coalitions is limited to two. Indeed, agent 1 prefers being alone over being with agent 2, and agent 2 prefers being with agent 1 over being alone.

Depending on the situation that the game models, the number of coalitions of any coalition structure can be either unconstrained, or upper bounded by some given parameter

---

[4] The weight of a task is counted x times if it is performed by x distinct coalitions.

[5] As opposed to *individual stability*, deviations that possibly hurt the members of the new coalition are allowed.

tasks	skills	weight
t_1	$\{s_1, s_2\}$	2

agents	skills
1	$\{s_1, s_2\}$
2	$\{s_2\}$

Table 2: Instance depicted in Example 2.

that we denote by q. Since the number of coalitions of a coalition structure cannot exceed the number of agents, $q \leq n$ can always be assumed without loss of generality.

Observe that in Example 2, agent 1 and the task both have more than one skill. Then, it is interesting to understand whether this fact has an impact on the Nash stable outcomes in hedonic skill games.

Definition 3. *An agent (resp., a task) is said to be* singleton *if she has (resp., it requires) a single skill.*

We will differentiate the instances solely composed of singleton agents (resp., singleton tasks), and the others called *general agents instances* (resp., *general tasks instances*) where the agents (resp., tasks) are not necessarily singleton.

As we will see in Section 4, this distinction has a significant influence on the various aspects of the hedonic skill game: existence of a Nash stable outcome, convergence of the better response dynamics, complexity of computing socially optimal coalition structures, and the price of anarchy. In addition, the upper bound q on the number of possible coalitions of the coalition structure can also play a role. Since studying all the aforementioned aspects for every possible value of q represents a huge amount of work, the focus is put on the two extreme cases, namely $q = n$ (the number of coalitions of the coalition structure is unconstrained) and $q = 2$.

4. Contribution

We first show in Section 5 that for general instances such that $q \geq 3$, the problem of deciding whether an instance of the hedonic skill game admits a Nash stable outcome is NP-complete (Theorem 1). We then provide in Sections 6 and 7 a complete picture about the existence of Nash stable outcomes in the hedonic skill game. Specifically, we demonstrate that a Nash stable outcome always exists if all the agents are singleton (Theorem 3 for $q = n$ and Proposition 3 for $q = 2$) or if all the tasks are singleton (Corollary 1). These results are obtained with different techniques. We notice that Example 2 covers the remaining cases (namely, non-existence of a Nash stable outcome for general agents/tasks instances) for all $q \geq 2$. We also show that computing a Nash stable outcome is a **PLS**-complete problem (Theorem 5) with singleton tasks, indicating that it is a difficult task, but the problem is in **P** if the agents are singleton (Theorem 3 for $q = n$ and Proposition 3 for $q = 2$).

Moreover, as in previous works on CSGs, we adopt the utilitarian social welfare, which is defined as the sum of the agents' utilities, as measure of the well-being of the system. It is known that computing a socially optimal coalition structure is **NP**-hard, even in restricted cases (Bachrach et al., 2010), i.e., for Single Task Skill Games (STSGs) where there is only one task and for WTSGs where tasks require at most two skills and skills are owned by at

<table>
<tr><th></th><th>singleton tasks</th><th>general tasks</th></tr>
<tr><td>singleton agents</td><td colspan="2">- A Nash stable outcome always exists and it can be computed in polynomial time when $q = n$ (Thm. 3) and when $q = 2$ (Prop. 3).
- Computing a social optimum is in **P** when $q = n$ (Thm. 2) and $q = 1$ (Prop. 3).</td></tr>
<tr><td>general agents</td><td>- A Nash stable outcome always exists (Cor. 1) but its computation is **PLS**-complete when $q = 2$ (Thm. 5).
- Computing a social optimum is **NP**-hard when $q = 2$ but it is in **P** when $q = n$ (Thm. 6).</td><td>- Existence of a Nash stable outcome is not guaranteed (Ex. 2).
- Deciding whether a Nash stable outcome exists is **NP**-complete (Thm. 1).
- Computing a social optimum is **NP**-hard (Bachrach et al., 2010).</td></tr>
</table>

Table 3: Existence and computation of Nash stable and socially optimal outcomes.

<table>
<tr><th></th><th>singleton tasks</th><th>general tasks</th></tr>
<tr><td>singleton agents agents</td><td rowspan="2">The BRD always converge (Thm. 4).</td><td>The BRD can cycle (Prop. 1) unless every skill is possessed by a single agent (Prop 2).</td></tr>
<tr><td>general agents</td><td>The BRD can cycle (Prop. 1).</td></tr>
</table>

Table 4: Convergence to a Nash stable outcome of the better response dynamics (BRD).

most two agents. In this work we show that a social optimum can be built in polynomial time for singleton agents instances when $q = n$ (Theorem 2) and when $q = 2$ (Proposition 3).[6] Concerning singleton tasks instances, we prove that maximizing the social welfare is **NP**-hard when $q = 2$ and polynomial time solvable when $q = n$ (Theorem 6).

Our results for existence and computation of Nash stable outcomes and social optima are summarized in Table 3.

We further study agent dynamics (see Table 4 for a summary of our results). Specifically, starting from any coalition structure, the better response dynamics (BRD, for short) consist of repeatedly applying better responses by single agents[7] as long as it is possible. If the BRD eventually stop, then we say that they converge and the final state must be a Nash stable outcome. Otherwise, we say that the dynamics cycle. See Algorithm 1 for a pseudo-code of BRD. We prove that the BRD always converge if the tasks are singleton (Theorem 4) but they can cycle with general tasks, even if agents are singleton (Proposition 1). Nevertheless, the BRD converge in singleton agents instances for which every skill is possessed by a single agent (Proposition 2).

Finally, before concluding with a list of possible future works, we study in Section 8 the quality of Nash stable outcomes by resorting to the price of anarchy (PoA, for short), which is the largest value, over all instances, of the optimal social welfare divided by the

---

[6] The problem is actually in **P** for every q, cf. Theorem 10 in the appendix.

[7] Choose one deviation arbitrarily if there is more than one.

Algorithm 1 Better response dynamics (BRD)

```
Input: an initial state $\sigma_0$ whose corresponding coalition structure contains at most $q$ coalitions
$\sigma \leftarrow \sigma_0$
while $\exists$ an agent $\ell \in \mathcal{N}$ and a coalition $C$ such that $u_\ell(\sigma_{-\ell}, C) > u_\ell(\sigma)$ {under the constraint that the number of coalitions never exceeds q, and $C = \emptyset$ corresponds to the case where the deviation of $\ell$ creates a new coalition $\{\ell\}$} do
    $\sigma_\ell \leftarrow C$
end while
return $\sigma$
```

PoA	singleton agents	singleton tasks
$q = 2$	PoA=3 if $\tau = 2$, PoA=5 if $\tau = 3$, and unbounded PoA if $\tau > 3$ (Thm. 8)	PoA=4/3 (Thm. 9)
$q = n$	$\frac{n}{2} \leq$PoA$\leq 2n$ if $\tau = 2$, and unbounded PoA if $\tau > 2$ (Thm. 7)	PoA=1

Table 5: Price of Anarchy (PoA for short).

social welfare (SW) of a Nash stable outcome (Koutsoupias & Papadimitriou, 1999). Our results, summarized in Table 5, show that the PoA is sometimes sensitive to a parameter τ defined as the maximum number of skills required by a task.

5. Hardness of Deciding if a Nash Stable Outcome Exists

We know from Example 2 that some instances fail to have a Nash stable outcome. Therefore, it is natural to settle the complexity of deciding whether a given general instance possesses a Nash stable outcome or not. In the following theorem we show that deciding whether an instance of the hedonic skill game admits a Nash stable outcome is a hard problem when $q \geq 3$.

Theorem 1. *Deciding whether an instance of the hedonic skill game admits a Nash stable outcome is an NP-complete problem when $q \geq 3$.*

Proof. The starting point is the NP-complete PARTITION problem. Given n positive integers $a_1, \ldots, a_n$ such that $\sum_{i=1}^{n} a_i = 2B$, decide if the numbers can be partitioned in two subsets which both sum up to B.

Given an instance $\mathcal{I}_P$ of PARTITION, an instance $\mathcal{I}_H$ of the hedonic skill game is built as follows.

- create an agent i with skills $\{s_i, s'_i, s''_i\}$ for all $i \in [n]$,
- create two agents with skills $\{s_0, s_1, s_2, \ldots, s_n\}$ and call these agents $n+1$ and $n+2$, respectively,

- create $q-2$ agents with skills $\{s_0, s_1, s_2, \ldots, s_n, s_{n+1}\} \cup \{s''_1, s''_2, \ldots, s''_n\}$ and call these agents $n+3, \ldots, n+q$,
- create a task of weight $4a_i$ requiring skills $\{s_i, s'_i\}$ for all $i \in [n]$,
- create a task of weight $\varepsilon = 0.4$ requiring skill s''_i for all $i \in [n]$,
- create a task of weight $4B - \varepsilon$ requiring skills $\{s_0, s_{n+1}\}$.

We claim that $\mathcal{I}_H$ admits a Nash stable outcome if and only if $\mathcal{I}_P$ is a yes instance.

The first step is to show that the existence of a partition implies the existence of a Nash stable outcome. Let X be a subset of indices such that $\sum_{i \in X} a_i = \sum_{i \in [n] \setminus X} a_i = B$. Consider the following state: agent i such that $i \in X$ is in the first coalition, agent i such that $i \in [n] \setminus X$ is in the second coalition, agent $n+1$ is in the first coalition, agent $n+2$ is in the second coalition, and every remaining coalition from 3 to q admits exactly one agent amongst $\{n+3, \ldots, n+q\}$. The given state is a Nash stable outcome for the following reasons. Every agent amongst $\{n+3, \ldots, n+q\}$ has utility equal to $4B + (n-1)\varepsilon$, and if such an agent moves, then her utility decreases because she has to share the reward associated with the task of weight $4B - \varepsilon$ and she looses at least $B - \varepsilon/4$. Agent $n+1$ has utility equal to B, and if she moves, then her utility decreases. Indeed, moving to coalition 2 leads to a utility of $2B/3$, and moving to some coalition $c \in \{3, \ldots, q\}$ leads to a utility of $(4B - \varepsilon)/4 < B$. As for agent $n+1$, agent $n+2$ has no incentive to deviate. Every agent $i \in [n]$ in coalition $c(i) \in \{1, 2\}$ has no incentive to move to coalition $3 - c(i)$, because her utility would be the same (namely, $3a_i + \varepsilon$). Similarly, agent $i \in [n]$ has no incentive to move to some coalition $c \in \{3, \ldots, q\}$ because her utility would decrease and become $3a_i + \varepsilon/2$.

Now let us show that the existence of a Nash stable outcome in $\mathcal{I}_H$ implies the existence of a partition in $\mathcal{I}_P$. We first notice that no agent $i \in \{n+3, \ldots, n+q\}$ wants to be in the same coalition as agent $j \in \{n+1, n+2, n+3, \ldots, n+q\} \setminus \{i\}$ in order to prevent sharing the reward of the task of weight $4B - \varepsilon$ with someone else. Agent $n+1$ (or $n+2$) must be in the same coalition as at least one agent $i \in [n]$, otherwise her utility would be null[8], and she can have a positive utility by choosing a coalition where at least one agent $i \in [n]$ is, because she participates in the completion of the task of weight $4a_i$. Let us explain why agents $n+1$ and $n+2$ cannot be in the same coalition in a Nash stable outcome. If, by contradiction, they are together, then we have seen that at least one agent $i \in [n]$ must be with them, but agent i can deviate to a coalition where no agent j such that $j > n$ is. With this deviation, agent i does not share the reward of $4a_i$ anymore, so it must be a profitable deviation for agent i, contradicting the Nash stability of the state. So far, we have proved that all the agents $n+1, n+2, \ldots, n+q$ occupy distinct coalitions (one such agent per coalition since there are at most q coalitions). Note that for all agent $i \in [n]$, it is better to be with $n+1$ or $n+2$, compared to be with $j \in \{n+3, \ldots, n+q\}$, because in the first case only the reward of $4a_i$ is shared, while in the second case, both $4a_i$ and ε are shared. Therefore, we can conclude that any Nash stable outcome is as follows: a first coalition contains agent $n+1$ plus some agents whose indices are in a set $X \subset [n]$, a second coalition contains agent

---

[8] We have just seen that no agent in $\{n+3, \ldots, n+q\}$ wants to be in the same coalition as $n+1$ or $n+2$, because the state is Nash stable by assumption.

$n+2$ plus the agents whose indices are in the set $[n] \setminus X$, and every other coalition contains exactly one agent in $\{n+3, \ldots, n+q\}$. The utility of agent $n+1$ is $\sum_{i \in X} a_i$, because she has skill s_i and she collaborates with agent i who has skills $\{s_i, s'_i\}$ for completing a task of weight $4a_i$ requiring skills $\{s_i, s'_i\}$. If $\sum_{i \in X} a_i < B$, then $\sum_{i \in X} a_i \leq B - 1$ because each a_i is an integer. Agent $n+1$ can deviate to the third coalition so that her utility becomes $(4B - \varepsilon)/4 = B - 0.1$, which is strictly larger than $B - 1$. Since the current state is Nash stable, we get a contradiction, meaning that $\sum_{i \in X} a_i \geq B$ must hold. Using similar arguments, we deduce that the utility of agent $n+2$ is $\sum_{i \in [n] \setminus X} a_i$, and $\sum_{i \in [n] \setminus X} a_i \geq B$. Since $\sum_{i \in X} a_i + \sum_{i \in [n] \setminus X} a_i = \sum_{i \in [n]} a_i = 2B$, we get that $\sum_{i \in X} a_i = \sum_{i \in [n] \setminus X} a_i = B$. In other words, any Nash stable outcome provides a partition of the numbers. $\square$

Despite that the problem is NP-complete in general, in the remainder of the paper we propose a characterization of the instances admitting a Nash stable outcome. We conclude this section by noticing that it remains open to study the complexity of deciding whether a given general instance possesses a Nash stable outcome or not for the case of $q = 2$.

6. Nash Stability with Singleton Agents

In this section we suppose that every agent has a single skill. The tasks can require more than one skill.

6.1 Unconstrained Number of Coalitions

We first consider the case where $q = n$. We start by showing in Theorem 2 that it is possible to compute in polynomial time a coalition structure which maximizes the social welfare (this problem is known to be **NP**-hard in general (Bachrach et al., 2010)). We build upon this first result to show in Theorem 3 how to compute a Nash stable outcome in polynomial time.

Theorem 2. *Finding a coalition structure which maximizes the social welfare SW can be done in polynomial time for singleton agents instances when $q = n$.*

Proof. The coalition structure is built from scratch with the help of a set of agents X. At the beginning X is equal to $\mathcal{N}$. Let $i = 1$. Afterwards, do the following steps repeatedly until $X = \emptyset$: create a coalition $C_i \subseteq X$ such that $|C_i| = |S(X)|$ and $S(C_i) = S(X)$, remove C_i from X, and $i \leftarrow i + 1$.

At each step, the algorithm creates a new coalition which has one representative (chosen arbitrarily) for every skill present in X. Let $(C_1, \ldots, C_t)$ denote the coalition structure built by the algorithm. We clearly have $S(C_1) \supseteq S(C_2) \supseteq \ldots \supseteq S(C_t)$.

For every skill $s \in \mathcal{S}$, let $n(s)$ be the total number of agents having skill s in the instance. For every subset of skills $Y \subseteq \mathcal{S}$, let $\mu(Y)$ denote $\min_{s \in Y} n(s)$. Each task $t \in T$ can be performed at most $\mu(S(t))$ times because at most $\mu(S(t))$ coalitions can contain all the needed skills.

By construction of $(C_1, \ldots, C_t)$, every skill s appears in coalitions $C_1, \ldots, C_{n(s)}$. Thus, for every non-empty $Y \subseteq \mathcal{S}$, we have $Y \subseteq C_i$ for $i = 1..\mu(Y)$. It follows that $(C_1, \ldots, C_t)$ maximizes the social welfare because every task t is executed its maximum number of times $\mu(S(t))$. $\square$

Note that Theorem 2 can be generalized to any value of q (see Theorem 10 in the appendix).

Theorem 3. *A Nash stable outcome always exists and it can be computed in polynomial time for singleton agents instances when* $q = n$.

Proof. We consider dedicated dynamics D which start from a particular coalition structure built in polynomial time, and at each step, D select a specific deviation. We will show that after a polynomial number of deviations, a Nash stable outcome is reached.

For every possible state, the corresponding coalition structure is represented with a $k \times n$ table Y where each row corresponds to a skill, each column is associated with a coalition, and the entry $y_{i,j}$ of Y is equal to the number of agents having skill s_i and belonging to coalition C_j. Given Y, we can define another $k \times n$ table L whose entry $L_{i,j}$ is the sum of rewards shared by the (possibly prospective) agents of skill s_i in coalition C_j, given that the skills $s_{i'} \neq s_i$ already present in C_j are in accordance with Y. More precisely, we consider that every skill $s_{i'}$ such that $i' \neq i$ is present in C_j if, and only if, $y_{i',j} > 0$. Then, $L_{i,j}$ is the sum of rewards that agents of skill s_i would collectively receive if they were present in C_j. Therefore, no matter what $y_{i,j}$ currently is, if exactly $x > 0$ agents having skill s_i are in C_j, then each of them would have utility $L_{i,j}/x$. In the following, if an agent belongs to coalition C_j and performs an improving move by joining coalition $C_{j'}$, then we call C_j and $C_{j'}$ the *departing* and *arrival* coalitions, respectively. A single agent move (simply called "move" afterwards) from C_j to $C_{j'}$ is *left* if $j > j'$ (resp., *right* if $j < j'$). A move is *best* (resp., *better*) if it is a best (resp., better) response. Given a and b satisfying $1 \leq a < b \leq n$, $C_a \overset{\text{best } i}{\leftarrow} C_b$ denotes a best left move by an agent having skill s_i from departing coalition C_b to arrival coalition C_a.

The initial state of the dynamics D is the coalition structure built in the proof of Theorem 2. Thus, each row of Y initially consists of consecutive 1s, followed by consecutive 0s. While the current state is not a Nash stable outcome, the next move of the dynamics D must be a best left move invariably selected as follows.

1. For every skill index i such that at least one agent having skill s_i wants to make a best left move, do: Each time there are two possible moves $C_a \overset{\text{best } i}{\leftarrow} C_b$ and $C_{a'} \overset{\text{best } i}{\leftarrow} C_b$ such that $a < a' < b$, discard $C_{a'} \overset{\text{best } i}{\leftarrow} C_b$. Each time there are two possible moves $C_a \overset{\text{best } i}{\leftarrow} C_{b'}$ and $C_a \overset{\text{best } i}{\leftarrow} C_b$ such that $a < b' < b$ and $y_{i,b'} = y_{i,b}$, discard $C_a \overset{\text{best } i}{\leftarrow} C_{b'}$.

2. Within the best left moves that survived step 1, execute the one whose departing coalition has smallest index. Break ties arbitrarily.

The dynamics D that we are going to describe maintain the following invariants:

$$y_{i,j} \geq y_{i,j+1} \text{ for all pair } (i,j) \tag{2}$$

$$L_{i,j} \geq L_{i,j+1} \text{ for all pair } (i,j) \tag{3}$$

$$\text{No better right move exists} \tag{4}$$

Invariants (2) and (3) are trivially satisfied by the initial state. The remainder of the proof relies on three intermediate Lemmas whose proofs appear in the appendix.

Lemma 1. *The initial state of D satisfies Invariant (4).*

tasks	skills	weight
t_{12}	$\{s_1, s_2\}$	78
t_{14}	$\{s_1, s_4\}$	96
t_{23}	$\{s_2, s_3\}$	114
t_{15}	$\{s_1, s_5\}$	102
t_{16}	$\{s_1, s_6\}$	54
t_{123}	$\{s_1, s_2, s_3\}$	18
t_{134}	$\{s_1, s_3, s_4\}$	180

agents	skills
1	$\{s_1\}$
2	$\{s_1\}$
3	$\{s_1\}$
4	$\{s_2\}$
5	$\{s_3\}$
6	$\{s_4\}$
7	$\{s_5\}$
8	$\{s_6\}$

Table 6: Instance used in Proposition 1.

It follows that if the initial state is not a Nash stable outcome, then the first move of D is a best left move. For the guaranteed existence of a Nash stable outcome, it remains to prove that the three invariants are maintained throughout the execution of D. Indeed, either a Nash stable outcome is reached, or the previous move (which is a best left move) never leaves the possibility for an agent to move to the right in the table Y, meaning that D eventually stops.

Lemma 2. *The dynamics D maintain Invariants (2) and (3).*

Lemma 3. *A best left move does not trigger the existence of a better right move, i.e., Invariant (4) is maintained.*

The convergence occurs after a polynomial number of steps because D solely perform left moves (there are n agents and each one can move at most $n-1$ times), and determining the next move performed by D (or that a Nash stable outcome is reached) can be done in $O(kn^2)$ by checking, for every skill index i, if it is worth moving from C_b to C_a, where $1 \leq a < b \leq n$. □

In the proof of Theorem 3, the existence of a Nash stable outcome is obtained with specific dynamics applied on a particular initial coalition structure. However, it is legitimate to ask whether a Nash stable outcome can be reached by natural dynamics such as the better response dynamics (BRD).

Proposition 1. *There exists an instance of the hedonic skill games with singleton agents such that $q = 4$ where the BRD cycle.*

Proof. Let us present an instance and a cyclic sequence of better response deviations. The instance consists of 8 agents and 7 tasks (cf. Table 6). The 8 states of the cycle are reported in Table 7 (one state per line), where each multiset represents a coalition and the skills of its members. There are 4 coalitions denoted by C_1, C_2, C_3, and C_4.

The first deviation is done from C_1 to C_2 by an agent having skill s_1. The initial and final utilities of the agent are $78/2 + 18/3 = 45$ and $96/2 = 48$, respectively. The second deviation is done from C_1 to C_2 by an agent having skill s_3. The initial and final utilities of the agent are $114/2 = 57$ and $180/3 = 60$, respectively. The third deviation is done from C_3 to C_2 by an agent having skill s_1. The initial and final utilities of the agent are

C_1	C_2	C_3	C_4
$\{\{s_1, s_2, s_3\}\}$	$\{\{s_4\}\}$	$\{\{s_1, s_5\}\}$	$\{\{s_1, s_6\}\}$
$\{\{s_2, s_3\}\}$	$\{\{s_1, s_4\}\}$	$\{\{s_1, s_5\}\}$	$\{\{s_1, s_6\}\}$
$\{\{s_2\}\}$	$\{\{s_1, s_3, s_4\}\}$	$\{\{s_1, s_5\}\}$	$\{\{s_1, s_6\}\}$
$\{\{s_2\}\}$	$\{\{s_1, s_1, s_3, s_4\}\}$	$\{\{s_5\}\}$	$\{\{s_1, s_6\}\}$
$\{\{s_2\}\}$	$\{\{s_1, s_1, s_1, s_3, s_4\}\}$	$\{\{s_5\}\}$	$\{\{s_6\}\}$
$\{\{s_1, s_2\}\}$	$\{\{s_1, s_1, s_3, s_4\}\}$	$\{\{s_5\}\}$	$\{\{s_6\}\}$
$\{\{s_1, s_2, s_3\}\}$	$\{\{s_1, s_1, s_4\}\}$	$\{\{s_5\}\}$	$\{\{s_6\}\}$
$\{\{s_1, s_2, s_3\}\}$	$\{\{s_1, s_4\}\}$	$\{\{s_5\}\}$	$\{\{s_1, s_6\}\}$

Table 7: The BRD cycle of Proposition 1.

$102/2 = 51$ and $96/4 + 180/6 = 54$, respectively. The fourth deviation is done from C_4 to C_2 by an agent having skill s_1. The initial and final utilities of the agent are $54/2 = 27$ and $96/6 + 180/9 = 36$, respectively. The fifth deviation is done from C_2 to C_1 by an agent having skill s_1. The initial and final utilities of the agent are $96/6 + 180/9 = 36$ and $78/2 = 39$, respectively. The sixth deviation is done from C_2 to C_1 by an agent having skill s_3. The initial and final utilities of the agent are $180/3 = 60$ and $18/3 + 114/2 = 63$, respectively. The seventh deviation is done from C_2 to C_4 by an agent having skill s_1. The initial and final utilities of the agent are $96/4 = 24$ and $54/2 = 27$, respectively. The last deviation goes back to the initial state. It is done from C_2 to C_3 by an agent having skill s_1. The initial and final utilities of the agent are $96/2 = 48$ and $102/2 = 51$, respectively. □

Nevertheless, the BRD always converge in singleton agents instances where every skill is possessed by a single agent when $q = n$. This is due to the existence of a potential.

Proposition 2. *The BRD always converge in singleton agents instances where every skill is possessed by a single agent and* $q = n$.

Proof. Consider the function $\Phi(\sigma) := \sum_{t \in \mathcal{T}} \frac{M(t,\sigma) \cdot w(t)}{|S(t)|}$ where $M(t, \sigma)$ denotes the number of occurrences of tasks t executed under σ. We are going to prove that Φ is an *exact potential function*, i.e., each time an agent makes a better response, her utility and Φ increase by the same amount (Monderer & Shapley, 1996). Every local optimum of an exact potential, corresponding to a state where no single agent can profitably deviate, is a Nash stable outcome. Therefore, the BRD always converge if the game admits an exact potential function Φ.

Since no two agents share the same skill, and each agent has exactly one skill, we can make the following observations: (*i*) There is a one-to-one correspondence between skills and agents. Thus, we can interchangeably mention the set of skills that a task requires and the corresponding set of agents. (*ii*) The utility that an agent i derives from an executed task t requiring her is exactly $\frac{w(t)}{|S(t)|}$. (*iii*) Every task is executed at most once in any given state. Thus, $M(t, \sigma) \in \{0, 1\}$ for all task t and all state σ.

Take an agent i who has a better response to the current state σ, and denote by σ' the resulting state. We assume w.l.o.g. that agent i moves from C_1 to C_2. The deviation *destroys* all the tasks executed within C_1 which require i. At the same time, some tasks requiring i are *created* within C_2. The deviation does not create any task within C_1, and no task is destroyed within C_2. Let $\mathcal{T}_d$ and $\mathcal{T}_c$ be the set of destroyed and created tasks by the deviation of i, respectively. We have that $u_i(\sigma') - u_i(\sigma) = \sum_{t \in \mathcal{T}_c} \frac{w(t)}{|S(t)|} - \sum_{t \in \mathcal{T}_d} \frac{w(t)}{|S(t)|}$. Indeed, agent i derives her utility under σ solely from the tasks of $\mathcal{T}_d$. Concerning σ', i derives her utility solely from the tasks of $\mathcal{T}_c$. Note that the only task requiring i that can possibly remain during the deviation is the one requiring only i.

Let $\mathcal{T}(\sigma)$ and $\mathcal{T}(\sigma')$ be the set of tasks executed under σ and σ', respectively. We have that $\Phi(\sigma') - \Phi(\sigma) = \sum_{t \in \mathcal{T}(\sigma')} \frac{w(t)}{|S(t)|} - \sum_{t \in \mathcal{T}(\sigma)} \frac{w(t)}{|S(t)|}$. Since the tasks not requiring i are not impacted by the deviation, $\Phi(\sigma') - \Phi(\sigma)$ is equal to $\sum_{t \in \mathcal{T}_c} \frac{w(t)}{|S(t)|} - \sum_{t \in \mathcal{T}_d} \frac{w(t)}{|S(t)|} = u_i(\sigma') - u_i(\sigma)$. □

6.2 At Most Two Coalitions

Theorem 3 holds for an unconstrained number of coalitions. The technique consists of applying specific dynamics on a particular initial coalition structure. However, when q is restricted, the technique is not guaranteed to work if the aforementioned initial state requires more than q coalitions. Therefore, we show in the following how a Nash stable state can be efficiently computed when $q = 2$.

Proposition 3. *When $q = 2$, hedonic skill games with singleton agents always have a Nash stable outcome which can be computed in polynomial time. Moreover, computing a state that maximizes the social welfare in this case can be done in polynomial time.*

Proof. The proof strategy consists of starting from a socially optimal state and each time an agent in coalition 2 can profitably move to coalition 1, then do the move. The whole process requires $O(n)$ steps.

Let $n(s)$ be the number of agents having skill s. For every skill s such that $n(s) > 1$, place $\lceil \frac{n(s)}{2} \rceil$ agents in coalition 1, and $\lfloor \frac{n(s)}{2} \rfloor$ agents in coalition 2. For every skill s such that $n(s) = 1$, place the agent in coalition 1.

At the beginning coalition 1 has all the skills, so all the tasks are executed in coalition 1. Moreover, for every given skill, the number of agents in coalition 1 is at least the number of agents in coalition 2, but at most one more. Thus, in the initial state, no agent in coalition 1 wants to deviate to coalition 2. Only a member of coalition 2 can profitably deviate to coalition 1. This property holds also afterwards. Indeed, no agent i with skill s in coalition 1 has incentive to deviate if another agent i' with skill s' has profitably joined coalition 1. Indeed, if i and i' have the same skill, then it would mean that the deviation of i' was not profitable (i and i' have the same utility when i' joins coalition 1, and i would have the previous utility of i' if she goes to coalition 2). If i and i' have different skills, then the current utility of i does not change (no new task is executed), and her utility if she moves to coalition 2 can not be better without i'. □

The question whether the BRD converge when $q = 2$ (and agents are singleton) is an intriguing open problem. Note that the instance given in Proposition 1 contains 4 coalitions, so the proposition holds when $q \geq 4$.

7. Nash Stability with Singleton Tasks

In this section we suppose that every task requires exactly one skill. There is a one-to-one correspondence between skills and tasks: each skill s is associated with a task t_s of weight w_s requiring s. The agents can have more than one skill.

Unlike the previous section, we do not approach the existence of a Nash stable outcome starting from a social optimum. We show that the BRD always converge, proving that a Nash stable outcome always exists. However, we demonstrate that it is unlikely that a Nash stable outcome can be computed efficiently.

Theorem 4. *For every parameter q, the BRD always converge for hedonic skill games with singleton tasks.*

The proof of Theorem 4 consists of showing that every instance is a *congestion game* (Rosenthal, 1973; Monderer & Shapley, 1996). Congestion games are central to the field of algorithmic game theory (Nisan, Roughgarden, Tardos, & Vazirani, 2007; von Stengel, 2021). They always admit an exact potential function (namely, Rosenthal's potential function (Rosenthal, 1973)). As mentioned in the proof of Proposition 2, an exact potential function Φ associates a real value with every state. Each time an agent ℓ makes a profitable deviation that turns the current state σ into a new state σ', it holds that $\Phi(\sigma') - \Phi(\sigma) = u_\ell(\sigma') - u_\ell(\sigma)$. Every local optimum of Φ, defined as a state such that no single agent can profitably deviate, is Nash stable. Thus, the BRD are known to always converge in congestion games.

A congestion game consists of a set of agents, a set of resources R, and a function $g : R \times \mathbb{N} \to \mathbb{R}$. The strategy space of every agent is a collection of subsets of R. If exactly x agents have a given resource r in their strategy, then the gain associated with r for every user of r is $g(r, x)$. The payoff of an agent is the total sum of her gains, over the subset of resources contained in her strategy. Rosenthal's potential of σ is defined as $\Phi(\sigma) = \sum_{r \in R} \sum_{x=1}^{n_r(\sigma)} g(r, x)$, where $n_r(\sigma)$ is the number of agents having resource r in their strategies.

Proof of Theorem 4. Start from an instance of the hedonic skill game and create a set R of $q|\mathcal{S}|$ resources as follows: r_i^j with $i \in \{1, \ldots, |\mathcal{S}|\}$ and $j \in \{1, \ldots, q\}$. For every agent $\ell \in \mathcal{N}$, let $I(\ell)$ denote the indices of agent ℓ's skills, i.e., $I(\ell) := \{i \mid s_i \in S(\ell)\}$. The strategy space of every agent ℓ includes $\{r_i^j \mid i \in I(\ell)\}$ for every $j \in \{1, \ldots, q\}$. The function $g(r_i^j, x)$ is defined as w_i/x where w_i is the weight of the task requiring only skill s_i.

If an agent ℓ plays $\{r_i^j \mid i \in I(\ell)\}$ for some $j \in \{1, \ldots, q\}$, then this means that ℓ is in coalition C_j. If ℓ has skill s_i, and in total x agents having skill s_i belong to coalition C_j, then each of these agents gets w_i/x, which corresponds to the reward associated with the task solely requiring skill s_i. $\square$

The following corollary is a direct consequence of Theorem 4.

Corollary 1. *For every q, hedonic skill games with singleton tasks always admit a Nash stable outcome.*

One can directly use the BRD for computing a Nash stable outcome, but the number of steps before convergence is not guaranteed to be polynomial in the parameters of the game.

A different algorithm may be used for efficiently computing a Nash stable outcome but we show below, with tools from local search theory, that its existence is unlikely when $q = 2$.

Theorem 5. *Computing a Nash stable outcome in hedonic skill games with singleton tasks is a* **PLS***-complete problem when $q = 2$.*

Proof. MAX CUT FLIP is the problem of computing a solution of MAX CUT[9] such that flipping a vertex (i.e., moving it to the other part of the bi-partition) does not increase the total weight of the cut. Such a stable solution is called a local optimum and MAX CUT FLIP is known to be **PLS**-complete (Schäffer & Yannakakis, 1991).

The problem of computing a Nash stable outcome of the hedonic skill games with singleton tasks and $q = 2$ is clearly in **PLS** since one can easily build an initial solution (e.g., the grand coalition), and one can decide in polynomial time if it is profitable for an agent to deviate to the other coalition.

Let us show that MAX CUT FLIP is **PLS**-reducible to the computation of a Nash stable outcome of the hedonic skill game with singleton tasks and $q = 2$.

Every instance $\mathcal{I}$ of MAX CUT FLIP is mapped to an instance $f(\mathcal{I})$ of hedonic skill games with $q = 2$ as follows. Each vertex v_i is associated with an agent i. Each edge (v_i, v_j) corresponds to a skill s_{ij} and that skill is solely held by agents i and j. Each edge (v_i, v_j) also corresponds to a task requiring skill s_{ij} and the weight of the task is equal to the weight of the edge. Each part of the bi-partition corresponds to a coalition so that every state of $f(\mathcal{I})$ is naturally associated with a solution of $\mathcal{I}$. It remains to prove that every Nash stable outcome of $f(\mathcal{I})$ is a local optimum of $\mathcal{I}$.

In a Nash stable outcome, no agent can profitably deviate. The utility of agent i is equal to $\frac{w_{in}(i)}{2} + w_{out}(i)$ where $w_{in}(i)$ denotes the weight of the tasks associated with edges (v_i, v_j) such that agent j is in the same coalition as i, while $w_{out}(i)$ stands for the weight of the tasks associated with edges (v_i, v_k) such that agent k is not in the same coalition as i. If agent i deviates, then her new utility becomes $w_{in}(i) + \frac{w_{out}(i)}{2}$. Thus, every agent i satisfies $w_{out}(i) \geq w_{in}(i)$ in a Nash stable outcome.

The weight of the cut is $w_{out}(i) + w_{rest}(i)$ where $w_{rest}(i)$ denotes the total weight of the edges (v_j, v_k) such that $i \notin \{j, k\}$, and j and k are in distinct coalitions. If i deviates, then the weight of the cut becomes $w_{in}(i) + w_{rest}(i)$. Since $w_{out}(i) \geq w_{in}(i)$ holds for all i, we get that $w_{out}(i) + w_{rest}(i) \geq w_{in}(i) + w_{rest}(i)$. In other words, no vertex flipping increases the weight of the cut. □

Observe that building a Nash stable outcome is an easy task when the number of coalitions is unconstrained ($q = n$): if every agent is alone in her coalition, then nobody wants to deviate since everybody enjoys her maximum possible utility. Let us conclude this section with computational complexity of optimizing the social welfare of the hedonic skill game with singleton tasks.

Theorem 6. *Regarding the hedonic skill game with singleton tasks, maximizing the social welfare SW is an* **NP***-hard problem when $q = 2$, and polynomial time solvable when $q = n$*

---

[9] Given a simple graph $G = (V, E)$ with positive weights on its edges, find a bi-partition (V_1, V_2) of V such that the total weight of the edges having an endpoint in V_1 and the other endpoint in V_2, a.k.a. *the cut*, is maximized (Karp, 1972).

Proof. An instance of MAX CUT consists of a simple graph $G = (V, E)$ with positive weights on its edges. The objective is to find a bi-partition (V_1, V_2) of V such that the total weight of the edges having an endpoint in V_1 and the other endpoint in V_2, a.k.a. *the cut*, is maximized. MAX CUT is **NP**-hard (Karp, 1972).

Take an instance of MAX CUT and build an instance of the hedonic skill game with $q = 2$ as follows. Each edge e is associated with a skill denoted by s_e, together with a singleton task t_e requiring s_e. The weight w_e of t_e is equal to the weight of e. Each vertex $v_i \in V$ is associated with an agent denoted by i. The skills of i are the skills of the edges incident to v_i.

In any state of the skill game, each task is executed at least once and at most twice. A task t_e is executed twice if and only if the two agents having skill s_e choose distinct coalitions. The social welfare being equal to the total weight of the executed tasks, maximizing the social welfare is equivalent to maximizing the total weight of the tasks executed twice, which is exactly the weight of the cut. □

For instances with singleton tasks, observe that maximizing the social welfare is easy when the number of coalitions is unconstrained ($q = n$). Indeed, every agent's maximum utility is the total weight of the tasks requiring her skills, and every agent can reach this utility by choosing to be alone in a coalition.

8. Price of Anarchy

This section is devoted to the *price of anarchy* (PoA) of hedonic skill games. This is the largest value (over the entire set of instances of the game) taken by the maximum social welfare divided by the social welfare of the worst Nash stable state. We consider the PoA with respect to the utilitarian social welfare SW which is defined as the sum of the agents' utilities (cf. (1) in Section 3.3).

More formally, the price of anarchy of an instance I, denoted by PoA(I), is equal to $\frac{SW(\sigma^*)}{SW(\sigma)}$, where σ^* is a social optimum for I, and σ is the Nash stable outcome of I whose value $SW(\sigma)$ is minimum. The price of anarchy of a class of instances $\mathcal{Q}$ is defined as the largest value taken by PoA(I), over all instances $I \in \mathcal{Q}$.

Instances of hedonic skill games having non-singleton agents and non-singleton tasks at the same time are not considered because the existence of a Nash stable outcome is not guaranteed (cf. Example 2). Our results focus on the two extreme values of q, namely 2 and n. As a reminder, q is an upper bound on the number of coalitions that the agents can form.

When both agents and tasks are singleton, then the PoA is 1 (Proposition 4). The PoA of the remaining singleton agents instances depends on the the following parameter.

Definition 4. *Let τ be the maximum number of skills required by a task. Hence, $\tau = 1$ corresponds to singleton tasks.*

In a nutshell the PoA of hedonic skill games is as follows: $O(n)$ when $\tau = 2$ and $q = n$ (Theorem 7), $O(1)$ when $\tau \in \{2, 3\}$ and $q = 2$ (Theorem 8), unbounded otherwise (Theorems 7 and 8). Concerning the singleton tasks instances, we show that the PoA is $4/3$ when $q = 2$ (Theorem 9), and 1 when $q = n$. All the results are tight except for the case of singleton agents, $\tau = 2$ and $q = n$ where the result is asymptotically tight.

Proposition 4. *For the case of singleton tasks* and *singleton agents, the PoA of hedonic skill games is 1 for all q.*

Proof. The proof consists of showing that the singleton task associated with any given skill must be executed the maximum possible number of times in any Nash stable outcome. To do so, take any skill $s \in \mathcal{S}$, its corresponding singleton task t_s of weight w_s, and let $n(s)$ be the number of agents having skill s.

Suppose $q \geq n(s)$. The task t_s can be executed at most $n(s)$ times. Every agent having skill s can always choose a coalition where she is the only holder of s, thus maximizing her utility. Therefore, t_s is executed $n(s)$ times in any Nash stable outcome.

Suppose $q < n(s)$. The task t_s can be executed at most q times. In a Nash stable outcome, every coalition contains at least one agent having skill s. Suppose by contradiction that it is not the case. There must exist a coalition with at least two agents with s, so their utility is at most $w_s/2$. Moreover, there must exist a coalition without s to which one of these agents can deviate and have a larger utility of w_s. Therefore, t_s is executed exactly q times in any Nash stable outcome. $\square$

8.1 Singleton Agents

In this section we consider singleton agents. We first consider the case of $q = n$ (Theorem 7) and then the case of $q = 2$ (Theorem 8). Our bounds on the PoA in Theorem 8 (resp., Theorem 7) are tight (resp., asymptotically tight).

Theorem 7. *If $q = n$, then the PoA of hedonic skill games with singleton agents is $\Theta(n)$ when $\tau = 2$, and unbounded when $\tau > 2$.*

Proof. Let us begin with the case $\tau = 2$.

Fix any Nash stable outcome σ. We say that a task t is *active* in σ if there exists at least one coalition $C \in \sigma$ which is able to perform the task t, i.e., $S(t) \subseteq S(C)$, and *non-active* otherwise. Let us denote the set of active tasks in σ by $A(\sigma)$, and the set of non-active tasks in σ by $\overline{A(\sigma)} = \mathcal{T} \setminus A(\sigma)$. Note that in any Nash stable outcome, all the singleton tasks are active, while any task of size two (i.e., requiring two skills) can be either active or non-active. We get that

$$SW(\sigma) = \sum_{\ell \in \mathcal{N}} u_\ell(\sigma) \geq \sum_{t: t \in A(\sigma) \wedge |S(t)|=1} w(t) + \sum_{t: t \in A(\sigma) \wedge |S(t)|=2} w(t). \tag{5}$$

For any agent $\ell \in \mathcal{N}$, let $T_\ell(\sigma) = \{t \in \overline{A(\sigma)} : S(\ell) \subset S(t) \wedge |S(t)| = 2\}$ be the set of non-active tasks of σ of size two that require skill $S(\ell)$. Moreover, let $m_\ell(\sigma) = \arg\max_{t \in T_\ell(\sigma)} w(t)$ be the task belonging to $T_\ell(\sigma)$ of maximum weight. When $T_\ell(\sigma)$ is empty, we consider that $m_\ell(\sigma)$ is a dummy task of weight zero. It is easy to see that, for any agent $\ell \in \mathcal{N}$, it holds that $u_\ell(\sigma) \geq \frac{w(m_\ell(\sigma))}{2}$. Indeed, if $u_\ell(\sigma) < \frac{w(m_\ell(\sigma))}{2}$, then σ is not a Nash stable outcome since agent ℓ can perform an improving move (assuming that $m_\ell(\sigma)$ requires skills $\{S(\ell), S(a)\}$, for some agent a, the improving move of agent ℓ is to join any coalition having an agent with skill $S(a)$) by making the task $m_\ell(\sigma)$ active and, in this way, getting utility of at least $\frac{w(m_\ell(\sigma))}{2}$. Therefore, we get that

$$SW(\sigma) = \sum_{\ell \in \mathcal{N}} u_\ell(\sigma) \geq \sum_{\ell \in \mathcal{N}} \frac{w(m_\ell(\sigma))}{2}. \tag{6}$$

In an optimal coalition structure, any singleton task can be performed by at most n coalitions, and any active task of size two can be performed by at most $\frac{n}{2}$ coalitions. Moreover, let us denote by $SW(\overline{A(\sigma)})$ the social welfare that any optimal solution extracts from non-active tasks. We have

$$SW(\overline{A(\sigma)}) \leq \frac{1}{2}\sum_{\ell\in\mathcal{N}}\sum_{t\in T_\ell(\sigma)} w(t) \leq \frac{1}{2}(n-1)\sum_{\ell\in\mathcal{N}} w(m_\ell(\sigma)), \tag{7}$$

where the first inequality holds since each non-active task of size two is counted at least twice, and the second inequality holds since, for any agent $\ell \in \mathcal{N}$, there exist in σ at most $n-1$ non-active tasks of size two that require skill $S(\ell)$ (because there are at most n coalitions) and each of them has weight at most $w(m_\ell(\sigma))$.

Therefore, given any optimal coalition structure σ^*, we have that

$$\begin{aligned} SW(\sigma^*) = \sum_{\ell\in\mathcal{N}} u_\ell(\sigma^*) &\leq (n \sum_{t:t\in A(\sigma)\wedge|S(t)|=1} w(t)) + (\frac{n}{2} \sum_{t:t\in A(\sigma)\wedge|S(t)|=2} w(t)) + SW(\overline{A(\sigma)}) \leq \\ &\leq (n \sum_{t:t\in A(\sigma)\wedge|S(t)|=1} w(t)) + (\frac{n}{2} \sum_{t:t\in A(\sigma)\wedge|S(t)|=2} w(t)) + (\frac{1}{2}(n-1)\sum_{\ell\in\mathcal{N}} w(m_\ell(\sigma))) \end{aligned} \tag{8}$$

By combining inequalities (5), (6) and (8) we get $\frac{SW(\sigma^*)}{2SW(\sigma)} \leq \frac{(n\sum_{t:t\in A(\sigma)\wedge|S(t)|=1} w(t)) + (\frac{n}{2}\sum_{t:t\in A(\sigma)\wedge|S(t)|=2} w(t)) + (\frac{1}{2}(n-1)\sum_{\ell\in\mathcal{N}} w(m_\ell(\sigma))}{\sum_{t:t\in A(\sigma)\wedge|S(t)|=1} w(t) + \sum_{t:t\in A(\sigma)\wedge|S(t)|=2} w(t) + \sum_{\ell\in\mathcal{N}} \frac{w(m_\ell(\sigma))}{2}} \leq n$, which implies that

$$\frac{SW(\sigma^*)}{SW(\sigma)} \leq 2n. \tag{9}$$

Since Inequality (9) holds for any Nash stable outcome, and in particular for the one with worse (i.e., smallest) social welfare, the upper bound on the PoA follows.

For the lower bound ($\tau = 2$), suppose the number of agents n is even. There are $n/2$ agents with skill s_1 and $n/2$ agents with skill s_2. There is a single task t of weight 1 which requires both s_1 and s_2. The grand coalition is a Nash stable outcome in which every agent has utility $1/n$. If an agent deviates, then she is alone in a new coalition, and her utility is 0. The social welfare of the grand coalition is 1. Consider the coalition structure composed of $n/2$ coalitions, each one having an agent with s_1 and an agent with s_2. The social welfare of this state is $n/2$. Therefore, PoA $\geq \frac{n}{2}$.

Finally, consider the case $\tau > 2$. We are going to provide an instance where the PoA is unbounded. Suppose n is a multiple of τ. There are τ skills, and n/τ agents per skill. There is a single task of weight 1 requiring all the skills. The state where all the agents are isolated (i.e., they form singleton coalitions) is a Nash stable outcome of null social welfare. The optimal social welfare is reached when n/τ coalitions of size τ are formed (each such coalition contains all the skills). Thus, the PoA is unbounded. □

Theorem 8. *If $q = 2$, then the PoA of hedonic skill games with singleton agents is* 3 *when $\tau = 2$, PoA=* 5 *when $\tau = 3$, and the PoA is unbounded when $\tau > 3$.*

tasks	skills	weight
t_1	$\{s_1, s_2\}$	2
t_2	$\{s_3, s_4\}$	2
t_3	$\{s_1, s_4\}$	4
t_4	$\{s_2, s_3\}$	4

agents	skills
1 and 2	$\{s_1\}$
3 and 4	$\{s_2\}$
5 and 6	$\{s_3\}$
7 and 8	$\{s_4\}$

Table 8: Instance used in Theorem 8 ($\tau = 2$).

Proof. Let us start with the case $\tau = 2$. For the lower bound on the PoA, consider an instance with 8 agents and 4 skills $\{s_1, s_2, s_3, s_4\}$ as described in Table 8. There are 2 agents per skill. There is a task of weight 2 requiring s_1 and s_2, a task of weight 2 requiring s_3 and s_4, a task of weight 4 requiring s_1 and s_4, and a task of weight 4 requiring s_2 and s_3. In an optimal state, each coalition has the 4 skills, and the social welfare is 24. The state where all the agents with skills s_1 and s_4 are on one side, and all the agents with skills s_2 and s_3 are on the other side, is a Nash equilibrium with social welfare 8. The PoA in this case is $24/8 = 3$.

For the upper bound on the PoA, consider a NE σ and a social optimum σ^*. Since there are at most two coalitions, a task can be executed at most twice. Let $W(a, b, c)$ be the total weight of the tasks requiring exactly $a \in \{1, 2\}$ skills, executed $b \in \{0, 1, 2\}$ times in σ and executed $c \in \{0, 1, 2\}$ times in σ^*. Note that

$$W(1, 0, 1) = W(1, 0, 2) = 0 \tag{10}$$

holds because a task requiring a single skill executed at least once in σ^* must also be executed at least once in σ.

By definition, $\sum_{i\in\mathcal{N}} u_i(z)$ is equal to the utilitarian social welfare of any state z. Thus,

$$\sum_{i\in\mathcal{N}} u_i(\sigma) = \sum_{a=1}^{\tau}\sum_{b=1}^{2}\sum_{c=0}^{2} b \cdot W(a, b, c), \tag{11}$$

$$\sum_{i\in\mathcal{N}} u_i(\sigma^*) = \sum_{a=1}^{\tau}\sum_{b=0}^{2}\sum_{c=1}^{2} c \cdot W(a, b, c). \tag{12}$$

Since σ is a Nash equilibrium, $u_i(\sigma) \geq u_i(\sigma_{-i}, \bar{\sigma}_i)$ holds for all $i \in \mathcal{N}$ where $\bar{\sigma}_i$ denotes the (unique) coalition where i is not under σ. Summing up this inequality over $\mathcal{N}$ gives

$$\sum_{i\in\mathcal{N}} u_i(\sigma) \geq \sum_{i\in\mathcal{N}} u_i(\sigma_{-i}, \bar{\sigma}_i). \tag{13}$$

Note that

$$\sum_{i\in\mathcal{N}} u_i(\sigma_{-i}, \bar{\sigma}_i) \geq W(2, 0, 1) + 2W(2, 0, 2) \tag{14}$$

holds. Indeed, take a task t requiring two skills (say s_1 and s_2 w.l.o.g.) executed once in σ^* but none in σ. If, starting from σ, the agent with skill s_1 moves then her utility is at least half of the weight of t. The same goes for the agent with skill s_2. Therefore $\sum_{i\in\mathcal{N}} u_i(\sigma_{-i}, \bar{\sigma}_i)$

covers the weight of any task like t (namely, twice $w(t)/2$), and the total weight of such tasks is $W(2,0,1)$. Now, suppose t is executed twice in σ^* but none in σ (we still assume that t requires s_1 and s_2). Because t is executed twice in σ^*, there must be at least two agents with skill s_1 and at least two other agents with skill s_2. Since t is not executed in σ, all the s_1 agents are in the same coalition, and all the s_2 agents are in the other coalition. If, starting from σ, an agent with skill s_1 (resp., s_2) moves, then her utility is at least half of the weight of t. Therefore $\sum_{i\in\mathcal{N}} u_i(\sigma_{-i},\bar{\sigma}_i)$ covers twice the weight of any task like t (namely, four times $w(t)/2$), and the total weight of such tasks is $W(2,0,2)$.

It follows from (13) and (14) that

$$\sum_{i\in\mathcal{N}} u_i(\sigma) \geq W(2,0,1) + 2W(2,0,2). \tag{15}$$

Using (11), we know that

$$\sum_{i\in\mathcal{N}} u_i(\sigma) \geq \sum_{a=1}^{2}\sum_{b=1}^{2}\sum_{c=1}^{2} b\cdot W(a,b,c) \tag{16}$$

holds (compared to (11), the terms where $c=0$, which have a non negative weight, are missing). Multiply (16) by two and add it to (15) to get that

$$3\sum_{i\in\mathcal{N}} u_i(\sigma) \geq W(2,0,1) + 2W(2,0,2) + 2\sum_{a=1}^{2}\sum_{b=1}^{2}\sum_{c=1}^{2} b\cdot W(a,b,c). \tag{17}$$

Use (12) and (10) to observe that the right hand part of Inequality (17) is greater than or equal to $\sum_{i\in\mathcal{N}} u_i(\sigma^*)$. Indeed, each term $W(a,b,c)$ appears exactly c times in (12) except for $W(1,0,1)$ and $W(1,0,2)$ which are equal to zero by (10). In the right hand part of (17), each possible term $W(a,b,c)$ appears at least c times, except for $W(1,0,1)$ and $W(1,0,2)$. Therefore, we get that $3\sum_{i\in\mathcal{N}} u_i(\sigma) \geq \sum_{i\in\mathcal{N}} u_i(\sigma^*)$, proving that the PoA is at most 3 when $\tau = 2$.

Let us move on to the case $\tau = 3$. For the lower bound on the PoA, consider an instance with 10 agents and 5 skills $\{s_1, s_2, s_3, s_4, s_5\}$; there are 2 agents per skill. There is a task of weight 6 requiring $\{s_1, s_2, s_3\}$, a task of weight 3 requiring $\{s_i, s_4, s_5\}$ for all $i \in \{1,2,3\}$, a task of weight 4 requiring $\{s_4, s_5\}$, and a task of weight 3 requiring $\{s_1, s_2, s_i\}$ for all $i \in \{4,5\}$. In an optimal state, each coalition has the 5 skills, and the social welfare is 50. The state where all the agents with skills s_1, s_2 and s_3 are on one side, and all the agents with skills s_4 and s_5 are on the other side, is a Nash equilibrium with social welfare 10. The PoA in this case is $50/10 = 5$.

For the upper bound on the PoA, we use the same notations and tools as for the case $\tau = 2$. Note that

$$\sum_{i\in\mathcal{N}} u_i(\sigma_{-i},\bar{\sigma}_i) \geq W(2,0,1) + 2W(2,0,2) + \frac{1}{3}\left(W(3,0,1) + 2W(3,0,2)\right) \tag{18}$$

holds. The justification of $\sum_{i\in\mathcal{N}} u_i(\sigma_{-i},\bar{\sigma}_i) \geq W(2,0,1) + 2W(2,0,2)$ is given above (case $\tau = 2$) when (14) has been proved. For the term $W(3,0,1)/3$, take a task t requiring three skills (say $\{s_1, s_2, s_3\}$ w.l.o.g.) executed once in σ^* but none in σ. Since t is not executed in σ, the skills are separated so that two of them are in one coalition, and the third one is in the other coalition. Suppose w.l.o.g. that s_1 is in one coalition while s_2 and s_3 are in the other coalition. If, starting from σ, the agent with skill s_1 moves, then her utility is at least one third of the weight of t. That is why $\sum_{i \in \mathcal{N}} u_i(\sigma_{-i}, \bar{\sigma}_i)$ covers one third of the weight of any task like t (namely, $W(3, 0, 1)/3$).

For the term $2W(3, 0, 2)/3$, take a task t requiring three skills (say $\{s_1, s_2, s_3\}$ w.l.o.g.) executed twice in σ^* but none in σ. Since t is executed twice in σ^*, there are at least two agents per skill in $\{s_1, s_2, s_3\}$. Since t is not executed in σ, the skills are separated so that two of them are in one coalition, and the third one is in the other coalition. If, starting from σ, the agent with the singled out skill moves, then t can be executed and the agent's utility is at least one third of the weight of t. That is why $\sum_{i \in \mathcal{N}} u_i(\sigma_{-i}, \bar{\sigma}_i)$ covers one third of the weight of any task like t (namely, $2W(3, 0, 2)/3$ because there are two agents having a singled out skill).

Using (11), we know that

$$\sum_{i \in \mathcal{N}} u_i(\sigma) \geq \sum_{a=1}^{3} \sum_{b=1}^{2} \sum_{c=1}^{2} b \cdot W(a, b, c) \tag{19}$$

holds. Add (19) multiplied by two to (18) multiplied by three to get that

$$5 \sum_{i \in \mathcal{N}} u_i(\sigma) \geq 3W(2, 0, 1) + 6W(2, 0, 2) + W(3, 0, 1) + 2W(3, 0, 2) + 2 \sum_{a=1}^{3} \sum_{b=1}^{2} \sum_{c=1}^{2} b \cdot W(a, b, c). \tag{20}$$

Use (12) to observe that the right hand part of Inequality (20) is greater than or equal to $\sum_{i \in \mathcal{N}} u_i(\sigma^*)$. Indeed, each term $W(a, b, c)$ appears exactly c times in (12) except for $W(1, 0, 1)$ and $W(1, 0, 2)$ which are equal to zero by (10). In the right hand part of (20), each possible term $W(a, b, c)$ appears at least c times, except for $W(1, 0, 1)$ and $W(1, 0, 2)$. We get that $5 \sum_{i \in \mathcal{N}} u_i(\sigma) \geq \sum_{i \in \mathcal{N}} u_i(\sigma^*)$, proving that the PoA is at most 5.

Finally, we suppose that $\tau > 3$. We are going to provide an instance where the PoA is unbounded. Consider the instance where each agent i has her own skill s_i. There is a single task of weight 1 requiring all the skills. The state where agents 1 and n are in the first coalition, whereas all the other agents are in the second coalition, is a NE with null social welfare. Indeed, $\tau > 3$ implies that there are at least two agents in the second coalition, and no agent can unilaterally deviate so that the unique task is executed. The grand coalition is a state with social welfare 1. Therefore, the PoA is unbounded. □

8.2 Singleton Tasks

In this section we consider singleton tasks. We first notice that when the number of coalitions is unconstrained ($q = n$), the PoA of hedonic skill games with singleton tasks is trivially equal to 1. This is because every agent can reach her maximum possible utility if she chooses to be alone in a coalition, and such a state is Nash stable. In the following theorem we deal with the case of $q = 2$ and provide a tight analysis of the PoA.

Theorem 9. *The PoA of hedonic skill games with singleton tasks is* $4/3$ *when* $q = 2$.

tasks	skills	weight
t_1	$\{s_1\}$	1
t_2	$\{s_2\}$	1
t_3	$\{s_3\}$	1
t_4	$\{s_4\}$	1

agents	skills
1	$\{s_1, s_2\}$
2	$\{s_2, s_3\}$
3	$\{s_1, s_4\}$
4	$\{s_3, s_4\}$

Table 9: Instance used in Theorem 9.

Proof. Fix an instance and consider a Nash stable outcome σ and a socially optimal outcome σ^*. Suppose there is a skill s that only one agent, say i, owns. The instance can be modified by deleting skill s because the PoA can only increase, and σ remains a Nash stable outcome (only the utility of agent i is decreased by the weight of the task associated with s, for every strategy that i can take). Therefore, we can suppose w.l.o.g. that no skill is owned by a single agent.

Let W_1 (resp., W_2) be the total weight of the tasks executed once (resp., twice) under σ. Let us prove the following intermediate result.

$$W_2 \geq W_1. \tag{21}$$

Since σ is Nash stable, no deviation is profitable. For every agent i, we know that $u_i(\sigma) \geq u_i(\sigma_{-i}, 3 - \sigma_i)$.[10] Summing up this inequality over the set $\tilde{\mathcal{N}}$ of agents having a skill whose corresponding task is executed once under σ, we get an inequality $W_A \geq W_B$. W_A is the sum of the utilities of the agents in $\tilde{\mathcal{N}}$ under σ, and W_B is the sum of the utilities of the agents in $\tilde{\mathcal{N}}$ if they unilaterally deviate from σ. Note that W_A is upper bounded by the total weight of the tasks executed under σ, i.e., $W_1 + W_2 \geq W_A$. Since we have excluded skills owned by a single agent, we know that the weight of the tasks executed once under σ is shared by at least two agents. It follows that $W_B \geq 2W_1$ because for each task executed once, at least two agents get its full weight if she unilaterally deviates. We obtain $W_1 + W_2 \geq W_A \geq W_B \geq 2W_1 \Leftrightarrow$ (21).

We have $SW(\sigma^*) \leq 2(W_1 + W_2)$ because a task is executed at most twice ($q = 2$), and $SW(\sigma) = W_1 + 2W_2$. Use (21) multiplied by $\frac{1}{2}$ to get that $SW(\sigma) = W_1 + \frac{3}{2}W_2 + \frac{1}{2}W_2 \geq \frac{3}{2}(W_1 + W_2)$. Therefore, the PoA $\frac{SW(\sigma^*)}{SW(\sigma)}$ is upper bounded by $\frac{2(W_1+W_2)}{\frac{3}{2}(W_1+W_2)} = \frac{4}{3}$.

To conclude, let us give an instance where the PoA is exactly 4/3 (cf. Table 9). There are 4 tasks with unit weight associated with 4 skills $\{s_1, s_2, s_3, s_4\}$, and 4 agents. Agents 1, 2, 3, and 4 have skills $\{s_1, s_2\}$, $\{s_2, s_3\}$, $\{s_1, s_4\}$, and $\{s_3, s_4\}$, respectively. If agents 1 and 2 are together in the first coalition, while agents 3 and 4 are together in the second coalition, then the state is a Nash stable outcome of social welfare 6. If agents 1 and 4 are in the same coalition, while agents 2 and 3 are in another coalition, then the social welfare is 8. Thus, the PoA of this instance is 4/3. $\square$

---

[10] Here we assume that $\sigma_i \in \{1, 2\}$, corresponding to the choice of coalition C_1 or coalition C_2, and $3 - \sigma_i$ simply designates the other strategy (i.e., 1 when $\sigma_i = 2$, and 2 when $\sigma_i = 1$). Moreover, (σ_{-i}, x) is the standard notation for the strategy profile σ' where $\sigma'_j = \sigma_i$ when $j \neq i$, and $\sigma'_j = x$ when $j = i$.

9. Conclusion and Future Work

In this article, we have focused on hedonic skill games and analyzed the existence, efficiency, and computation of Nash stable outcomes. Deciding whether a general instance admits a Nash stable outcome is an NP-complete problem when $q \geq 3$. A Nash stable outcome exists for every singleton agent instance, and it can be computed in polynomial time, but natural dynamics like the BRD can cycle. Nash stable outcomes also exist in singleton task instances, and the game admits a potential, meaning that the BRD always converge in this case, but it is unlikely that a polynomial algorithm can compute a Nash stable outcome.

Several research directions that are worth investigating arise from this work. For instance, what is the difficulty of deciding whether an instance of the hedonic skill game admits a Nash stable outcome when $q = 2$? What is the exact PoA for singleton agents instances when $q = n$ and $\tau = 2$? In many cases we focused on $q \in \{2, n\}$, but examining the same problems for any value of q leaves many open questions. A lot of problems related to the better response dynamics remain unresolved when a Nash stable outcome exists. We have seen that the BRD always converge if the tasks are singleton, but it can cycle for singleton agents. Special singleton agents instances, especially when the number of coalitions is small (e.g., $q = 2$), deserve attention. Another interesting question deals with the convergence of the *best* response dynamics, in which every deviation is a best response instead of a better response.

We believe that it would also be useful to consider the ϵ-Nash stable outcomes, where $\epsilon \geq 1$. They are outcomes where no agent can improve her utility by a multiplicative factor strictly greater than ϵ (Chien & Sinclair, 2011; Caragiannis, Fanelli, Gravin, & Skopalik, 2015). This is particularly interesting for the setting with non-singleton agents for which the existence of a Nash stable outcome is not guaranteed or, it is hard to compute it (as in the case of singleton tasks). Other solution concepts can be considered for hedonic skill games, such as individual stability, contractual Nash stability, or contractual individual stability (Aziz & Savani, 2016).

We have adopted the classic utilitarian social welfare, however, other kinds of social welfare could be considered, like for instance, the egalitarian social welfare which is defined as the minimum utility among all the agents' utilities, and the Nash social welfare which is defined as the product of the agents' utilities.

Finally, we think that it would be important to consider different rewarding schemes (i.e., alternative ways to define the non-transferable utility of the agents) that can guarantee the existence of stable outcomes and possibly show a lower price of anarchy. Indeed, we considered a fair division of the weights of the tasks performed by a coalition, but alternative ways to capture fairness exist and can be studied in the future. In addition, other ways to define the non-transferable utility of the agents can derive from the context, i.e., a real-life situation different from the ones listed in the introduction, and modeled as a hedonic skill game.

Acknowledgments

A preliminary version of this work appeared in the proceedings of the 23rd International Conference on Autonomous Agents and Multi-Agent Systems, AAMAS 2024 (Gourvès &

Monaco, 2024).
Laurent Gourvès is supported by Agence Nationale de la Recherche (ANR), project THEMIS ANR-20-CE23-0018.
Gianpiero Monaco is supported by PNRR MIUR project FAIR - Future AI Research (PE00000013), Spoke 9 - Green-aware AI.

Appendix A. Proofs

Theorem 10. *For every parameter q, finding a coalition structure which maximizes the utilitarian social welfare can be done in $|\mathcal{S}|q$ steps if all the agents are singleton.*

Proof. For every skill $s \in \mathcal{S}$, let $n(s)$ be the total number of agents having skill s in the instance. The Euclidean division of $n(s)$ by q gives a quotient α_s and a remainder β_s, i.e., $n(s) = \alpha_s q + \beta_s$ for all $s \in \mathcal{S}$, where $\beta_s < q$.

Create q disjoint sets of agents $C_1, \dots, C_q$ as follows: for every skill s, sets C_1 to C_{β_s} contain exactly α_s+1 agents of skill s each (chosen arbitrarily), and sets C_{β_s+1} to C_q contain exactly α_s agents of skill s each. Note that for some instances, some sets $C_{q'+1}$ to C_q can be empty. The coalition structure consists of the non-empty sets $(C_1, \dots, C_{q'})$ which are clearly built in linear time.

If $\alpha_s > 0$, then every coalition in $(C_1, \dots, C_{q'})$ contains at least one member having skill s. Otherwise, only coalitions C_1 to C_{β_s} contain a member having skill s. Let $h(s) = q$ when $\alpha_s > 0$, otherwise $h(s) = \beta_s$. For every subset of skills $Y \subseteq \mathcal{S}$, let $\mu(Y)$ denote $\min_{s \in Y} h(s)$. Each task $t \in \mathcal{T}$ can be performed at most $\mu(S(t))$ times because at most $\mu(S(t))$ coalitions can contain all the skills that t requires. By construction of $(C_1, \dots, C_{q'})$, for every non-empty $Y \subseteq \mathcal{S}$, we have $Y \subseteq C_i$ for $i = 1..\mu(Y)$. Thus, coalitions C_1 to $C_{\mu(S(t))}$ contain the skills $S(t)$ for every task t. In other words, every task t is performed its maximum possible number of times $\mu(S(t))$, implying that $(C_1, \dots, C_{q'})$ is a social optimum. □

Proof of Lemma 1 stating that the initial state of D satisfies Invariant (4)

Proof. By contradiction, suppose the initial state admits a better right move for some agent of skill s_i, departing from C_a and going to C_b such that $a < b$. Since Y is made of consecutive 1s followed by consecutive 0s, the better right move would imply that either $\frac{L_{i,a}}{1} < \frac{L_{i,b}}{1}$ or $\frac{L_{i,a}}{1} < \frac{L_{i,b}}{2}$ (depending on whether $y_{i,b} = 0$ or $y_{i,b} = 1$ in the initial state). Both cases are impossible because they contradict Invariant (3). □

Proof of Lemma 2 stating that the dynamics D maintain Invariants (2) and (3)

Proof. Let us first observe that Invariant (2) implies Invariant (3). Indeed, Invariant (2) states that if coalition C_{j+1} has at least one agent having skill s_i, then C_j also contains at least one agent having skill s_i. This means that $S(C_j) \supseteq S(C_{j+1})$. We deduce that $T(C_j) \supseteq T(C_{j+1})$, and thus $L_{i,j} \geq L_{i,j+1}$. Therefore, it suffices to show that every step of D maintains Invariant (2) in order to guarantee that both invariants are maintained.

Consider any best left move $C_a \overset{\text{best } i}{\leftarrow} C_b$ of the dynamics D. Suppose for the sake of contradiction that $y_{i,a-1} < y_{i,a}$ holds after that move.[11] Since Invariant (2) was true before

---

[11] When $a = 1$, the invariant cannot be destroyed at coalition C_a since coalition C_{a-1} does not exist.

the move, we had $y_{i,a-1} = y_{i,a}$ beforehand. Since Invariant (3) was true before the move, we have $L_{i,a-1} \geq L_{i,a}$. Thus, $C_{a-1} \overset{\text{best } i}{\leftarrow} C_b$ was also a best left move. By item 1 of the dynamics D, $C_a \overset{\text{best } i}{\leftarrow} C_b$ should have been discarded because of the presence of $C_{a-1} \overset{\text{best } i}{\leftarrow} C_b$, leading to a contradiction.

Now suppose that $y_{i,b} < y_{i,b+1}$ holds after move $C_a \overset{\text{best } i}{\leftarrow} C_b$ of the dynamics D.[12] Since Invariant (2) was true before the move, we had $y_{i,b} = y_{i,b+1}$ beforehand. Since Invariant (3) was true before the move, we have $L_{i,b} \geq L_{i,b+1}$. Thus, $C_a \overset{\text{best } i}{\leftarrow} C_{b+1}$ was also a best left move. By item 1 of the dynamics D, $C_a \overset{\text{best } i}{\leftarrow} C_b$ should have been discarded because of the presence of $C_a \overset{\text{best } i}{\leftarrow} C_{b+1}$, leading to a contradiction. □

Proof of Lemma 3 stating that a best left move does not trigger the existence of a better right move, i.e., Invariant (4) is maintained

Proof. Let us show that a best left move $C_a \overset{\text{best } i}{\leftarrow} C_b$ by some agent ℓ of skill s_i does not trigger the existence of a better right move for some agent ℓ' of skill $s_{i'}$. By contradiction, we suppose that a better right move from C_c to C_d, where $c < d$, exists after $C_a \overset{\text{best } i}{\leftarrow} C_b$ is performed (hence $a < b$). Let us denote the better right move by $C_c \overset{\text{better } i'}{\rightarrow} C_d$. The proof consists of two cases.

- Case $i = i'$. If $c = b$, then $a < b = c < d$ and the right move was already possible before $C_a \overset{\text{best } i}{\leftarrow} C_b$ (the fact that ℓ has left C_b improved the situation for an agent of C_b who has the same skill as ℓ), contradicting Invariant (4). Thus, $c \neq b$. Suppose $c = a$, and then $d > a$. It cannot be $d = b$ ($C_c \overset{\text{better } i'}{\rightarrow} C_d$ is not profitable since it is the exact opposite of $C_a \overset{\text{best } i}{\leftarrow} C_b$). If $d > b$, then composing $C_a \overset{\text{best } i}{\leftarrow} C_b$ with $C_c \overset{\text{better } i'}{\rightarrow} C_d$ means that agent ℓ had a better right move from C_b to C_d, contradicting Invariant (4). The last sub-case of $c = a$ is $a < d < b$: composing $C_a \overset{\text{best } i}{\leftarrow} C_b$ with $C_c \overset{\text{better } i'}{\rightarrow} C_d$ means that agent ℓ had an improving left move from C_b to C_d which outperformed $C_a \overset{\text{best } i}{\leftarrow} C_b$, i.e., $C_a \overset{\text{best } i}{\leftarrow} C_b$ was not a *best* left move, contradiction.

It remains to study the case where $c \notin \{a, b\}$. Suppose $d = a$, thus $c < a$. The right move was already possible before $C_a \overset{\text{best } i}{\leftarrow} C_b$ (the fact that ℓ has joined C_a deteriorated the situation for an agent who has the same skill as ℓ and wants to join C_a), contradicting Invariant (4). Suppose $d \notin \{a, b\}$, (and of course $d \neq c$). The right move is completely independent of $C_a \overset{\text{best } i}{\leftarrow} C_b$, meaning that it was already possible before $C_a \overset{\text{best } i}{\leftarrow} C_b$, contradicting Invariant (4).

Suppose $d = b$, thus $c < b$ but $c \neq a$. If $c < a$, then a right better move from C_c to C_b was already possible before $C_a \overset{\text{best } i}{\leftarrow} C_b$ is performed, contradicting Invariant (4). Indeed, an agent ℓ' in C_c with skill s_i joining C_b after ℓ has left C_b gets the utility agent ℓ had before deviating. Then, if moving to C_b is an improvement for ℓ', then going directly to C_a is even better because ℓ improves her utility by moving from C_b to C_a. The final case is $a < c < d = b$. Using the previous arguments (i.e., the new utility of ℓ' is the old utility of ℓ who could profitably move to C_a), we get that it was profitable for agent ℓ' with skill s_i to leave C_c and join C_a before performing $C_a \overset{\text{best } i}{\leftarrow} C_b$. However, the dynamics D favor

---

[12] If b is the index of the last coalition, then the invariant cannot be destroyed at coalition C_b since coalition C_{b+1} does not exist.

left best moves whose departing coalition has smallest index (see item 2). Here, the index of the departing coalition of ℓ' is smaller than the one of ℓ. Since Invariant (4) held before performing $C_a \overset{\text{best } i}{\leftarrow} C_b$, ℓ' had a best left move. The only possibility is that the best left move of ℓ' was discarded during item 1 of the dynamics. The best left move of ℓ' was discarded in favor of another best left move with departing coalition having a larger index, but the number of agents with the same skill had to be the same. Using Invariant (2) and the fact that $C_a \overset{\text{best } i}{\leftarrow} C_b$ was chosen as the next deviation of the dynamics D, we deduce that $y_{i,c} = y_{i,b}$ was true[13] before executing $C_a \overset{\text{best } i}{\leftarrow} C_b$. However, Invariant (3) guarantees that $L_{i,c} \geq L_{i,b}$ was true before executing $C_a \overset{\text{best } i}{\leftarrow} C_b$. This means that $\frac{L_{i,c}}{y_{i,c}} = \frac{L_{i,c}}{y_{i,b}} \geq \frac{L_{i,b}}{y_{i,b}} = \frac{L_{i,d}}{y_{i,d}}$. We eventually get a contradiction with the fact that agent ℓ' moving right from C_c to C_d after $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed is profitable (by assumption, it is a better right move) because the utility of agent ℓ' is $\frac{L_{i,c}}{y_{i,c}}$ before the move, and $\frac{L_{i,d}}{y_{i,d}}$ after it.

- Case $i' \neq i$. Suppose by contradiction that a best left move $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed by an agent of skill s_i and it creates an incentive for agent ℓ' with a different skill $s_{i'}$ to move from C_c to C_d such that $c < d$. Since so far, all the deviations were done to the left, and using Invariant (2), we observe that no entry of L has increased.[14] The fact that agent ℓ' did not want to move to the right before $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed (true by Invariant (4)) but wants to do so right after $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed, can only be due to the fact that $L_{i',c}$ has strictly decreased while $L_{i',d}$ has not changed because $C_a \overset{\text{best } i}{\leftarrow} C_b$ cannot simultaneously decrease both $L_{i',c}$ and $L_{i',d}$.[15] Since $s_i \neq s_{i'}$, the decrease of $L_{i',c}$ is due to the fact that ℓ has left C_c (thus, $C_b = C_c$) and she was the only agent with skill s_i in C_c (i.e., $y_{i,c}$ was equal to 1 before $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed). If $y_{i',c} = 1$, then it is not difficult to see that ℓ' has no incentive to do a right move. Indeed, $y_{i',d} \in \{0, 1\}$ by Invariant (2), and $L_{i',c} \geq L_{i',d}$ by Invariant (3). Thus, moving from C_c to C_d gives a utility of at most $L_{i',d}$, which is not larger than the current utility $L_{i',c}$. Therefore, we can suppose that $y_{i',c} > 1$, meaning that between the initial state and the current one (i.e., right after $C_a \overset{\text{best } i}{\leftarrow} C_b$ is completed), some agent with skill $s_{i'}$ has made a best left move, say $C_c \overset{\text{best } i'}{\leftarrow} C_e$ with $a < b = c < e$. Since the dynamics D favor best left moves having the smallest index of departing coalition, $C_a \overset{\text{best } i}{\leftarrow} C_b$ should have been preferred to $C_c \overset{\text{best } i'}{\leftarrow} C_e$ if they were both applicable. This means that $C_a \overset{\text{best } i}{\leftarrow} C_b$ was not applicable when $C_c \overset{\text{best } i'}{\leftarrow} C_e$ has been completed. More generally, no best left move whose departing coalition index is at most c was applicable when $C_c \overset{\text{best } i'}{\leftarrow} C_e$ has been completed. However, a best left move (namely, $C_a \overset{\text{best } i}{\leftarrow} C_b$) has been executed afterwards. Therefore, we need to analyze how the incentive to do $C_a \overset{\text{best } i}{\leftarrow} C_b$ appeared.

The reason comes from at least one of the following two situations: the utility at the departing coalition has decreased, or the prospective utility at the arrival coalition has increased. The first situation occurs if (i) the number of agents with skill s_i has increased

---

[13] By transitivity, the best left move of ℓ' is discarded in favor of a move whose number of agents with skill s_i in its departing coalition is $y_{i,c}$, and that move can also be discarded in favor of a move whose number of agents with skill s_i in its departing coalition is $y_{i,c}$, and so on, until we reach the departing coalition of the move that is actually performed, that is $C_a \overset{\text{best } i}{\leftarrow} C_b$, meaning that C_b had $y_{i,c}$ agents with skill s_i.

[14] A coalition can loose a skill, but no coalition gains a new skill.

[15] It is a single agent move.

in C_b, or (*ii*) the last agent whose skill s_h is different from s_i has left C_b, implying that a task involving both s_h and s_i is not executed in C_b anymore. The second situation occurs if (*iii*) the number of agents with skill s_i has decreased in C_a, or (*iv*) $L_{i,a}$ has increased.

Case (*i*) is not compatible with the dynamics D because the move, say $C_b \overset{\text{best } i}{\leftarrow} C_e$, that created the incentive to do $C_a \overset{\text{best } i}{\leftarrow} C_b$ could have been composed with $C_a \overset{\text{best } i}{\leftarrow} C_b$. Indeed, the dynamics D would have executed $C_a \overset{\text{best } i}{\leftarrow} C_e$ instead of $C_b \overset{\text{best } i}{\leftarrow} C_e$ because $a < b$ (see the first item of the dynamics). Case (*ii*) is not possible because $b = c$, and we have seen that no best left move whose departing coalition index is at most c was applicable when $C_c \overset{\text{best } i'}{\leftarrow} C_e$ has been completed. Here, we use the argument recursively, i.e., creating the incentive to do a left move with departing coalition index at most c requires the execution (and thus, its existence) of a best left move with departing coalition index at most c. The same argument excludes case (*iii*) because $a < b = c$. Finally, case (*iv*) is impossible because, as already mentioned, no entry of L increases when only left moves are performed.

In all, the appearance of the incentive to do the best left move $C_a \overset{\text{best } i}{\leftarrow} C_b$ after $C_c \overset{\text{best } i'}{\leftarrow} C_e$ is executed is not possible under the dynamics D. $\square$

References

Anshelevich, E., Dasgupta, A., Kleinberg, J. M., Tardos, É., Wexler, T., & Roughgarden, T. (2008). The price of stability for network design with fair cost allocation. *SIAM J. Comput.*, *38*(4), 1602–1623.

Aziz, H., & de Keijzer, B. (2011). Complexity of coalition structure generation. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 191–198. IFAAMAS.

Aziz, H., & Savani, R. (2016). Hedonic games. In Brandt, F., Conitzer, V., Endriss, U., Lang, J., & Procaccia, A. D. (Eds.), *Handbook of Computational Social Choice*, pp. 356–376. Cambridge University Press, USA.

Bachrach, Y., Meir, R., Jung, K., & Kohli, P. (2010). Coalitional structure generation in skill games. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI)*, pp. 703–708. AAAI Press.

Bachrach, Y., Parkes, D. C., & Rosenschein, J. S. (2013). Computing cooperative solution concepts in coalitional skill games. *Artificial Intelligence*, *204*, 1–21.

Bachrach, Y., & Rosenschein, J. S. (2008). Coalitional skill games. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1023–1030. IFAAMAS.

Banerjee, S., Konishi, H., & Sönmez, T. (2001). Core in a simple coalition formation game. *Social Choice and Welfare*, *18*(1), 135–153.

Bilò, V., Fanelli, A., Flammini, M., Monaco, G., & Moscardelli, L. (2019). Optimality and Nash stability in additive separable generalized group activity selection problems. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 102–108. ijcai.org.

Bogomolnaia, A., & Jackson, M. O. (2002). The stability of hedonic coalition structures. *Games and Economic Behavior*, *38*(2), 201–230.

Caragiannis, I., Fanelli, A., Gravin, N., & Skopalik, A. (2015). Approximate pure Nash equilibria in weighted congestion games: Existence, efficient computation, and structure. *ACM Trans. Economics and Comput.*, *3*(1), 2:1–2:32.

Caskurlu, B., Kizilkaya, F. E., & Ozen, B. (2024). Hedonic expertise games. *Ann. Math. Artif. Intell.*, *92*(3), 671–690.

Cechlárová, K., & Romero-Medina, A. (2001). Stability in coalition formation games. *International Journal of Game Theory*, *29*(4), 487–494.

Chien, S., & Sinclair, A. (2011). Convergence to approximate Nash equilibria in congestion games. *Games Econ. Behav.*, *71*(2), 315–327.

Darmann, A., Elkind, E., Kurz, S., Lang, J., Schauer, J., & Woeginger, G. J. (2018). Group activity selection problem with approval preferences. *Int. J. Game Theory*, *47*(3), 767–796.

Drèze, J. H., & Greenberg, J. (1980). Hedonic coalitions: Optimality and stability. *Econometrica*, *48*(4), 987–1003.

Elkind, E., & Rothe, J. (2016). Cooperative game theory. In Rothe, J. (Ed.), *Economics and Computation, An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, Springer texts in business and economics, pp. 135–193. Springer.

Gale, D., & Shapley, L. S. (1962). College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, *69*, 9–15.

Gourvès, L., & Monaco, G. (2024). Nash stability in hedonic skill games. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 706–714. ACM.

Hajdjuková, J. (2006). Coalition formation games: a survey. *International Game Theory Review*, *8*(4), 613–641.

Igarashi, A., Peters, D., & Elkind, E. (2017). Group activity selection on social networks. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence (AAAI)*, pp. 565–571. AAAI Press.

Karp, R. M. (1972). Reducibility among combinatorial problems. In *Proceedings of a symposium on the Complexity of Computer Computations*, The IBM Research Symposia Series, pp. 85–103. Plenum Press.

Koutsoupias, E., & Papadimitriou, C. H. (1999). Worst-case equilibria. In *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*, Vol. 1563 of *LNCS*, pp. 404–413. Springer.

Lappas, T., Liu, K., & Terzi, E. (2009). Finding a team of experts in social networks. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pp. 467–476. ACM.

Monderer, D., & Shapley, L. S. (1996). Potential games. *Games and Economic Behavior*, *14*, 124–143.

Nisan, N., Roughgarden, T., Tardos, É., & Vazirani, V. V. (Eds.). (2007). *Algorithmic Game Theory.* Cambridge University Press.

Okimoto, T., Schwind, N., Clement, M., Ribeiro, T., Inoue, K., & Marquis, P. (2015). How to form a task-oriented robust team. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 395–403. ACM.

Rosenthal, R. W. (1973). A class of games possessing pure-strategy Nash equilibria. *International Journal of Game Theory*, *2*, 65–67.

Roth, A. E., & Sotomayor, M. A. O. (1990). *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis.* Econometric Society Monographs. Cambridge University Press.

Saad, W., Han, Z., Basar, T., Debbah, M., & Hjorungnes, A. (2011). Hedonic coalition formation for distributed task allocation among wireless agents. *IEEE Transactions on Mobile Computing*, *10*(9), 1327–1344.

Schäffer, A. A., & Yannakakis, M. (1991). Simple local search problems that are hard to solve. *SIAM Journal on Computing*, *20*(1), 56–87.

von Stengel, B. (2021). *Game Theory Basics.* Cambridge University Press, Cambridge (UK).

Woeginger, G. J. (2013). Core stability in hedonic coalition formation. In *Proceedings of the 39th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, Vol. 7741 of *LNCS*, pp. 33–50. Springer.

Wooldridge, M. J., & Dunne, P. E. (2006). On the computational complexity of coalitional resource games. *Artificial Intelligence*, *170*(10), 835–871.