

Graph-based Coalitional Games - An Analysis via Characteristics

Frank Nebel

Department of Computer Science, University of Leicester, U.K.
Frank.Nebel@mcs.le.ac.uk

ABSTRACT

In this paper we motivate a new approach to analyse the computational complexity of solution concepts and player-based properties, as well as other properties of coalitional games. This approach is based on the idea to abstract away from detailed game representations to analyse games via standard complexity proofs, towards a more abstract approach, where games are analysed by focusing on influential characteristics of related games. The core of this structure-centered perspective on coalitional games is to determine and systematically analyse promising characteristics, so that they can be used later to analyse games of a similar type. This may be particularly interesting in a well-restricted class of games, like in the context of networks. To give a first example of this approach, we concentrate on a specific type of game, namely graph-based games. In the course of our analysis we will provide various results for shortest path games, results also interesting for their own sake.

Categories and Subject Descriptors

F.1.3 [Theory of Computation]: Computation by Abstract Devices—*Complexity Measures and Classes*

General Terms

Coalitional Games, Solution Concepts, Computational Complexity

Keywords

Shortest Path Games, Graph-based Games, characteristic-based Analysis

1. INTRODUCTION

Over the last few years a series of papers [1, 2, 3, 5, 6, 12, 15, 21, 24] has been published that analyse the computational complexity of solution concepts applied to different types of coalitional games, as well as the complexity to

determine various other properties of those games. However, the coalitional games that have been analysed so far were studied in isolation from each other. For instance, results for remotely related classes of games, like graph-based games, have not been systematically compared with respect to their characteristics, computational complexity of different problems or their expressive power. If we are exclusively interested in a specific type of coalitional game, this is certainly adequate, whereas a one-by-one analysis of complexity-theoretic problems is rather impractical when we want to analyse a wider range of remotely related coalitional games. Another scenario, where it would be interesting to abstract away from specific game details and concentrate on characteristics of games, is to design or optimize coalitional games in a modular way by adding or removing characteristics, which are known to uniformly effect certain properties or complexity results for a particular type of game. This would omit the tedious process of having to repetitively reprove various game properties and complexity results for different configurations in the design process of a game.

To tackle this issue, we want to motivate a new approach, which is based on structural properties (characteristics) instead of precise game details. The basic idea is that once promising characteristics, shared by related types of coalitional games, have been determined and systematically analysed with respect to their general influence on computational complexity, expressive power, etc., they could then later be used to analyse games of a similar type. This would for example allow someone to estimate the computational complexity for certain solution concepts applied to games of this type, or to obtain an idea of game properties, like the expressive power of games or their succinctness. So, informally it can be seen as an additional layer to capture intuitions in relation with structural properties of particular games and their effect on complexity results of these games, or as a best-practice approach to generate coalitional games with particular properties.

To be clear at this point and to omit any confusion or misconception, the actual aim of such an approach is not to give standard complexity theoretic proofs, because this cannot be formally achieved by such a characteristic-based approach, but to give indications about characteristics and their computational effects on coalitional games, which fit into a certain pattern. It is therefore more a heuristic-based view, which we think is quite similar in nature to the idea of pattern-based design in programming or pattern-based learning in chess.

The rest of this paper is organised as follows. In Sec-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GAMECOMM 2011, May 16, Paris, France
Copyright © 2011 ICST 978-1-936968-09-1
DOI 10.4108/icst.valuetools.2011.245796

tion 2, we give more details regarding the new approach and motivate a test setting, a particular graph-based coalitional game. In Section 3, we present basic properties and concepts in cooperative game theory. In the next section, we introduce a basic variant of a shortest path game, which serves as a prototype to easily introduce new variants afterwards. We determine a list of characteristics to be analysed, and define several variants of shortest path games based on these. Then, in Section 5, we determine the computational complexity of various problems for different variants of shortest path games. Based on these results, we analyse in Section 6 how different characteristics of shortest path games influence the computational complexity of solution concepts or player-based properties. We conclude with Section 7.

2. COMPLEXITY VIA GAME CHARACTERISTICS

In this section, we introduce and motivate the idea to analyse the computational complexity of solution concepts and player-based properties of coalitional games via characteristics of games, and compare this with the standard approach to analyse complexity-theoretic problems in this context.

To omit confusion, the notion “characteristics” as used in this paper should not be seen as a pure technical notion. Its main purpose is to broadly refer to properties of graphs, like for instance directed acyclic graphs and trees, or particular properties shared by different graph-based games. But, as indicated later, it can also be seen in a different way, where “characteristics” are parameters as used in the context of parametrised complexity. So, expressed rather imprecise, “characteristics” are ideally properties of games or classes of games that trigger some inherent computational complexity.

The reader should be aware that this particular idea, namely to analyse games via characteristics can hardly be described as entirely novel. It has always been present, maybe not very distinctive, in the perception of the research community, but to our knowledge and surprise never resulted in a detailed account focusing on the idea and the challenges involved.

2.1 Approach and Context

We start with the standard technique to analyse coalitional games. For a given coalitional game, complexity-theoretic problems are analysed directly by applying reduction proofs. Hence, results are obtained, without taking structural similarities between related games into account. In some cases, proofs may be reused to a certain degree, but in a more extensive setting, where various games of remotely related type have to be analysed (one-by-one) or in cases where a coalitional game with special properties has to be modeled, this strategy can be quite tedious to apply.

This is the point, where complexity via game characteristics can be quite interesting to consider. The basic idea behind this perspective is that for a more general class of coalitional games, like graph-based games, specific structural characteristics of this type of game, which have to be identified first, are analysed with respect to their influence on the computational complexity of particular problems, or other properties. Hence, the key is to determine if a characteristic influences different, but related types of games, and their results in a particular (uniform) way. If there is a strong positive correlation between structural characteris-

tics and the computational complexity of specific problems, it would be possible to analyse which complexity-theoretic signature a given coalitional game (of a more general class of games) might have with respect to a given selection of complexity-theoretic problems. Of course, this would not be a proper proof, but more like an estimation of which kind of results we could expect. As an ultimate step, a standard complexity theoretic proof would still be necessary to confirm an “estimated” result.

This is why we want to emphasise at this point that we do not see this new approach as a substitute, but more like an amendment for the standard approach. It would offer a new perspective on problems and could also be useful in particular problem domains. In a more general sense, it can be seen as a characteristic-centered view on coalitional games and their properties to capture or extend structure-based intuitions. As already indicated above, we think that this new approach could especially be beneficial to model a specific type of game with requested properties or to optimise a given coalitional game.

2.2 Motivating the Approach

After positioning the approach and advertising the possible benefits, one of the key questions is, if the approach is feasible. Thus, does it make sense to look for characteristics of games or sets of characteristics that trigger some inherent computational behaviour for classes of coalitional games in a uniform way. Taking into account the heuristic nature of this approach, a concrete justification may only be given by carefully analysing various types of coalitional games and their characteristic-based variants, as well as the solution concepts and other game-based properties involved. In general, it has to be acknowledged that much more research would have to be conducted in the future to give an entirely satisfying answer. This is the crux of the problem at hand. Hence, the main motivation of this paper is to make the first step by giving a small working example on how to find characteristics for a certain type of games.

Related to the above question, we have to consider yet another key question, namely how to effectively determine promising characteristics, if they exist. For a start, we decided to apply a “brute-force” strategy to extract possible influential characteristics. Note that there may be other, possibly more efficient ways to determine and analyse promising characteristics or to extract intuitions, which will not be used here, but are nevertheless worthwhile to consider in the future. We now focus on our working example:

As a first step we decided to concentrate our research on one restricted type of coalitional game to show that this approach can be interesting and worthwhile to consider, at least for restricted types of coalitional games. Due to the fact that different coalitional games often have quite different representations and therefore different characteristics, we focused our research on a specific type of coalitional game, namely graph-based games. The reasons why we have selected this type of coalitional games are the following: Graph-based games are particularly interesting to solve network-based problems of all sorts, which often occur in operations research. They offer a manifold of graph-based characteristics that can be analysed, and several graph-based games have already been treated in the literature over the last three years [2, 5, 6, 12].

To analyse the influence of graph-based characteristics on

a working-example we have to focus on a particular type of graph-based game, which has to be analysed exhaustively by considering all variations of interesting characteristics for this type of game. At this point, we have to be systematic, because only by an extensive analysis of all the involved factors, we are able to give a meaningful interpretation of the obtained results. We finally decided to focus on shortest path games, a game which has two interesting variants [17, 29] and which also extends the corpus of complexity-theoretic results for graph-based games that have already been treated in the literature [2, 5, 8, 16, 18, 22]. More details about shortest path games, as well as various complexity-theoretic results are given in Section 4 and 5.

3. BACKGROUND

We now recall some standard concepts from cooperative game theory [10].

DEFINITION 3.1. A coalitional game with transferable utility (TU-game) is a pair $\langle N, v \rangle$, where

- N is the set of players and
- $v : 2^N \rightarrow \mathbb{R}_0^+$ is a function that maps each group of players $S \subseteq N$ to a positive real-valued payoff.

DEFINITION 3.2. A game v is monotonic if for all $S, T \subseteq N$, $S \subseteq T$ implies $v(S) \leq v(T)$.

DEFINITION 3.3. A coalitional game $\langle N, v \rangle$ is called a simple game¹ if $v(S) \in \{0, 1\}$ for all $S \subseteq N$, $v(\emptyset) = 0$ and $v(N) = 1$.

DEFINITION 3.4. A coalition $S \subseteq N$ in a simple game $\langle N, v \rangle$ is called a minimal winning coalition if $v(S) = 1$ and for every $T \subset S$, $v(T) = 0$. We denote by $M(v)$ the set of minimal winning coalitions of v and by $M_i(v)$ the subset of $M(v)$ formed by coalitions $S \subseteq N$ such that $i \in S$

DEFINITION 3.5. A player $i \in N$ in a simple game² $\langle N, v \rangle$ is said to be:

- a null player if $v(S) = v(S \cup \{i\})$ for all $S \subset N \setminus \{i\}$
- a dictator if $v(S) = 1$ iff $i \in S$
- a veto player if $v(S) = 1$ implies $i \in S$.

We now present various solution concepts. We start with an important stability concepts, namely the *core* and then continue with power indices. But before we can give the actual definition of the core, we first have to introduce what is an allocation of a game: An *allocation* is a function x from N to \mathbb{R} . We denote its i -th component by $x(i)$ or x_i and for a coalition $S \subseteq N$, we abbreviate $\sum_{i \in S} x_i$ by $x(S)$. An essential type of allocation is an *imputation*, which is an allocation that is *efficient* ($x(N) = v(N)$) and *individual rational* (for all $i \in N$, $x_i \geq v(i)$). If we now add *group rationality* (for all $S \subseteq N$, $x(S) \geq v(S)$) we get the concept of the core.

¹In the literature simple games are sometimes defined as being monotonic as well.

²The definition of null and veto players can be generalised for coalitional games.

DEFINITION 3.6. Let $\langle N, v \rangle$ be a cooperative game. Then the core of v is defined as: $Core(v) := \{x \in \mathbb{R}^n \mid x \text{ is an imputation satisfying group rationality}\}$

Now, we present various power indices, which were originally proposed in the context of voting, but can also be used for monotonic games in general (depending on the power index). In principle, it can be said that a power index allows a more detailed view on power distributions in voting situations (or monotonic games) and give insights that are not obvious at first glance.

DEFINITION 3.7. Given a monotonic game $\langle N, v \rangle$, the Shapley-Shubik index [25] of player i is given by

$$\varphi_i(N, v) = \frac{\sum_{S \subseteq N \setminus \{i\}} |S|!(|N| - |S| - 1)! [v(S \cup \{i\}) - v(S)]}{|N|!}$$

DEFINITION 3.8. Given a monotonic game $\langle N, v \rangle$, the Banzhaf index [7] of player i is given by

$$\beta_i(N, v) = \frac{\sum_{S \subseteq N \setminus \{i\}} [v(S \cup \{i\}) - v(S)]}{2^{|N|-1}}$$

DEFINITION 3.9. The Deegan-Packel power index [11] of a player i in the simple game $\langle N, v \rangle$ is give by:

$$p_i(N, v) = \frac{1}{|M(v)|} \sum_{S \in M_i(v)} \frac{1}{|S|}$$

DEFINITION 3.10. Given a simple game $\langle N, v \rangle$, the Public Good index (PGI) [19] assigns to each player $i \in N$ the real number:

$$\delta_i(N, v) = \frac{|M_i(v)|}{\sum_{j \in N} |M_j(v)|}$$

4. SHORTEST PATH GAMES

The class of shortest path games is a type of coalitional game that has not yet been considered, to our knowledge, in the context of computational complexity. There are two different definitions of how to model shortest path problems in a game-theoretic context, which were introduced by Fragnelli, Garcia-Jurado, and Mendez-Naya [17] and Voorneveld and Grahn [29]. Based on these definitions, we present a basic variant of a shortest path game, which has very restricted structural properties. This shall give the reader the flavour of shortest path games and allows us to compare this type of game with other graph-based games later.

4.1 Value Shortest Path Games (VSPG)

The shortest path games that we consider here are limited to a finite set of players, where each player owns arcs or vertices in a finite network. There are costs associated to the use of each arc or vertex and there are rewards involved with the transport of an item from the source to the sink of the network. Then, if a coalition transports goods from the source to the sink of a graph using a path the coalition owns, it will receive a specified reward and also incur costs, namely the costs of the shortest path owned by the coalition. If the difference between the reward and the cost is positive, the coalition makes a profit. Otherwise, the coalition can generate profit zero by simply doing nothing.

Now we present a basic class of shortest path games, called VSPG, in a more formal way. This variant of a shortest path

game borrows its basic outline from the original definitions of shortest path games by Fragnelli et al. [17] and Voorneveld and Grahn [29], where we restricted some properties of the model to have a more basic version.

DEFINITION 4.1. *A shortest path pre-problem Σ is a tuple $\langle V, A, So, Si \rangle$, where $\langle V, A \rangle$ is a directed graph with a source (So) and a sink (Si). $A \subseteq V \times V$ is the set of directed arcs in the network.*

Let Σ be a shortest path pre-problem, where the arcs of the graph $\langle V, A \rangle$ are owned by a finite set of players N according to a total bijective map $o : A \rightarrow N$. Furthermore, we also have a cost map c that assigns to every arc $a \in A$ a non-negative real number.

Given the simple structure of *VSPG*, a path owned by players of coalition S is a sequence (v_1, v_2, \dots, v_m) of vertices such that $v_1 = So$, $v_m = Si$ and for each $k \in \{1, \dots, m-1\}$ the arc (v_k, v_{k+1}) is owned by player $i_k \in S$. Let $P(S)$ denote the collection of all paths owned by coalition S . For any path P we denote the set of corresponding owners by $o(P)$.

Suppose that the transportation of a certain good from the source to the sink of Σ produces an income r and a cost given by the length of the path that was used. In particular, the costs associated to a path $p = (v_1, v_2, \dots, v_m) \in P(S)$ are defined as the sum of the costs of its arcs:

$$cost(p) = \sum_{k=1}^{m-1} c(v_k, v_{k+1})$$

Note that a coalition $S \subseteq N$ can transport the good only through paths owned by it and a path P is owned by a coalition S if $o(P) \subseteq S$.

If a coalition S has to find a path from source to sink, it will choose among its alternatives in $P(S)$ the path with minimal costs. Define for each $S \in 2^N \setminus \{\emptyset\}$:

$$c(S) = \begin{cases} \min_{p \in P(S)} cost(p) & \text{if } P(S) \neq \emptyset \\ \infty & \text{otherwise} \end{cases}$$

We overloaded c at this point, but it will be obvious from the context which function is meant. The shortest path can easily be computed in polynomial time using Dijkstra's [13] well-known algorithm. We now introduce the notion of a *shortest path game environment* σ , which is any such tuple $\langle N, \Sigma, o, c, r \rangle$.

DEFINITION 4.2. *Let σ be a shortest path game environment, then the shortest path game $\langle N, v_\sigma \rangle$ of type *VSPG* is given by: For every $S \subseteq N$*

$$v_\sigma(S) = \begin{cases} r - c(S) & \text{if } S \text{ owns a path in } \Sigma \text{ and } c(S) < r \\ 0 & \text{otherwise} \end{cases}$$

Note that the game *VSPG* defined above has a global reward scheme. This means that a successful coalition S receives a "global" reward r , whereas an individual reward scheme is specified as $r = \sum_{i \in S} r_i$, where r_i is an individual reward assigned to each player of the game.

4.2 Characteristics and Variants of Shortest Path Games

In the following, we propose various characteristics for graph-based coalitional games in general, as well as more specific characteristics for shortest path games. For each

characteristic we have two options, which might influence the expressive power of a graph-based game and also may have an effect on the computational complexity of solutions concepts or graph-based properties.

1. *Players of graph-based games are attached to the graph's arcs or vertices.* [OWNARC, OWNVERTEX]
2. *An arc (vertex) in a graph-based game can be owned by one player, or it can be owned by at least one player.* [OWNED1, OWNED*]
3. *A player in a graph-based game can own one arc (vertex), or at least one arc (vertex).* [OWN, OWN*]
4. *The underlying graph of a graph-based game is directed or acyclic directed.*
5. *A graph-based game is based on a directed graph or tree.*
6. *The source and sink of a shortest path or flow game are simple vertices or sets of vertices.* [SOSI-VERTEX, SOSI-SET]
7. *A shortest path game can have a global reward scheme or an individual reward scheme.* [GREW, IREW]

For our analysis we only considered characteristics (2), (3), (4), (5) and (7) in a systematic way. A more detailed analysis and discussion about why the following characteristics are more interesting than others can be found elsewhere [23]. The characteristics (1) to (5) are applicable to all kind of graph-based games, whereas characteristic (6) and (7) have been specifically selected for shortest path games.

For characteristic (4) and (5), we use the label "DAG" (TREE) to restrict the underlying graph of a game to an acyclic directed graphs (tree). We furthermore use the notation $M(\text{DAG})$ ($M(\text{TREE})$) to indicate that a class of shortest path game, let's say M , is considered for inputs restricted to acyclic directed graphs (trees). For all the other characteristics we define a label (added in parenthesis), which can be used to characterize the variants of shortest path games that we are going to present. There is another important characteristic, which will not be analysed in the same way, but is nevertheless important and well used: *A shortest path game can return its value or be converted into a simple game by introducing a threshold* [VAL,THOLD]. These games are then called threshold games.

In Table 1 we summarize the variants of shortest path games that will be analysed in this paper. Note that SPG-F is the shortest path game introduced by Fragnelli et al. [17] and SPG-VG the game introduced by Voorneveld and Grahn [29].

A detailed analysis of the expressive power of the games involved and how the variants of shortest path games relate to each other, as well as different variants of various graph-based games can be found elsewhere [23]. For our account of computational complexity for shortest path games it is sufficient to know that all the variants of shortest path games are monotonic, threshold games are simple and that all non-threshold games with individual reward scheme have a non empty core.

Variant	Characteristics
<i>VSPG</i>	OWNARC, OWNED1, OWN1, SoSi-VERTEX, GREW, VAL
<i>VSPG*</i>	OWNARC, OWNED1, OWN*, SoSi-VERTEX, GREW, VAL
<i>VSPG*+</i>	OWNARC, OWNED*, OWN*, SoSi-VERTEX, GREW, VAL
<i>TSPG</i>	OWNARC, OWNED1, OWN1, SoSi-VERTEX, GREW, THOLD
<i>SPG-F</i>	OWNVERTEX, OWNED1, OWN*, DAG, SoSi-SET, GREW, VAL
<i>XSPG</i>	OWNARC, OWNED1, OWN1, SoSi-VERTEX, IREW, VAL
<i>XSPG*</i>	OWNARC, OWNED1, OWN*, SoSi-VERTEX, IREW, VAL
<i>XSPG*+</i>	OWNARC, OWNED*, OWN*, SoSi-VERTEX, IREW, VAL
<i>TXSPG</i>	OWNARC, OWNED1, OWN1, SoSi-VERTEX, IREW, THOLD
<i>SPG-VG</i>	OWNARC, OWNED*, OWN*, DAG, SoSi-VERTEX, IREW, VAL

Table 1: Variants of Shortest Path Games

5. COMPUTATIONAL COMPLEXITY

In this section, we present various complexity-theoretic results for player-based properties and solution concepts, which are applied on variants of shortest path games. All proofs, as well as some immediate results for threshold variants of shortest path games (monotonic simple games) or complexity results for the core can be found in full detail elsewhere [23]. For later reference and to compare and interpret the results in the next section, an overview of all the results for shortest path games is given in Table 2.

Due to the multitude of different complexity problems involving various variants of coalitional games and solution concepts (player-based properties), we present the complexity problems in a parametrised form. This allows us a more compact representation of the results. For example, we introduce the X-NULL-PLAYER decision problem as the general null player problem, where X indicates a particular type of game. Thus, if we want to consider the null player problem with respect to shortest path games of type *VSPG*, we refer to this particular problem as VSPG-NULL-PLAYER.

We generally assume that the reader is familiar with standard complexity theory and omit any further explanation apart from the slightly more uncommon complexity class, $\#\mathcal{P}$ (pronounced “number P” or “sharp P”), which is the set of counting problems associated with the decision problems in \mathcal{NP} . This complexity class has been introduced by Valiant [28] and will be used for various results in this section. Expressed more formally:

DEFINITION 5.1. $\#\mathcal{P}$ is the complexity class of all the functions $f : \Sigma^* \rightarrow \mathbb{N}$ such that there exists a non-deterministic polynomial time Turing machine M and for all inputs $x \in \Sigma^*$, $f(x)$ is the number of accepting paths of M .

This particular complexity class is used in the context of various coalitional games, because it often allows a more precise classification of problems involving power indices.

5.1 Player-based Properties

DEFINITION 5.2. X-NULL-PLAYER: Given a game v of type X and a player a_i , test whether a_i is a null-player in v .

LEMMA 5.3. VSPG-NULL-PLAYER and XSPG-NULL-PLAYER are co- \mathcal{NP} -complete

Lemma 5.3 is based on the fact that *VSPG* expresses all directed connectivity games (see [23]) and a complexity result from Bachrach and Rosenschein [5], which is about

null-players³ in directed connectivity games. The same result follows immediately for *VSPG**, *VSPG*+*, *TSPG* and *TXSPG*. Due to the fact that *XSPG* (*XSPG** and *XSPG*+*) does not directly express the class of connectivity games, a standard reduction proof with respect to directed connectivity games has to be applied.

Similar to the case of network flow games [5], we have the following: if the input is restricted to cases where the graph is a directed acyclic graph (DAG), we can prove that the problem is tractable.

LEMMA 5.4. VSPG(DAG)-NULL-PLAYER, VSPG*(DAG)-NULL-PLAYER, VSPG*+(DAG)-NULL-PLAYER, SPG-F-NULL-PLAYER and TSPG(DAG)-NULL-PLAYER are in \mathcal{P} .

We will concentrate on VSPG(DAG). For the other variants, the proof can easily be adapted.

PROOF. Let v_σ be an N -player VSPG in a shortest path cooperative situation σ . Furthermore, let a_i be a player and $e = (u, v)$ an arc such that $o(e) = a_i$. We assume w.l.o.g that $u \neq So$ and $v \neq Si$. We determine the shortest path P_1 from So to u and the shortest path P_2 from v to Si . This can be easily done using Dijkstra’s algorithm. We now have two main cases to consider:

Case 1: If both paths exist, we can define coalition $C := o(P_1) \cup o(P_2) \cup o(\{(u, v)\})$, which has $R := P_1 \circ e \circ P_2$ as the shortest path through e .

Claim 1: P_1 and P_2 are vertex-disjoint, i.e. they do not share vertices except So and Si .

Proof (Claim 1): Let’s assume for the sake of a contradiction that this is not the case. So, there is a vertex w that is contained in both paths. Given that w is a vertex in P_1 and P_2 , there must be a path from w to u and v to w respectively. Hence, there is a cycle $u \rightarrow v \rightarrow w \rightarrow u$ and therefore the assumption that the graph is an acyclic directed graph (DAG) is violated.

Claim 2: There is no path $T \in P(C)$ such that $cost(T) < cost(R)$. Furthermore it is the case that $c(C \setminus \{a_i\}) = \infty$.

Proof (Claim 2): By definition of path R , it is clearly the case that there is no path $T \in P(C)$ such that $e \in T$ and $cost(T) < cost(R)$. Hence, if there should be some $T \in P(C)$ such that $cost(T) < cost(R)$, then it must be the case that $e \notin T$. So, let’s assume that $e \notin T$. We now have to check if there is another path from So to Si in C . But given that P_1 and P_2 are vertex-disjoint (Claim 1), there

³Bachrach and Rosenschein refer to them as dummy-players in [5].

does not even exist a path $T \in P(C \setminus \{a_i\})$ from So to Si . Hence, $\infty = \text{cost}(T) > \text{cost}(R)$ and $c(C \setminus \{a_i\}) = \infty$.

After having proved the two claims, we have to consider two sub-cases:

Case 1.1: ($c(C) \geq r$). In this case all possible paths P through arc e automatically have the property $c(P) \geq r$. Hence, for all these paths P with the corresponding coalition $D = o(P) \subseteq N$ we have $v_\sigma(D) = 0$ and after removing arc e , there is no path anymore and therefore $v_\sigma(D \setminus \{a_i\}) = 0$ as well. But these paths are the only possibility for player a_i to influence the value of the game. So, without participation in a profitable path, a_i must be a null player.

Case 1.2: ($c(C) < r$). So, $v_\sigma(C) = r - c(C) > 0$ and by claim 2 we get $v_\sigma(C \setminus \{a_i\}) = r - c(C \setminus \{a_i\}) = 0$. Hence, a_i is not a null player.

Case 2: If P_1 or P_2 does not exist, there is no path from So to Si that includes arc e . Hence, by including a_i to a coalition the value of v_σ will not change. The same holds, when we remove the player. Hence, a_i is a null player.

To sum it up: The shortest paths P_1 and P_2 can be computed in polynomial time. Then it has to be checked if both paths exist (Condition 1) and if this is the case yet another condition has to be checked, namely if $c(C) \geq r$ (Condition 2). By Claim 2 we even know that $c(C) = \text{cost}(R)$ and therefore it is enough to check if $\text{cost}(R) = \text{cost}(P_1) + \text{cost}(e) + \text{cost}(P_2) \geq r$. All this can be done in polynomial time. Hence, given that we covered all cases, it can be tested in polynomial time if a_i is a null-player in v_σ . \square

The same result holds as well for those variants of shortest path games with an individual reward scheme.

LEMMA 5.5. XSPG(DAG)-NULL-PLAYER, XSPG*(DAG)-NULL-PLAYER, XSPG*+(DAG)-NULL-PLAYER, SPG-VG-NULL-PLAYER and TXSPG-NULL-PLAYER is in \mathcal{P} .

PROOF. Let v_σ be an N -player XSPG in a shortest path cooperative situation σ and let a_i be an arbitrary player. We now have to consider two cases regarding the individual reward assigned to player a_i :

Case 1: ($r_i = 0$) Under this assumption player a_i can only increase the value of a coalition he joins by offering cost improvements with respect to path routing, thus offering a shorter path. The proof is similar to the proof of VSPG: We first generate coalition C and given that $r_i = 0$, we have $r(C) = r(C \setminus \{a_i\})$. Hence, we have a fixed reward with respect to the coalitions we are interested in and can therefore continue as in the proof of VSPG.

Case 2: ($r_i > 0$) We first check if there is a coalition $S \subseteq N$ such that $v_\sigma(S) > 0$. This can be easily done by determining the shortest path for the grand coalition N , calculating the cost of the shortest path and subtracting it from $r(N)$. If $v_\sigma(N) = 0$, then we can deduce from the fact that v_σ is monotonic that $v_\sigma(S) = 0$ for all $S \subseteq N$, and therefore a_i is clearly a null-player. If this is not the case, then we can reason as follows: For any coalition $S \subseteq N$ ($v_\sigma(S) > 0$) containing a_i , if we remove a_i from the coalition, the coalition's value decreases due to the fact that r_i is subtracted and furthermore given that the routing can clearly not be improved with one arc removed, we have $v_\sigma(S \setminus \{a_i\}) < v_\sigma(S)$. Now, if we assume that there is at least one effective path, we know that $v_\sigma(N) > 0$ must hold for the grand coalition N and therefore by the deduction step above, a_i is not a null-player. \square

DEFINITION 5.6. X-VETO-PLAYER: Given a game v of type X and a player a_i , test whether a_i is a veto player in v .

PROPOSITION 5.7. VSPG-VETO-PLAYER is in \mathcal{P} .

The same follows for VSPG*, VSPG*+ and also for XSPG, XSPG*, XSPG*+ and SPG-VG. Due to the fact that VSPG*(DAG) and SPG-F are similar (only differ for tags: SoSi-Set/SoSi-Vertex and OwnVertex/OwnArc) and by taking into account that these characteristics do not influence the computational complexity in this case, the following result holds.

PROPOSITION 5.8. SPG-F-NUL-PLAYER and SPG-F-VETO-PLAYER is in \mathcal{P} .

5.2 Power Indices

After determining the computational complexity of VSPG-NUL-PLAYER and XSPG-NUL-PLAYER, as well as for various other variants, we can apply a well-known fact to determine the computational complexity of power indices: To compute a power index (Shapley-Shubik, Banzhaf, Deegan-Packel and Public-Good index) for a player is at least as difficult as testing for a null-player.

DEFINITION 5.9. X-Y-INDEX: Given a game v of type X and a player i , compute the Y power index of player i in game v .

LEMMA 5.10. To compute VSPG-SHAPLEY-SHUBIK-INDEX (XSPG-SHAPLEY-SHUBIK-INDEX), VSPG-BANZHAF-INDEX (XSPG-BANZHAF-INDEX), VSPG-DEEGAN-PACKEL-INDEX (XSPG-DEEGAN-PACKEL-INDEX) and VSPG-PUBLIC-GOOD-INDEX (VSPG-PUBLIC-GOOD-INDEX) is intractable.

The same result follows for VSPG*, VSPG*+ and TSPG (XSPG*, XSPG*+ and TXSPG). We can even prove a stronger result:

THEOREM 5.11. VSPG-BANZHAF-INDEX and XSPG-BANZHAF-INDEX is $\#\mathcal{P}$ -complete.

To prove this result⁴, we reduce the S-T-CONNECTEDNESS problem, which was introduced by Valiant [28] as one of the first problems known to be $\#\mathcal{P}$ -complete, to our problem. The detailed definition of the problem is:

Instance: $G = \langle V, A \rangle$; $s, t \in V$
Question: Number of subgraphs of G in which there is a (directed) path from s to t

PROOF. We start with $\#\mathcal{P}$ -hardness: Let's assume that we have an instance of S-T-CONNECTEDNESS, hence a graph $G = \langle V, A \rangle$ and two distinct vertices $s, t \in V$. We first add another vertex Si to the graph and another arc $a' = (t, Si)$. So, we have $G' = \langle V', A' \rangle$, where $V' = V \cup \{Si\}$ and $A' = A \cup \{a'\}$. Let $So = s$. Then we define the set of players $N = \{1, 2, \dots\}$, where $|N| = |A'|$ and we assign to every player exactly one arc. We refer to the player owning arc a' as i' . Furthermore, let $\pi : A \rightarrow N \setminus \{i'\}$ be an arbitrary bijective mapping and for any $a \in A'$, we define the ownership mapping o as follows:

$$o(a) = \begin{cases} i' & \text{if } a = a' \\ \pi(a) & \text{otherwise} \end{cases}$$

⁴Note that the proof is for VSPG, but can easily be replayed for XSPG.

We define a cost function that assigns cost 0 to every arc of G' and set $r := 1$. This gives us a shortest path cooperative situation σ with the corresponding game v_σ ($VSPG$). Note that this transformation takes only polynomial time. For convenience we introduce the following notation $\bar{\beta}_i = \{S \subseteq N \setminus \{i\} \mid v_\sigma(S \cup \{i\}) - v_\sigma(S) = 1\}$, which is a subterm used in the definition of the Banzhaf index. We now show that there is a direct correspondence of the following form:

Claim: $X \in \bar{\beta}_{i'}(N, v_\sigma)$ iff subgraph H of G induced by $\sigma^{-1}(X)$ ⁵ is such that there exists a path P in H .

(\Rightarrow): Let $X \in \bar{\beta}_{i'}(N, v_\sigma)$, i.e. by definition $X \subseteq N \setminus \{i'\}$, $v_\sigma(X \cup \{i'\}) = 1$ and $v_\sigma(X) = 0$. We can now deduce from $v_\sigma(X \cup \{i'\}) = 1$ and the definition of $VSPG$ that for v there is a path from $s = So$ to Si , and therefore also a path from s to t in the subgraph induced by $\sigma^{-1}(X)$.

(\Leftarrow): We assume that H is a subgraph of G , where $S \subseteq A$ is the set of arcs of H , such that there exists a path in H . Given that i' is a veto-player of v_σ and $i' \notin \sigma^{-1}(S)$ we have $v_\sigma(S) = 0$. Furthermore, given that there is a path in S (from $s = So$ to t), we get a path So to Si by adding a' to the coalition S . Thus $v_\sigma(S \cup \{i'\}) = 1$, and therefore $S \in \bar{\beta}_{i'}(N, v_\sigma)$.

So, we have a bijection between the two sets, and therefore the cardinality of both sets is equal. Hence, if we were able to compute the Banzhaf index $\beta_{i'}(N, v_\sigma)$ for player $i' \in N$ in polynomial time, we could multiply it with 2^{N-1} to get $\#\bar{\beta}_{i'}(N, v_\sigma)$. This would give us the number of solutions for the S-T-CONNECTEDNESS instance, and therefore would lead to a contradiction, because S-T-CONNECTEDNESS is $\#\mathcal{P}$ -complete. We now continue to prove $\#\mathcal{P}$ -membership:

The Banzhaf index of player $i \in N$ of a shortest path game (N, v_σ) of type $VSPG$ is $\beta_i(N, v_\sigma)$. It is the proportion of all winning coalitions where i is critical, out of all winning coalitions that contain i . Let $S \subseteq N$ be any coalition, it can be checked in polynomial time whether $i \in S$, $v_\sigma(S) = 1$ and $v_\sigma(S \setminus \{i\}) = 0$. The last two conditions are polynomial by definition of shortest path games (based on Dijkstra's algorithm). Hence, it can easily be checked if $i \in S$ and if player i is critical for a coalition S ($v_\sigma(S) - v_\sigma(S \setminus \{i\}) = 1$).

Due to the fact that we can construct a deterministic polynomial Turing machine M that tests if a player is critical in a coalition, as shown above, we can now construct a non-deterministic Turing machine M' that first chooses a coalition, under the conditions that i is in the coalition, non-deterministically and then tests if i is critical for that coalition. The number of accepting paths of M' is the number of coalitions that contain i where i is critical. Let $|N| = n$. As introduced above we denote the number of such accepting paths of M' as $\#\bar{\beta}_i(N, v_\sigma)$. Then the Banzhaf power index of agent i is $\beta_i(N, v_\sigma) = \frac{\#\bar{\beta}_i(N, v_\sigma)}{2^{n-1}}$.

Calculating the numerator $\#\bar{\beta}_i(N, v_\sigma)$ is a $\#\mathcal{P}$ problem. Since the denominator is constant (given a domain with n players), $VSPG$ -BANZHAF-INDEX is in $\#\mathcal{P}$. So, we have shown that $VSPG$ -BANZHAF-INDEX is $\#\mathcal{P}$ -complete. \square

The same result follows for $VSPG^*$, $VSPG^*+$ and $SPG-F$ ($XSPG^*$, $XSPG^*+$ and $SPG-VG$) immediately. This also holds for $TSPG$ ($TXSPG$) when we set the threshold to 1. Based on the result for $TSPG$ -BANZHAF-INDEX ($TXSPG$ -BANZHAF-INDEX) and a proposition by Aziz et al. [2] (Proposition 4.3 and 4.4) we can deduce:

⁵Normally we speak of a graph being induced by a set of vertices, but we can also use arcs instead.

COROLLARY 5.12. $TSPG$ -SHAPLEY-SHUBIK-INDEX and $TXSPG$ -SHAPLEY-SHUBIK-INDEX is $\#\mathcal{P}$ -complete.

5.3 Monotonic Simple Games

Various complexity results follow directly from the fact that a game is monotonic or simple. These complexity results are rather straightforward to prove. Henceforth, we only sketch or even omit some of the proofs.

PROPOSITION 5.13. Let $\langle N, v \rangle$ be a monotonic simple game such that for all coalitions $S \subseteq N$, $v(S)$ can be computed in polynomial time and let $a_i \in N$. We can determine in polynomial time if a_i is a veto-player (dictator) in v .

Using the generalised definition of a veto-player for coalitional games, which we indicated above, the proposition can be generalised to hold for monotonic coalitional games as well. We now present some complexity problems with respect to the core of simple monotonic games.

DEFINITION 5.14. X-EMPTYCORE: Given a game v of type X , test whether $Core(v) = \emptyset$

PROPOSITION 5.15. For a monotonic simple game v the decision problem EMPTY-CORE is in \mathcal{P} .

PROOF. Due to the fact that a simple game v has a non-empty core iff v has veto players ([10], Theorem 1.10.6), we can immediately deduce that the result must hold, since by Proposition 5.13 we can check for every monotonic simple game in polynomial time if a player is a veto-player. Hence, we can determine in polynomial time if there is at least one veto-player in N by testing all players. \square

We can even do more, namely compute the elements of the core in polynomial time for simple monotonic games.

DEFINITION 5.16. X-ELEMENTSCORE: Given a game v of type X , return a set X such that $X \subseteq Core(v)$

To prove this result we use the following folk theorem: If there are no veto players in a simple game v , then the core is empty. Otherwise, let a_1, \dots, a_m be the veto players in v . Then the core is the set of imputations that distribute the gains over all veto players, and only the veto players: $Core(v) = \{(p_1, \dots, p_m) \mid \sum_{i=1}^m p_i = 1\}$. Details and a proof for this theorem can be found in a paper by Bachrach et al.[4]. Hence, to compute the core we just have to compute all veto players. But this can easily be done, because to check for a veto player is in \mathcal{P} .

DEFINITION 5.17. X-COREMEMBERSHIP: Given a game v of type X and an allocation x , check whether $x \in Core(v)$

PROPOSITION 5.18. Let v be a monotonic simple game and x an imputation. The decision problem COREMEMBERSHIP is in \mathcal{P} .

To obtain this result, we just have to compute all veto players and then check if the imputation x fits into the pattern described by the folks theorem above.

Finally, given that $TSPG$ and $TXSPG$ are monotonic simple games and the value for an arbitrary coalition can be computed in polynomial time (Dijkstra), the results can be immediately applied.

5.4 Minimal Winning Coalitions

Minimal winning coalitions are an important kind of coalition. Expressed informally, they can be seen as limit cases that have maximal power, but minimal effort. Due to their significance, we were interested in determining the computational complexity to count the number of minimal winning coalitions in threshold shortest path game.

We already introduced a formal notion of a minimal winning coalition (see Definition 3.4). We now want to reinterpret minimal winning coalitions in the context of *TSPG*. This is easily done, because minimal winning coalitions are exactly all the (simple) paths P (having non repeating vertices) from S_o to S_i such that $r - \text{cost}(P) \geq T$, thus all the profitable paths.

DEFINITION 5.19. *X-#MWC: Given a game v of type X , return the number of minimal winning coalitions of game v .*

THEOREM 5.20. *TSPG-#MWC and TXSPG-#MWC is #P-complete.*

To prove this result, the S-T-PATHS⁶ problem, which was introduced by Valiant [28], is used. The detailed definition of the problem is:

Instance: $G = \langle V, A \rangle; s, t \in V$
Question: Number of paths from s to t that visit every vertex at most once.

PROOF. We first prove #P-hardness: We have an instance of S-T-PATHS, hence a graph $G = \langle V, A \rangle$ and two distinct vertices $s, t \in V$. We now define the set of players $N = \{1, 2, \dots\}$, where $|N| = |A|$ and we assign to every player exactly one arc. We also define a cost function that assigns cost 0 to every arc of G and set $r := 1$ and $T := 1$. So, we have a shortest path game environment σ and the corresponding game v_σ of type *TSPG*. Note that this transformation takes only polynomial time. Now we have to show that there is a direct correspondence between the concept of a minimal winning coalition (MWC) and an s-t-path.

Claim: X is a MWC iff X is a s-t-path

(\Rightarrow): Let X be a MWC. Then, as stated previously, X must be a profitable (simple) path from s to t . Hence, X is an s-t-path.

(\Leftarrow): Let P be an s-t-path. So, we have a (simple) path from s to t . Given that we assigned a cost of 0 to all arcs, $\text{cost}(P) = 0$ and therefore $r - c(P) = 1 - 0 = 1 \geq T = 1$. Hence, $v_\sigma(P) = 1$. Given that P is a path, we have $v_\sigma(P') = 0$ for every $P' \subset P$, because $r - c(P') = 0 < T = 1$. Thus, by definition P is a minimal winning coalition.

Now we prove #P-membership: Let $S \subseteq N$ be any coalition, it can be checked in polynomial time if S is a MWC:

Step 1: We first have to check if there is a shortest path P from S_o to S_i involving all players in S . This can be done by applying Dijkstra's algorithm to determine the shortest path P in S . Then we have to check if $o(P) = S$. Note that both steps can be done in polynomial time.

Step 2: If this is not the case, S cannot be a MWC. Otherwise we check if the corresponding path P is profitable, which can be done in polynomial time as well.

⁶The decision problem heavily relies on the notion of a simple path (also referred to as self-avoiding walks). For many similar decision problems, leaving this property out, polynomial time algorithms have been found. The decision problem applies to both, directed and undirected graphs.

Step 3: If path P is not profitable, then S cannot be a MWC. Otherwise it must be a MWC, because $v_\sigma(S) = 1$ and given that it is a path, the reduction of the coalition by any player $T \subset S$ will lead to an interrupted path, and therefore infinite costs. Hence, $v_\sigma(T) = 0$.

Due to the fact that we can construct a deterministic polynomial Turing machine M that tests if a coalition $S \subseteq N$ is a MWC, as shown above, we can now construct a non-deterministic Turing machine M' that first chooses a coalition S non-deterministically and then tests if S is a MWC. The number of accepting paths of M' is then the number of MWC. Now, according to the Definition 5.1, TSPG-#MWC is in #P. So, finally we have that TSPG-#MWC is #P-complete. \square

6. INTERPRETATION AND DISCUSSION

In the previous section we obtained various results, which are summarised in Table 2. The notions and symbols that we use have the following interpretation:

- Complexity results separated by “|” indicate that we have two results for different kinds of complexity problems (e.g. decision, function or counting) or a result for a restricted class of problems (e.g. DAG or Tree).
- If a field in the table contains “-”, then the property or solution concept either cannot formally be applied for this game or a previous results already indicates that a solution may or may not exist.
- An open problem is indicated by “?”.
- Results for games with global and individual rewards often coincide. If this is not the case, then we indicate the difference by “ $x[y]$ ”, where x is the result for the global and y for the individual reward scheme.

The obtained results are quite interesting on their own and more interestingly they also show a remarkable resemblance with respect to complexity results for related kinds of games, like network flow games [20], vertex connectivity games [6], minimum cost spanning tree games [9] and spanning connectivity games [2], as will be commented on later.

As mentioned in Section 2, our main interest is to analyse the influence of different characteristics on the computational complexity of player-based properties and solution concepts applied to shortest path games.

As can be seen immediately, the results for games with individual and global reward scheme, are nearly identical and only differ for ELEMENTCORE. Hence, characteristic (7) has no influence in our test setting. This is a surprising result, because we initially expected it to be a strong candidate to trigger computational behaviour. But apart from (7), also other characteristics that we considered, i.e. (2) and (3), have no influence on the computational complexity of power indices or player-based properties. These results seem rather disappointing regarding the initial expectations, but there is more to observe:

- The main reason, why those characteristics have no influence for shortest path games is that for basic variants of shortest path games (e.g. *VSPG* or *XSPG*) the problems are already intractable. In a small case study (see [23]), we confirmed that various graph-based

Global Reward Scheme (GRew)	VSPG/VSPG*/VSPG*+	TSPG	SPG-F
Shapley-Shubik-Index	\mathcal{NP} -hard	$\#\mathcal{P}$ -complete	?
Banzhaf-Index	$\#\mathcal{P}$ -complete	$\#\mathcal{P}$ -complete	$\#\mathcal{P}$ -complete
Deegan-Packel-Index	-	\mathcal{NP} -hard	-
Public-Good Index	-	\mathcal{NP} -hard	-
Null-Player	$co\text{-}\mathcal{NP}$ -complete \mathcal{P} (DAG)	$co\text{-}\mathcal{NP}$ -complete \mathcal{P} (DAG)	\mathcal{P}
Veto-Player	\mathcal{P}	\mathcal{P}	\mathcal{P}
Dictator	-	\mathcal{P}	-
Minimal Winning Coalition	-	$\#\mathcal{P}$ -complete	-
EmptyCore	? [-]	\mathcal{P}	? [-]
CoreMember	?	\mathcal{P}	?
ElementCore	? [\mathcal{P}]	\mathcal{P}	? [\mathcal{P}]
Individual Reward Scheme (IRew)	XSPG/XSPG*/XSPG*+	TXSPG	SPG-VG

Table 2: Complexity-theoretic results for different variants of shortest path games.

games, which share basic characteristics, have rather similar (mostly intractable) complexity results. For our case study we have taken the following coalitional games (with corresponding results) into account: network flow games [2, 5], vertex connectivity games [6], minimum cost spanning tree games [8, 22] and spanning connectivity games [2]. So, this case study indicates that characteristics, like (2) and (3), as well as (7) for shortest path games, are in a sense not strong enough to influence the computational complexity for those problems that have been considered.

- On the other hand, those characteristics have quite some influence with respect to the expressive power of those games. More details about expressive power and the comparison between different coalitional games, like flow games, market games, linear production games can be found elsewhere ([23], Chapter 4).
- From our analysis of shortest path games and the case study involving other graph-based games, we have an indication that characteristics (4) and (5), namely DAG and TREE, have quite some influence with respect to the computational complexity of power indices and the determination of player-based properties. Some examples for various types of graph-based games can be found in [2, 5, 6, 8, 22].

To sum it up, the results for shortest path games only supported the advocated characteristic-based perspective to a limited extent. The working example, which was introduced with the intention to emphasize the key challenge of the approach, namely the extraction of influential game characteristics, did not live up to the initial expectations. Nevertheless, by taking other graph-based coalitional games into account, we were able to present some interesting observations about characteristics and their influence, which can be useful for further work in this context. Various observations regarding graph-based games seem to suggest that there is a general tendency for problems in this context to be intractable. This indicates that we have to focus on more influential characteristics of games (graphs) in general, like reduction to DAG or tree-like structures, etc. Another way to find such characteristics might be to directly analyse solution concepts and determine which kind of characteristics of games might ease the computational complexity to determine these solution concept.

7. CONCLUSION

We proved various standard complexity-theoretic results for different variants of shortest path games, results which are interesting in their own right. But more importantly, and apart from the technical results for shortest path games, we motivated a new perspective to analyse complexity-theoretic problems in a game-theoretic context, namely complexity via characteristics of games. For this reason we presented a test environment to motivate this approach. We argued why this new approach is worthwhile to consider and discussed its benefits and limitations. The results we gained for our test setting, motivate our abstract approach to analyse problems for coalitional games only to a limited extent, but this is not surprising given the nature of the newly proposed perspective on coalitional games. Nevertheless, independent from this new perspective, the results showed to a certain extent that it can be interesting to discuss characteristics of games and their influence.

To conclude, we think that an analysis via characteristics, as described in this paper, could be especially interesting in very refined and restricted domains, like networks (or even more specific communication networks), where there is a high similarity between the various cooperative games. This makes it easier to identify characteristics of games in the first place and then use them to model networks having certain computational properties with respect to the application of solution concepts or the determination of player-based properties. Of course, many properties of graphs and networks have been studied in graph theory and some choices of “characteristics” may appear trivial, nevertheless we think that there is a general lack of characteristic-based studies regarding the application of solution concepts to graph-based games and their computational complexity.

7.1 Related and Future Work

Our analysis of the influence of characteristics is (so far) limited to “yes-no” properties. But there is another useful paradigm, which could be rather promising to analyse the computational complexity of coalitional games and the influence of various components of their input (with respect to the problem), namely the application of parametrised complexity [14]. This is particularly interesting in situations where many interesting complexity problems are intractable, like for example in the case of coalition formation for coalition resource games. In this context, the paradigm has re-

cently been advocated by Shrot et al. [26, 27] to determine the parameters of the input that influence the computational complexity of a problem. By fixing particular parameters of the input, they showed that various previously intractable problems are tractable if those parameters are bound. There seems to be quite some potential in using this paradigm to obtain more fine-grained results for intractable problems. Given that structural characteristics, as mentioned above, and parameters used in parametrised complexity are virtually the same with respect to the idea to abstract away from detailed game definitions, we think that it might be interesting to include parameters in our further work.

In general, at this stage of our work, it would be essential to extend the sample space of results by systematically analysing the influence of parameters and characteristics for various games and their complexity-theoretic problems, as well as to think about efficient strategies to obtain influential characteristics and parameters.

Acknowledgments

This paper includes results from my master thesis, which was supervised by Dr. Ulle Endriss. I would like to thank him for all his help and support. Furthermore, I also would like to thank the Netherlands organization for international cooperation in higher education (Nuffic) for supporting me with a Huygens scholarship during my studies at the University of Amsterdam.

8. REFERENCES

- [1] H. Aziz. Complexity of comparison of influence of players in simple games. In *Proc. of the 2nd COMSOC Workshop*, pages 61–72, 2008.
- [2] H. Aziz, O. Lachish, M. Paterson, and R. Savani. Power indices in spanning connectivity games. In *AAIM*, pages 55–67, 2009.
- [3] H. Aziz and M. Paterson. Computing voting power in easy weighted voting games. *International Symposium on Combinatorial Optimization*, 2008.
- [4] Y. Bachrach, R. Meir, M. Zuckerman, J. Rothe, and J. S. Rosenschein. The cost of stability in weighted voting games. In *AAMAS '09: Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems*, pages 1289–1290. International Foundation for Autonomous Agents and Multiagent Systems, 2009.
- [5] Y. Bachrach and J. S. Rosenschein. Power in threshold network flow games. *Autonomous Agents and Multi-Agent Systems*, 18(1):106–132, 2009.
- [6] Y. Bachrach, J. S. Rosenschein, and E. Porat. Power and stability in connectivity games. In *Proc. of the 7th AAMAS conference*, pages 999–1006, 2008.
- [7] J. F. Banzhaf. Weighted voting doesn't work: A mathematical analysis. *Rutgers Law Review*, 19(2):317–343, 1965.
- [8] C. J. Bird. On cost allocation for a spanning tree: A game theoretic approach. *Networks*, 3:335–350, 1976.
- [9] A. Claus and D. J. Kleitman. Cost allocation for a spanning tree. *Network*, 6:335–350, 1973.
- [10] I. Curiel. *Cooperative Game Theory and Applications*. Kluwer Academic, Boston, 1997.
- [11] J. Deegan and P. E.W. A new index of power for simple n-person games. *International Journal of Game Theory*, 7:113–123, 1978.
- [12] X. Deng and C. H. Papadimitriou. On the complexity of cooperative solution concepts. *Math. Oper. Res.*, 19(2):257–266, 1994.
- [13] E. W. Dijkstra. A note on two problems in connection with graphs. *Numerische Mathematik*, 1:269–271, 1959.
- [14] R. G. Downey, M. R. Fellows, R. Niedermeier, P. Rossmanith, and R. G. Parameterized complexity, 1998.
- [15] E. Elkind, L. A. Goldberg, P. W. Goldberg, and M. Wooldridge. Computational complexity of weighted threshold games. In *Twenty-Second AAAI Conference*, 2007.
- [16] U. Faigle, S. Fekete, W. Hochstätter, and W. Kern. On the complexity of testing membership in the core of min-cost spanning tree games. *International Journal of Game Theory*, 26:361–366, 1997.
- [17] V. Fragnelli, I. Garcia-Jurado, and L. Mendez-Naya. On shortest path games. *Mathematical Methods of Operations Research*, 52:251–264, 2000.
- [18] D. Granot and G. Huberman. Minimum cost spanning tree games. *Mathematical Programming*, 21:1–18, 1981.
- [19] M. Holler. Forming coalitions and measuring voting power. *Political Studies*, 30(2):262–271, 1982.
- [20] E. Kalai and E. Zemel. Totally balanced games and games of flow. *Math. Oper. Res.*, 7:476–478, 1982.
- [21] Y. Matsui and T. Matsui. NP-completeness for calculating power indices of weighted majority games. *Theor. Comput. Sci.*, 263(1-2):306–310, 2001.
- [22] N. Megiddo. Computational complexity of the game theory approach to cost allocation for a tree. *Mathematics of Operation Research*, 3(3):189–196, 1978.
- [23] F. Nebel. Shortest path games: Computational complexity of solution concepts. Technical Report 2010-01, ILLC, UvA, February 2010. <http://www.illc.uva.nl/Publications/ResearchReports/MoL-2010-01.text.pdf>.
- [24] K. Prasad and J. S. Kelly. NP-completeness of some problems concerning voting games. *Int. J. Game Theory*, 19(1):1–9, 1990.
- [25] L. Shapley and M. Shubik. A method for evaluating the distribution of power in a committee system. *American Political Science Review*, 48:787–792, 1954.
- [26] T. Shrot, Y. Aumann, and S. Kraus. Easy and hard coalition resource game formation problems: a parameterized complexity analysis. In *Proc. of The 8th AAMAS Conference*, pages 433–440, 2009.
- [27] Y. A. Tammar Shrot and S. Kraus. On agent types in coalition formation problems. In *Proc. of 9th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2010)*, 2010.
- [28] L. G. Valiant. The complexity of enumeration and reliability problems. *SIAM Journal on Computing*, 8(3):410–421, 1979.
- [29] M. Voorneveld and S. Grahn. Cost allocation in shortest path games. *Mathematical methods of operations research*, 56(2):323–340, 2002.