Values for level structures with polynomial-time algorithms, relevant coalition functions, and general considerations

Besner, Manfred

HFT Stuttgart, University of Applied Sciences

30 March 2020

Values for level structures with polynomial-time algorithms, relevant coalition functions, and general considerations

Manfred Besner*

July 1, 2020

Abstract

Exponential runtimes of algorithms for values for games with transferable utility like the Shapley value are one of the biggest obstacles in the practical application of otherwise axiomatically convincing solution concepts of cooperative game theory. We investigate to what extent the hierarchical structure of a level structure improves runtimes compared to an unstructured player set. Representatively, we examine the Shapley levels value, the nested Shapley levels value, and, as a new value for level structures, the nested Owen levels value. For these values, we provide polynomial-time algorithms (under normal conditions) which are exact and therefore not approximation algorithms. Moreover, we introduce relevant coalition functions where all coalitions that are not relevant for the payoff calculation have a Harsanyi dividend of zero. Our results shed new light on the computation of values of the Harsanyi set as the Shapley value and many values from extensions of this set.

Keywords Cooperative game · Polynomial-time algorithm · Level structure · (Nested) Shapley/Owen (levels) value · Harsanyi dividends

1 Introduction

Since the introduction of the Shapley value (Shapley, 1953b), many cooperative game theorists have accumulated an ever-growing pool of axiomatizations of values for cooperative games with transferable utility (*TU-values*). These axiomatizations offer convincing arguments for one or the other TU-value in a variety of situations and applications. But what use is the most beautiful model if the complexity, even for small applications, is so high that they cannot be computed in applicable time or if not all necessary data is available or can be captured?

Within economics, the important concept of *bounded rationality* (Simon, 1972) means that rationally acting individuals must take limited information and cognitive limitations into account in their choices. The time required for decision-making and the limited computing capacity must also be considered. In this respect, we refer, for example, to Futia (1977), Rubinstein (1986), or Kalai and Stanford (1988). Bounded rationality, therefore,

*Manfred Besner
Department of Geomatics, Computer Science and Mathematics,
HFT Stuttgart, University of Applied Sciences, Schellingstr. 24
D-70174 Stuttgart, Germany
E-mail: manfred.besner@hft-stuttgart.de

requires that in deciding which value should we use for the payoff calculation in practice, *computational ease* has always to be satisfied. From a complexity theory perspective, computational ease for a TU-value implies that the payoff can be calculated *efficiently* (i.e. in polynomial-time with respect to the number of players).

Take, e.g., the Shapley value as a central single-valued solution concept. Usually, when computing the Shapley value, the worths of all possible coalitions of players have to be considered. In other words, if n is the number of players, we get an exponential runtime in n, since we have 2^n many different coalitions.[1] Therefore, in general, one has to rely on approximation methods or heuristics, even for a relatively small number of players. Reference is made here, e.g., to Zlotkin and Rosenschein (1994), Castro et al. (2009), and van Campen et al. (2018).

Our approach, at least initially, is to compute exact values. Several classes of coalition functions are known for which we can compute payoffs in polynomial-time using the Shapley value. For example, formulas exist for airport games (Littlechild and Owen, 1973) and for k-games (van den Nouweland et al., 1996), which coincide with weighted hypergraph games with hyperedges of size k (Deng and Papadimitriou, 1994), which require only a selection of all coalitions for computation. Since the number of these coalitions is polynomial in n, the payoff computations can be done efficiently.

For the Shapley levels value (Winter, 1989), Winter introduced a hierarchical structure of coalitions, called *level structure*, which is related to the tree data structure. A level structure comprises a series of ordered partitions (the *levels*) of the player set, each higher level being coarser than the previous one, i.e., each component of a higher level contains at least one or more components of the previous level which together contain the same players (see Figures 1 and 2). Therefore, a level structure can also be seen as an extension of a *coalition structure* (Aumann and Drèze, 1974; Owen, 1977) which has only three levels if we count the partition containing all singletons and the partition containing only the grand coalition as levels.

Meanwhile, some different values for level structures (*LS-values*) exist, like the six values for level structures in Chantreuil (2001), the value for level structures in Gómez-Rúa and Vidal-Puga (2011), the Banzhaf levels value in Alvarez-Mozos and Tejada (2011), or the class of weighted Shapley hierarchy levels values (Besner, 2019b) which contain also the Shapley levels value and the just mentioned LS-value from Gómez-Rúa and Vidal-Puga. Sastre and Trannoy (2002) suggested an extension of their nested Shapley value[2] to level structures which we will call nested Shapley levels value. We find a somewhat different approach in Sánchez-Sánchez and Vargas-Valencia (2018), who proposed a value for cooperative nested games which satisfy nested constraints on a level structure. This value can be seen as an extension of the collective value in Kamijo (2013) for coalition structures.

In this study, we take advantage of the tree-like structure of level structures to obtain algorithms for LS-values which have a polynomial runtime. We investigate the Shapley

[1] This means that we have a huge amount of data to manage. Currently (as of 2019), the largest hard disk drives available on the market have a storage capacity of 16 terabytes, which is less than 2^{44} bytes. But even if we did not have to store the worths of the coalitions, but could somehow use a subroutine that computes them ad hoc at unit cost as proposed in Faigle and Kern (1992), we are already reaching our limits here with a set of maybe 50 players. Purely theoretically, a 3.4 Ghz processor needs already about 92 hours for 2^{50} calculation steps (elementary operations). Even a processor 1000 times faster could only cope with a set of $\log_2 1000 \approx 10$ players more at the same time.

[2] Kamijo (2009) called this value two-step Shapley value.

levels value, the nested Shapley levels value, and as a new LS-value, the nested Owen levels value. Similar to the Shapley levels value, we can this value also interpret as an extension of the Owen value (Owen, 1977) to LS-values. For ordinary level structures, meaning that there are no redundant levels and the number of subcomponents within a component is bounded by a fixed degree, we get polynomial runtimes for algorithms for the last two LS-values mentioned above. If we additionally require that each component of a higher level contains at least two subcomponents in the lower ones, we also obtain a polynomial-time algorithm for the Shapley levels value.

The decisive factor in getting polynomial runtimes is that not all coalitions have to be taken into account in the payoff calculation. We call these coalitions *relevant coalitions*. All other coalitions can take any worth, and we still get the same payoff. This leads us to introduce *relevant coalition functions* where the relevant coalitions receive their original worth and the other coalitions receive a worth so that their (Harsanyi) *dividend* (Harsanyi, 1959) is zero. Dividends can be seen as the cooperation benefits of one coalition over the cooperation benefits of its subcoalitions.

Using relevant coalition functions, we also obtain polynomial runtimes for the Shapley levels value, under the above conditions, if we use the well-known formula with dividends in Calvo et al. (1996, Eq.(1)) as the basis for an algorithm. It turns out that games with relevant coalition functions are closely related to the weighted hypergraph games with variable-size hyperedges, mentioned in Deng and Papadimitriou (1994).

By adapting an algorithm in Algaba et al. (2007), we can compute the dividends of relevant coalitions for a relevant coalition function in polynomial-time if the coalitions are known and their number is polynomially bounded. Thus, we obtain algorithms with polynomial runtime for values with a dividend representation like the values from the Harsanyi set (Hammer et al., 1977; Vasil'ev, 1978) or the proportional Shapley value (Béal et al., 2018; Besner, 2019a) if we know all coalitions with positive or negative dividends and their number is polynomially bounded.

To legitimize the introduced nested Owen levels value and the nested Shapley levels value in cooperative game theory, we provide new axiomatizations in the spirit of balanced contributions as in Calvo et al. (1996).

The paper is organized as follows. Some preliminaries are given in Section 2. Section 3 confirms the exponential runtime of the Shapley value in general and offers some classes of coalition functions, where the Shapley value can be computed efficiently. In Section 4, three LS-values are presented with a short axiomatization, in Section 5, we provide algorithms with polynomial runtime for LS-values, introduce relevant coalition functions and a new formula with dividends for the Shapley levels value. Section 6 generalizes our results, and Section 7 concludes and discusses some ideas for future work. For the sake of readability, the Appendix (Section 8) contains all the more extensive proofs.

2 Preliminaries

2.1 TU-games

Given a countably infinite set $\mathfrak{U}$, the universe of players, we denote by $\mathcal{N}$ the set of all finite subsets of $\mathfrak{U}$. A **TU-game** (N, v) consists of a player set $N \in \mathcal{N}$ and a **coalition function** $v : 2^N \to \mathbb{R}$, $v(\emptyset) = 0$. Each subset $S \subseteq N$, is called a **coalition**, $v(S)$ is called the **worth** of S, Ω^S denotes the set of all non-empty subsets of S, and (S, v) is the **restriction** of (N, v) to the player set $S \in \Omega^N$. We denote by $n := |N|$ the cardinality of N and the set of all TU-games (N, v) is denoted by $\mathbb{V}^N$. A game (N, u_S), $S \in \Omega^N$, defined for all $T \subseteq N$ by $u_S(T) = 1$ if $S \subseteq T$ and $u_S(T) = 0$ otherwise, is called an **unanimity game**. For all $S \subseteq N$, the (Harsanyi) **dividends** $\Delta_v(S)$ (Harsanyi, 1959) are defined inductively by

$$\Delta_v(S) := \begin{cases} 0, \text{ if } S = \emptyset, \\ v(S) - \sum_{R \subsetneq S} \Delta_v(R), \text{ otherwise.} \end{cases} \tag{1}$$

$S \subseteq N$ is called **essential** in (N, v) if $\Delta_v(S) \neq 0$. A player $i \in N$ is called a **dummy player** in (N, v) if $v(S \cup \{i\}) = v(S) + v(\{i\})$, $S \subseteq N \backslash \{i\}$. If we have additionally $v(\{i\}) = 0$, the dummy player i is called a **null player**. Two players $i, j \in N$, $i \neq j$, are called **symmetric** in (N, v), if $v(S \cup \{i\}) = v(S \cup \{j\})$ for all $S \subseteq N \backslash \{i, j\}$.

For all $N \in \mathcal{N}$, a **TU-value** or **solution** ϕ is an operator that assigns to any $(N, v) \in \mathbb{V}^N$ a payoff vector $\phi(N, v) \in \mathbb{R}^N$. For all $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$, the **Shapley value** Sh (Shapley, 1953b) is defined by

$$Sh_i(N, v) := \sum_{S \subseteq N,\, S \ni i} \frac{(|S| - 1)!\,(n - |S|)!}{n!} \big[v(S) - v(S \backslash \{i\})\big] \text{ for all } i \in N. \tag{2}$$

A well-known equivalent formula for the Shapley value is given by

$$Sh_i(N, v) := \sum_{S \subseteq N,\, S \ni i} \frac{\Delta_v(S)}{|S|} \text{ for all } i \in N. \tag{3}$$

We refer to the following axioms for TU-values ϕ on $\mathbb{V}^N$ which hold for all $N \in \mathcal{N}$.

Efficiency^0, E^0. For all $(N, v) \in \mathbb{V}^N$, we have $\sum_{i \in N} \phi_i(N, v) = v(N)$.

Dummy player^0, D^0. For all $(N, v) \in \mathbb{V}^N$ and $i \in N$ a dummy player in (N, v), we have $\phi_i(N, v) = v(\{i\})$.

Additivity^0, A^0. For all $(N, v), (N, w) \in \mathbb{V}^N$, we have $\phi(N, v) + \phi(N, w) = \phi(N, v + w)$.

Symmetry^0, S^0. For all $(N, v) \in \mathbb{V}^N$, and $i, j \in N$ such that i and j are symmetric in (N, v), we have $\phi_i(N, v) = \phi_j(N, v)$.

Balanced contributions^0, BC^0 (Myerson, 1980)**.** For all $(N, v) \in \mathbb{V}^N$ and $i, j \in N$, we have $\phi_i(N, v) - \phi_i(N \backslash \{j\}, v) = \phi_j(N, v) - \phi_j(N \backslash \{i\}, v)$.

Balanced contributions means that for any two players the amount that a player would win or lose if the other player is eliminated from the game should be the same for both players.

2.2 LS-games

Since we also look at games on level structures where coalitions of players are regarded as players, we want to exclude from the outset inconsistencies in the worths of coalitions in the original game and associated coalitions in games where components are the players.[3]

[3] Let, e.g., $N := \{i, j, \{i, j\}\}$ be a player set with three players where the third player is a coalition. In a TU-game $(N, v) \in \mathbb{V}^N$, it does not matter if the worth $v(\{i, j\})$ of the coalition $\{i, j\}$ differ from the worth $v(\{\{i, j\}\})$ of the singleton $\{\{i, j\}\}$.

Therefore, we use in this context only player sets $N \in \mathcal{N}$, where no individual $i \in \mathfrak{U}$ is a sub-member of any two players $j, k \in N$, where an individual $i \in \mathfrak{U}$ is called a sub-member of a player $j \in \mathfrak{U}$ if $i = j$ or j is a set which contains as elements "individuals", or "sets of individuals", or "individuals and sets of individuals", or "sets of sets of individuals", and so on, and i is an element of one of these possibly nested sets. Thus, to avoid complicating the notation, we tacitly assume that, in this context, $\mathcal{N}$ only contains sub-member-disjoint player sets. A partition $\mathcal{B} := \{B_1, ..., B_m\}$ of a player set $N \in \mathcal{N}$, i.e., $B_k \neq \emptyset$ for all k, $1 \leq k \leq m$, $B_k \cap B_\ell = \emptyset$, $1 \leq k < \ell \leq m$, and $\bigcup_{k=1}^{m} B_k = N$, is called a **coalition structure** on N. Each $B \in \mathcal{B}$ is called a **component** and $\mathcal{B}(i)$ denotes the component that contains the player $i \in N$.

For any $N \in \mathcal{N}$, a **level structure** (Winter, 1989) on N, is a finite sequence $\underline{\mathcal{B}} := \{\mathcal{B}^0, ..., \mathcal{B}^{h+1}\}$ of coalition structures $\mathcal{B}^r$, $0 \leq r \leq h+1$, on N such that $\mathcal{B}^0 = \big\{\{i\} \colon i \in N\big\}$, $\mathcal{B}^{h+1} = \{N\}$, and $\mathcal{B}^{r+1}$ is coarser than $\mathcal{B}^r$ for each r, $0 \leq r \leq h$, i.e., $\mathcal{B}^r(i) \subseteq \mathcal{B}^{r+1}(i)$ for all $i \in N$. For each r, $0 \leq r \leq h+1$, $\mathcal{B}^r$ denotes the r-th **level** of $\underline{\mathcal{B}}$. We denote by $\overline{\underline{\mathcal{B}}}$ the set of all components $B \in \mathcal{B}^r$ of all levels $\mathcal{B}^r \in \underline{\mathcal{B}}$, $0 \leq r \leq h$, and $\mathbb{L}^N$ denotes the set of all level structures with player set N.

For $B \in \mathcal{B}^k$, $0 \leq k \leq r \leq h+1$, $\mathcal{B}^r(B)$ denotes the component of the r-th level that contains as a (not necessary proper) subset the component B and is called an **ancestor** of B, if $k < r$. If $r = k+1$, we call the ancestor also **parent** of B. All components with the same parent $B \in \mathcal{B}^r$, $1 \leq r \leq h+1$, are called **children** of B and two different children of B are called **siblings** in $\mathcal{B}^{r-1}$. Note that a component B can be its own parent or child (in different levels). For $B_\ell \in \mathcal{B}^r$, we define $\langle B_\ell \rangle^r := \{B : B \text{ is a child of } \mathcal{B}^{r+1}(B_\ell)\}$ as the set of all children of $\mathcal{B}^{r+1}(B_\ell)$ if $0 \leq r \leq h$, and $\langle B_\ell \rangle^r := \{N\}$ if $r = h+1$. By $|\langle B_\ell \rangle^r|$, $0 \leq r \leq h$, we denote the **degree of the component** $\mathcal{B}^{r+1}(B_\ell)$. The **degree of a level structure** $\underline{\mathcal{B}}$ is the maximal degree of all components $B \in (\overline{\underline{\mathcal{B}}} \cup \{N\})$ which are also parents.

Keep in mind that the definition of level structures also allows identical consecutive levels. A level structure $\underline{\mathcal{B}}$ is called **strict** if $\mathcal{B}^r(i) \subsetneq \mathcal{B}^{r+1}(i)$ for all r, $0 \leq r \leq h$, and at least one $i \in N$, possibly different for each level (see Figure 1), we call $\underline{\mathcal{B}}$ **totally strict** if $\mathcal{B}^r(i) \subsetneq \mathcal{B}^{r+1}(i)$ for all r, $0 \leq r \leq h$, and all $i \in N$ (see Figure 2).

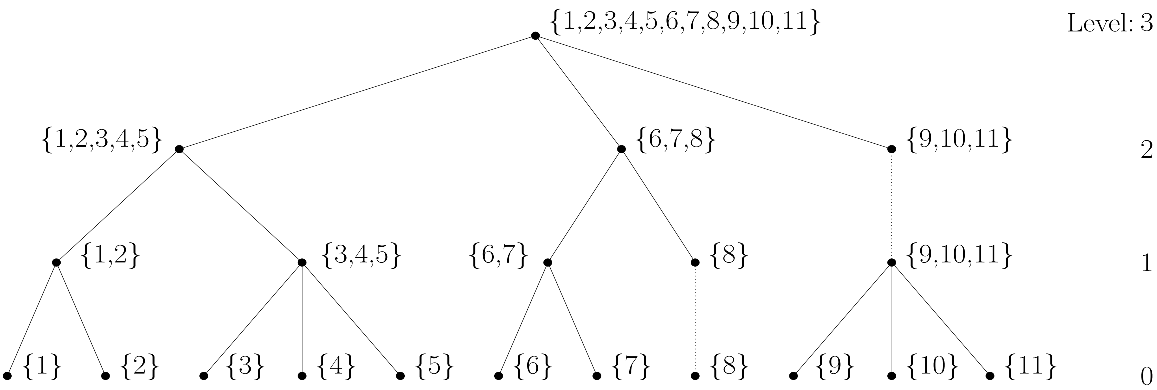

Figure 1: Structure of the components of a strict level structure in different levels

For any $N \in \mathcal{N}$, an **LS-game** is a triple $(N, v, \underline{\mathcal{B}})$ consisting of a TU-game $(N, v) \in \mathbb{V}^N$ and a level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$. We denote the set of all LS-games on N by $\mathbb{VL}^N$.

We define $\underline{\mathcal{B}^r} := \left\{\mathcal{B}^{r^0}, ..., \mathcal{B}^{r^{h+1-r}}\right\} \in \mathbb{L}^{\mathcal{B}^r}$, $0 \leq r \leq h$, as the **induced rth level structure** from $\underline{\mathcal{B}} = \{\mathcal{B}^0, ..., \mathcal{B}^{h+1}\}$, where we regard the components $B \in \mathcal{B}^r$ as players. Each

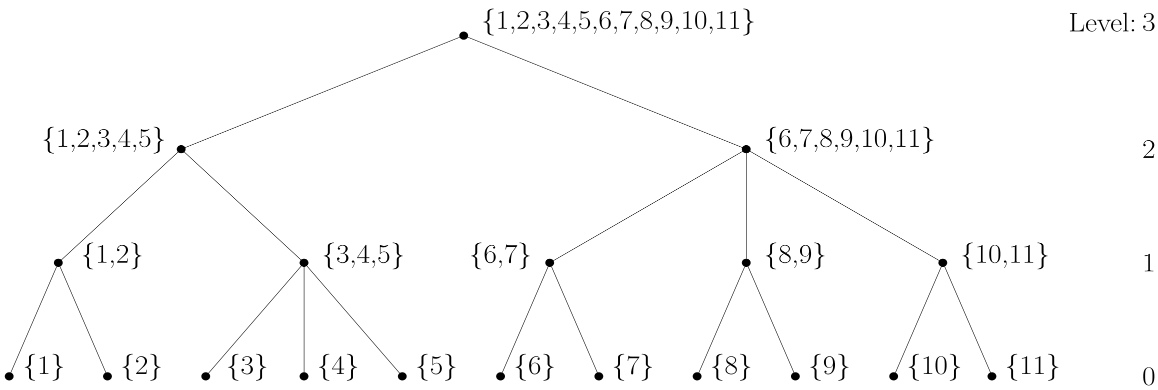

Figure 2: Structure of the components of a totally strict level structure in different levels

element of a coalition structure $\mathcal{B}^{r^k} := \big\{\{B \in \mathcal{B}^r : B \subseteq B'\} : \text{for all } B' \in \mathcal{B}^{r+k}\big\}$, $0 \leq k \leq h+1-r$, is a set of all components of the r-th level which are subsets of the same component of the $(r+k)$-th level. $\big(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}\big) \in \mathbb{VL}^{\mathcal{B}^r}$ is called the **induced rth level game** from $\underline{\mathcal{B}}$ and is given by

$$v^r(\mathcal{Q}) := v\big(\bigcup_{B \in \mathcal{Q}} B\big) \text{ for all } \mathcal{Q} \subseteq \mathcal{B}^r.$$

The following example illustrates our definitions.

Example 2.1. Let $N = \{1,2,3\}$ and $\underline{\mathcal{B}} = \{\mathcal{B}^0, \mathcal{B}^1, \mathcal{B}^2\}$ be given by $\mathcal{B}^0 = \{\{1\},\{2\},\{3\}\}$, $\mathcal{B}^1 = \{\{1\},\{2,3\}\}$, and $\mathcal{B}^2 = \{N\}$. We regard, e.g., the components of the first level as players. Then, the induced first level structure $\underline{\mathcal{B}^1} = \{\mathcal{B}^{1^0}, \mathcal{B}^{1^1}\}$ from $\underline{\mathcal{B}}$ is given by $\mathcal{B}^{1^0} = \big\{\{\{1\}\}, \{\{2,3\}\}\big\}$, and $\mathcal{B}^{1^1} = \big\{\{\{1\},\{2,3\}\}\big\}$.

For $T \in \Omega^N$ and coalition structures $\mathcal{B}^r|_T := \{B \cap T : B \in \mathcal{B}^r,\ B \cap T \neq \emptyset\}$, $0 \leq r \leq h+1$, we denote by $\underline{\mathcal{B}}|_T := \{\mathcal{B}^0|_T, ..., \mathcal{B}^{h+1}|_T\} \in \mathbb{L}^T$ the **restricted** level structure of $\underline{\mathcal{B}}$ on T. Then, $(T, v, \underline{\mathcal{B}}|_T) \in \mathbb{VL}^T$ is called the **restriction** of $(N, v, \underline{\mathcal{B}})$ to T and $\big(\mathcal{B}^r|_T, v^r, \underline{\mathcal{B}^r}|_T\big) \in \mathbb{VL}^{\mathcal{B}^r|_T}$ is the induced rth level game from the restriction of $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ on T.

For $0 \leq r \leq h$, $\underline{\mathcal{B}_r} := \big\{\mathcal{B}^0, ..., \mathcal{B}^r, \{N\}\big\} \in \mathbb{L}^N$ is called the **rth cut level structure** from $\underline{\mathcal{B}}$ where all levels between the rth and the $(h+1)$th level are cut out from $\underline{\mathcal{B}}$. $(N, v, \underline{\mathcal{B}_r})$ is called the **rth cut** of $(N, v, \underline{\mathcal{B}})$. Notice that for each $\underline{\mathcal{B}} = \{\mathcal{B}^0, ..., \mathcal{B}^{h+1}\}$ we also have $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$. Thus, we often write briefly $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$ to make clear how many levels the level structure comprises. For each $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ with a **trivial level structure** $\underline{\mathcal{B}} = \underline{\mathcal{B}_0}$ exists a corresponding TU-game (N, v) and for each $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ with $\underline{\mathcal{B}} = \underline{\mathcal{B}_1}$ exists a corresponding game with coalition structure (Aumann and Drèze, 1974; Owen, 1977).

For all $N \in \mathcal{N}$, an **LS-value** φ is an operator that assigns to any $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ a payoff vector $\varphi(N, v, \underline{\mathcal{B}}) \in \mathbb{R}^N$.

Let $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $T \in \Omega^N$, $T \ni i$, and

$$K_{\underline{\mathcal{B}},T}(i) := \prod_{r=0}^{h} K^r_{\underline{\mathcal{B}},T}(i), \quad \text{where } K^r_{\underline{\mathcal{B}},T}(i) := \frac{1}{|\{B \in \mathcal{B}^r : B \subseteq \mathcal{B}^{r+1}(i),\ B \cap T \neq \emptyset\}|}. \tag{4}$$

Then, the **Shapley Levels value** Sh^L (Winter, 1989) is given by[4]

[4]This formula for the Shapley levels value comes from Calvo et al. (1996, Eq.(1)).

$$Sh_i^L(N, v, \underline{\mathcal{B}}) := \sum_{T \subseteq N, T \ni i} K_{\underline{\mathcal{B}},T}(i) \Delta_v(T) \text{ for all } i \in N. \tag{5}$$

If $h = 0$, Sh^L coincides with Sh; if $h = 1$, a level structure coincides with a coalition structure and it is well-known, that the **Owen value** Ow (Owen, 1977) can therefore alternatively, as a special case of the Shapley levels value, be defined by

$$Ow_i(N, v, \underline{\mathcal{B}_1}) := \sum_{T \subseteq N, T \ni i} K_{\underline{\mathcal{B}_1},T}(i) \Delta_v(T) \text{ for all } i \in N.$$

We refer to the following axioms for LS-values φ on $\mathbb{VL}^N$ which hold for all $N \in \mathcal{N}$.

Efficiency, E. For all $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, we have $\sum_{i \in N} \varphi_i(N, v, \underline{\mathcal{B}}) = v(N)$.

Null player, N. For all $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ and $i \in N$ a null player in (N, v), we have $\varphi_i(N, v, \underline{\mathcal{B}}) = 0$.

Level game property, LG (Winter, 1989)**.** For all $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $B \in \mathcal{B}^r$, $0 \leq r \leq h$, we have

$$\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \varphi_B(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}).$$

This property states that the total payoff obtained by all members of a component is equal to the component's payoff in the corresponding induced level game where the component is regarded as a player.

Balanced contributions, BC (Calvo et al., 1996)**.** For all $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, and two siblings $B_k, B_\ell \in \mathcal{B}^r$, $0 \leq r \leq h$, we have

$$\sum_{i \in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_k} \varphi_i(N \backslash B_\ell, v, \underline{\mathcal{B}}|_{N \backslash B_\ell}) = \sum_{i \in B_\ell} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_\ell} \varphi_i(N \backslash B_k, v, \underline{\mathcal{B}}|_{N \backslash B_k}).$$

BC means that for any two siblings, the sum of the amount that all players of one sibling would win or lose if the other sibling is eliminated from the game should be the same for both siblings.

2.3 Notes on time complexity

By time complexity we understand an estimation of the time to run an algorithm. Usually, the time is specified by the number of elementary operations the algorithm needs to execute. For simplicity's sake, a fixed constant time is assumed for each elementary operation. If we are interested in an (asymptotic) upper bound, we use big-O notation. In case that we are interested in a (asymptotic) lower bound, we use the big-Ω notation as suggested by Knuth (1976). Normally, the argument of the function used within the big-O or the big-Ω notation is the input size. In this respect, we cite Deng and Papadimitriou (1994) who stated the following:

> "There is a catch, however: If the game is defined by the 2^n coalition values, there may be little to be said about the computational complexity of the various solution concepts, because the input is already exponential in n, and thus, in most cases, the computational problems above can be solved very 'efficiently'."

It is therefore common practice in this context, to use the number of players as the reference for the time complexity analysis. Hence, we say that an algorithm is **efficient** if it runs in polynomial-time with respect to the number n of players.

Notation 2.2. By $t(A)$ we denote the number of elementary operations of algorithm A, by $t(F_r)$ those within the FOR-loop starting in line r, by $t(L_r)$ those of the assignments within line r, and by $t(IF_r)$ and $t(ELSE_r)$ those within the IF- or ELSE-branch starting in line r.

3 The Shapley value

If we look at formulas (2) or (3) for computing the Shapley value, we see that even the input of the used worths or dividends requires exponential time. But are we perhaps simply not yet able to find an algorithm that does not need the worths of all coalitions for the input? We will see later that for the Shapley levels value, which has with formula (5) a very similar formula to formula (3), a formula can be found which, except in degenerated cases, only requires the worths of polynomially many coalitions. Whether linear programs can be solved in polynomial-time has long been an open problem, especially when it became clear that the simplex algorithm as the main solution method requires exponential time. Finally, the ellipsoid algorithm in Khachiyan (1979) showed that linear programs are solvable in polynomial-time. However, the fact that generally no algorithm with polynomial-time can be found for the Shapley value is confirmed by the following proposition.[5]

Proposition 3.1. *There is no algorithm that computes the Shapley value in polynomial-time for all $(N, v) \in \mathbb{V}^N$ and $N \in \mathcal{N}$ with respect to the number of players n.*

Proof. Let $N \in \mathcal{N}$, $(N, v_1), (N, v_2) \in \mathbb{V}^N$, $K \subseteq N$, $v_1(K) \neq v_2(K)$, and $v_1(S) = v_2(S)$ for all $S \in \Omega^N, S \neq K$, such that each worth $v_1(S), v_2(S)$ is independent from all other worths $v_1(T), v_2(T)$, $T \in \Omega^N, T \neq S$. By (2), we have, for all $i \in K$,

$$\begin{aligned}
&Sh_i(N, v_2) - Sh_i(N, v_1)\\
&\quad = \sum_{\substack{S\subseteq N,\\ S\ni i}} \frac{(s-1)!(n-s)!}{n!}\big[v_2(S) - v_2(S\backslash\{i\})\big] - \sum_{\substack{S\subseteq N,\\ S\ni i}} \frac{(s-1)!(n-s)!}{n!}\big[v_1(S) - v_1(S\backslash\{i\})\big]\\
&\quad = \frac{(k-1)!(n-k)!}{n!}\big[v_2(K) - v_1(K)\big] \neq 0,
\end{aligned}$$

where $k := |K|$ and $s := |S|$. In other words, any algorithm that computes the Shapley value returns a different result for the two coalition functions v_1, v_2. Therefore, since K was arbitrary, the payoff to a player i depends on each worth of the 2^{n-1} coalitions $S \subseteq N$ containing the player i as long as the worths of the coalitions are independent of each other. Consequently, all worths must be used at least once in the algorithm, i.e. they require at least one elementary operation, which corresponds to a runtime of $\Omega(2^{n-1})$ for a single player. □

Erroneously stated in Castro et al. (2009) and van Campen et al. (2018)), the computation of the Shapley value is not an NP-complete problem.

[5] A similar result can be found in Faigle and Kern (1992, Theorem 3).

Remark 3.2. *As the proof of Proposition 3.1 shows, not even a guess about the solution generated in a non-deterministic way can be verified in polynomial time with respect to the number of players by a deterministic Turing machine.*[6]

Fortunately, there are some classes of games where the Shapley value can be computed efficiently. Airport games are one possibility, as shown in Littlechild and Owen (1973). This type of cost games can be decomposed into a sum of games where all players are symmetric or null players. Therefore, here the additive Shapley value can be calculated very efficiently by symmetry and the null player property of the value.

Another possibility are k-games, introduced by van den Nouweland et al. (1996). A k-game coincides to a weighted hypergraph game with hyperedges of size k, introduced by Deng and Papadimitriou (1994). A TU-game is called a **k-game** if the coalition function takes the form:

$$v(S) = \sum_{T \subseteq S,\, |T|=k} v(T),\ k \geq 0.$$

As long as k is fixed and thus does not depend on n, we can compute the Shapley value for such games in polynomial-time. This aspect is discussed in more detail in Section 6. As a more recent result, we refer to Maafa et al. (2018), where it is shown that the Shapley value of weighted graph games on a product of chains of equal fixed length in polynomial time can be computed.

4 Values for level structures

In this section, we examine LS-values that generalize the Shapley value to LS-games and calculate the payoff in a top-down procedure: We distribute the worth of the grand coalition to its children, the components of the hth level, using a TU-value. Then, each payoff of a component of the hth level is divided by the same TU-value among all its children, and so on for all levels. Finally, we distribute the payoffs of the first level components to their children and thus to the original players. The various LS-values differ in the definition of the intermediate games[7], i.e., the steps for each level.

4.1 The Shapley levels value as a weighted Shapley hierarchy levels value

We recall the definition of the Shapley levels value as a special case of the weighted Shapley hierarchy levels values (Besner, 2019b) and a related notation.

Notation 4.1. For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $i \in N$, and $T \in \Omega^{\mathcal{B}^k(i)}$, $0 \leq k \leq h$, we denote by $\mathcal{T}^{\mathcal{B}^k(i)} := \{B \in \mathcal{B}^k : B \subseteq \mathcal{B}^{k+1}(i),\, B \neq \mathcal{B}^k(i)\} \cup \{T\}$ the set of all children of the component $\mathcal{B}^{k+1}(i)$, where the child $\mathcal{B}^k(i)$ is replaced by coalition T.

Definition 4.2. (see Besner (2019b, Remark 3.5)) *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $i \in N$, and for all k, $0 \leq k \leq h$, $T \in \Omega^{\mathcal{B}^k(i)}, \mathcal{T}^{\mathcal{B}^k(i)}$ the set from Notation 4.1, define*

[6] Actually, this is not really a decision problem but a #P-complete counting problem (see Faigle and Kern (1992) and Brightwell and Winkler (1991)).

[7] Owen (1977) called such games quotient games.

$\bar{v}_i^{h+1} := v$, *and* $\bar{v}_i^k$ *by*

$$\bar{v}_i^k(T) := Sh_T(\mathcal{T}^{\mathcal{B}^k(i)}, \tilde{v}_i^k) \text{ for all } T \in \Omega^{\mathcal{B}^k(i)},$$

where $\tilde{v}_i^k$ *is specified recursively via*

$$\tilde{v}_i^k(\mathcal{Q}) := \bar{v}_i^{k+1}(\bigcup_{S \in \mathcal{Q}} S) \text{ for all } \mathcal{Q} \subseteq \mathcal{T}^{\mathcal{B}^k(i)}.$$

Then the Shapley levels value Sh^L *is given by*

$$Sh_i^L(N, v, \mathcal{B}) := \bar{v}_i^0(\{i\}) \text{ for all } i \in N.$$

We use the following axiomatization as a starting point for further axiomatizations.

Theorem 4.3. (Calvo et al., 1996) *Sh^L is the unique LS-value that satisfies **E** and **BC**.*

4.2 The nested Shapley levels value

In many hierarchically structured organizations, it is common for the actors of a single organizational unit to act only among themselves. Interaction across organizational units only takes place at a higher level. The top-down payoff calculation of the following value is based on this principle.

Definition 4.4. *Let $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $i \in N$, $\bar{v}_i^{h+1}(N) := v(N)$, and for all k, $0 \leq k \leq h$, be $\bar{v}_i^k(\mathcal{B}^k(i))$ given by*

$$\bar{v}_i^k(\mathcal{B}^k(i)) := Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{v}_i^k), \tag{6}$$

where $\tilde{v}_i^k$ *is specified recursively via*

$$\tilde{v}_i^k(\mathcal{Q}) := \begin{cases} \bar{v}_i^{k+1}(\mathcal{B}^{k+1}(i)), & \text{if } \mathcal{Q} = \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \\ v(\bigcup_{B \in \mathcal{Q}} B) & \text{if } \mathcal{Q} \subsetneq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}. \end{cases} \tag{7}$$

*Then the **nested Shapley levels value** Sh^{NL}, suggested in Sastre and Trannoy (2002), is given by*

$$Sh_i^{NL}(N, v, \mathcal{B}) := \bar{v}_i^0(\{i\}) \text{ for all } i \in N.$$

Remark 4.5. *Due to the additivity of the Shapley value, we can interpret the top-down distribution mechanism also in this way: Within each (parent) component, there is a recursive two-step bargaining process. In a first step, the children divide as players in a game, restricted to their parent, the original worth of the parent via the Shapley value. In a second step, the surplus that the parent has received as a player over what it has earned alone is additionally distributed evenly among the children. We obtain the following equivalent definition that especially shows the coincidence of the value with the nested Shapley value[8] defined in Sastre and Trannoy (2002) in case of a level structure with $h = 1$:*

For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, Sh^{NL} is recursively defined by

$$Sh^{NL}_{\mathcal{B}^k(i)}(\mathcal{B}^k, v^k, \underline{\mathcal{B}^k}) := \begin{cases} Sh_{\mathcal{B}^h(i)}(\mathcal{B}^h, v^h), & \text{if } k = h, \\ Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, v^k) + \dfrac{Sh^{NL}_{\mathcal{B}^{k+1}(i)}(\mathcal{B}^{k+1}, v^{k+1}, \underline{\mathcal{B}^{k+1}}) - v(\mathcal{B}^{k+1}(i))}{|\langle \mathcal{B}^k(i) \rangle^k|}, & \text{if } 0 \leq k < h, \end{cases} \tag{8}$$

and $Sh_i^{NL}(N, v, \underline{\mathcal{B}}) := Sh^{NL}_{\{i\}}(\mathcal{B}^0, v^0, \underline{\mathcal{B}^0})$ *for all* $i \in N$.

[8] Kamijo (2009) called this value two-step Shapley value.

We introduce a new axiom that coincides obviously for a trivial level structure with **BC**.

Nested balanced contributions, NBC. For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, two siblings $B_k, B_\ell \in \mathcal{B}^r$, $0 \le r \le h$, we have

$$\sum_{i \in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_k} \varphi_i\big(\mathcal{B}^{r+1}(i) \backslash B_\ell, v, \underline{\mathcal{B}_r}|_{\mathcal{B}^{r+1}(i) \backslash B_\ell}\big) = \sum_{i \in B_\ell} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_\ell} \varphi_i\big(\mathcal{B}^{r+1}(i) \backslash B_k, v, \underline{\mathcal{B}_r}|_{\mathcal{B}^{r+1}(i) \backslash B_k}\big).$$

An interpretation of this property would be as follows: The sum of the amount that all players of one sibling would win or lose if the other sibling dropped out of the game and this would result in a game then being played only within the parent component and no longer on the entire level structure, should be the same for both siblings. Of course, the higher redundant levels are then obsolete.

Proposition 4.6. *Sh^{NL} satisfies **E**, **LG**, and **NBC** but not **N**.*

We present an axiomatization of the nested Shapley levels value.

Theorem 4.7. *Sh^{NL} is the unique LS-value that satisfies **E** and **NBC**.*

Similar as for the weighted Shapley hierarchy Shapley levels values, we can obtain the class of **nested weighted Shapley levels values**, if we replace the Shapley value in (6) with a weighted Shapley value (Shapley,1953a). A **nested weighted balanced contributions** axiom could be used for axiomatization. We will not go into that here.

4.3 The nested Owen levels value

We can now imagine that the active interaction of components which are siblings no longer takes place only within the parent component, but also with the siblings of the parent component or even with siblings of other ancestors. The extreme case is the Shapley levels value that takes into account the siblings of all ancestors in the payoff calculation. By the following LS-value, we consider only the siblings of the parent component. The same approach, restricted to a coalition structure, is used by Owen (1977) in his famous value. Therefore, our LS-value, like the Shapley levels value, can be seen as an extension of the Owen value to level structures. Again, we use a notation. In contrast to Notation 4.1, here we define a set of players with components as players, where one component can only be replaced by unions of its children and not by any subset.

Notation 4.8. Let $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $i \in N$, $S \in \Omega^{\mathcal{B}^k(i)}$ be such that $S = \bigcup_{B \in \mathcal{B}^{k-1}, B \subseteq S} B$ is a union of children of $\mathcal{B}^k(i)$ if $1 \le k \le h$, and $S = \{i\}$ if $k = 0$. We denote by $\mathcal{S}^{\mathcal{B}^k(i)} := \{B \in \mathcal{B}^k : B \subseteq \mathcal{B}^{k+1}(i),\, B \neq \mathcal{B}^k(i)\} \cup \{S\}$ the set containing all children of the component $\mathcal{B}^{k+1}(i)$, where the child $\mathcal{B}^k(i)$ is replaced by coalition S.

Definition 4.9. *Let Let $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $i \in N$, define $\bar{v}_i^{h+1} := v$, and let $\bar{v}_i^k(S)$ for all $S \subseteq \mathcal{B}^k(i)$, $S = \bigcup_{B \in \mathcal{B}^{k-1}, B \subseteq S} B$ if $1 \le k \le h$, or for $S = \{i\}$ if $k = 0$, be given by*

$$\bar{v}_i^k(S) := \begin{cases} Sh_S(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{v}_i^k), & \text{if } S = \mathcal{B}^k(i), \\ Sh_S(\mathcal{S}^{\mathcal{B}^k(i)}), \ \mathcal{S}^{\mathcal{B}^k(i)} \text{ the set from Notation 4.8}, & \text{otherwise,} \end{cases}$$

where v_i^k is given by

$$v_i^k(\mathcal{Q}) := v(\bigcup_{T\in\mathcal{Q}} T) \text{ for all } \mathcal{Q} \subseteq \mathcal{S}^{\mathcal{B}^k(i)}$$

and $\tilde{v}_i^k$ is specified recursively via

$$\tilde{v}_i^k(\mathcal{Q}) := \bar{v}_i^{k+1}(\bigcup_{T\in\mathcal{Q}} T) \text{ for all } \mathcal{Q} \subseteq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}.$$

Then the ***nested Owen levels value*** *Ow^{NL} is given by*

$$Ow_i^{NL}(N, v, \mathcal{B}) := \bar{v}_i^0(\{i\}) \text{ for all } i \in N.$$

Remark 4.10. *Due to the additivity of the Shapley value (and thus of the Owen and the nested Owen levels value), similar to the nested Shapley levels value, we can give an alternative definition of the nested Owen levels value that justifies the naming. Within each parent of a (parent) component B, a recursive two-step bargaining process is installed. In a first step, all children of B receive as players in a game, restricted to the parent of B, a share of the original worth of the parent of B via the Owen value. In a second step, the surplus that B as a player on the whole game has received over what it has earned in the restriction on its parent is additionally distributed evenly among the children of B. We obtain the following equivalent definition, where $\underline{\mathcal{B}_{k+1}^k}|_{\mathcal{B}^{k+2}(i)}$ means the induced kth level structure of the $(k+1)$th cut of $\underline{\mathcal{B}}|_{\mathcal{B}^{k+2}(i)}$:*

For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, Ow^{NL} is recursively defined by

$$\begin{aligned} &Ow^{NL}_{\mathcal{B}^k(i)}(\mathcal{B}^k, v^k, \underline{\mathcal{B}^k}) \\ &:= \begin{cases} Sh_{\mathcal{B}^h(i)}(\mathcal{B}^h, v^h), \ \text{if } k = h, \\ Ow_{\mathcal{B}^k(i)}\big(\mathcal{B}^k|_{\mathcal{B}^{k+2}(i)}, v^k, \underline{\mathcal{B}^k_{k+1}}|_{\mathcal{B}^{k+2}(i)}\big) \\ + \dfrac{Ow^{NL}_{\mathcal{B}^{k+1}(i)}\big(\mathcal{B}^{k+1}, v^{k+1}, \underline{\mathcal{B}^{k+1}}\big) - Sh_{\mathcal{B}^{k+1}(i)}\big(\mathcal{B}^{k+1}|_{\mathcal{B}^{k+2}(i)}, v^{k+1}\big)}{|\left\langle \mathcal{B}^k(i)\right\rangle^k|}, \ \text{if } 0 \le k \le h-1, \end{cases} \end{aligned} \tag{9}$$

and $Ow_i^{NL}(N, v, \underline{\mathcal{B}}) = Ow^{NL}_{\{i\}}(\mathcal{B}^0, v^0, \underline{\mathcal{B}^0})$ for all $i \in N$.

Remark 4.11. *Ow^{NL} coincides with Sh if $h = 0$ and with Ow if $h = 1$.*

The following property is similar to **NBC**.

Nested balanced Owen contributions, NBOC. For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, two siblings $B_k, B_\ell \in \mathcal{B}^r$, $0 \le r \le h$, we have

$$\begin{aligned} &\sum_{i\in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_k} \varphi_i\big(\mathcal{B}^{r+2}(i)\backslash B_\ell, v, \underline{\mathcal{B}_{r+1}}|_{\mathcal{B}^{r+2}(i)\backslash B_\ell}\big) \\ &\qquad = \sum_{i\in B_\ell} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_\ell} \varphi_i\big(\mathcal{B}^{r+2}(i)\backslash B_k, v, \underline{\mathcal{B}_{r+1}}|_{\mathcal{B}^{r+2}(i)\backslash B_k}\big), \end{aligned} \tag{10}$$

where $\mathcal{B}^{r+2}(i) := \mathcal{B}^{h+1}(i)$ and $\underline{\mathcal{B}_{r+1}} := \underline{\mathcal{B}_h}$ if $r = h$.

Also the interpretation is similar to **NBC**. Suppose one sibling leaves the game and this would lead to a situation where the other sibling can only play a game within the parent component of its parent (without its sibling). Then the sum of the payoffs that all players of a sibling win or lose is the same for both siblings.

Proposition 4.12. *Ow^{NL} satisfies **E**, **LG**, and **NBOC** but not **N**.*

Theorem 4.13. *Ow^{NL} is the unique LS-value that satisfies **E** and **NBOC**.*

5 Runtime complexity for algorithms of LS-values

As far as we know, there are no studies of how the extension of a solution such as the Shapley value to an LS-value such as the Shapley levels value affects time complexity. The hierarchical structure of level structures is related to the data structure of trees in computer science or rooted trees in graph theory[9]. In computer science, trees are one of the most fundamental concepts for coping with complexity. In this context, only the use of trees in databases, hierarchical file systems in operating systems, or search trees for the management of information should be mentioned. We will show below that level structures can analogously reduce complexity.

Proposition 5.1. *For all $N \in \mathcal{N}$ and each level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, we have*

(i) *$h \le n-2$, if $\underline{\mathcal{B}}$ is strict,*

(ii) *$h \le (\log_2 n) - 1$, if $\underline{\mathcal{B}}$ is totally strict.*

Proof. Let $N \in \mathcal{N}$.

(i) For a strict level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, we have $|\mathcal{B}^{r+1}| < |\mathcal{B}^r|$ for all r, $0 \le r \le h$. Due to $|\mathcal{B}^0| = n$, it follows $|\mathcal{B}^{h+1}| \le n - (h+1)$ and thus, by $|\mathcal{B}^{h+1}| = 1$, $h \le n-2$.

(ii) Let $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, be totally strict. If $h = 0$, we have $2^{h+1} = 2 \le n$. For each additional level, the size of the player set must at least double. Therefore, for arbitrary h, we have $2^{h+1} \le n \Leftrightarrow h \le (\log_2 n) - 1$. $\square$

5.1 Relevant coalitions

In the following, we want to state that we only need the worths of certain coalitions for the computation of the LS-values in Section 4. In principle, the same observations also apply to corresponding weighted variants.

Remark 5.2. *For all $N \in \mathcal{N}$, to compute the Shapley levels value for a player $i \in N$ and a level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, based on Definition 4.2, we need only to take into account the worths of two groups of coalitions $T \subseteq N$: first, the singleton containing the player i, all siblings of this singleton, and all siblings of ancestors of this singleton and second, all coalitions that these components can form as unions. We denote the set of all these coalitions by $\mathcal{R}^i_{\underline{\mathcal{B}}}$ as the set of **relevant coalitions** for player i on $\underline{\mathcal{B}}$. The worths of all other coalitions $S \in \Omega^N \backslash \mathcal{R}^i_{\underline{\mathcal{B}}}$ can take any worth and we get the same payoff for player i.*

[9]In a different perspective, Álvarez-Mozos et al. (2017) describe how hierarchical structures can be transformed into level structures.

Remark 5.3. *For all $N \in \mathcal{N}$, to compute the nested Shapley levels value for a player $i \in N$ and a level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, based on Definition 4.4, we need only to take into account the worths of two groups of coalitions $T \subseteq N$: first, all components $B \in \overline{\underline{\mathcal{B}}}$, $B \ni i$, and their siblings, and second all coalitions that children within one parent, containing player i, can form as unions among themselves. We denote the set of all coalitions from these both groups by $\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}$ as the set of* ***relevant nested Shapley coalitions*** *for player i on $\underline{\mathcal{B}}$. The worths of all coalitions $S \in \Omega^N \backslash \mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}$ can take any worth and we get the same payoff.*

Remark 5.4. *For all $N \in \mathcal{N}$, to compute a nested Owen levels value for a player $i \in N$ and a level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, based on Definition 4.9, we need only to take into account the worths of three groups of coalitions $T \subseteq N$: first, all components $B \in \overline{\underline{\mathcal{B}}}$, $B \ni i$, and their siblings, and second all coalitions that children within one parent, containing player i, can form as unions among themselves, and third all coalitions that each of these coalitions can form with siblings of their parent as unions. We denote the set of all coalitions from these three groups by $\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}$ as the set of* ***relevant nested Owen coalitions*** *for player i on $\underline{\mathcal{B}}$. The worths of all coalitions $S \in \Omega^N \backslash \mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}$ can take any worth and we get the same payoff.*

5.2 Runtimes of LS-values defined in a top-down procedure

If the degree of a level structure is not bounded, we cannot expect to find a polynomial-time algorithm for our LS-values, since, e.g., all values for a trivial level structure coincide with the Shapley value. Therefore, we use level structures of fixed degree for the algorithms. Note that this means that the level structure must contain a certain number of levels, depending on the degree and size of the player set. First, we indicate the complexities of the intermediate games.

Theorem 5.5. *For all $D \in \mathcal{N}$, $(D, v) \in \mathbb{V}^D$, $d := |D|$, it requires to compute $Sh_i(D, v)$ for a single player $i \in D$ a time $O(d2^d)$.*

Proof. We give a pseudocode algorithm based on Formula (2).

Algorithm 5.1. *Compute $Sh_i(D, v)$*

Input: *A player $i \in D$ and $v(S)$ for all $S \subseteq D$.*

1: $sum := 0$
2: ***for all*** $S \subseteq D$, $S \ni i$, ***do***
3: $\quad sum := sum + \dfrac{(|S| - 1)!\,(d - |S|)!}{d!}\big[v(S) - v(S \backslash \{i\})\big]$ // (2)
4: ***end for***
5: $Sh_i(D, v) := sum$
6: ***return*** $Sh_i(D, v)$.

Complexity: We have $t(Algorithm\ 5.1) = 1 + t(F_2) + 1 = 2 + 2^{d-1}t(L_3)$. If the factorials are not stored, we have $t(L_3) \in O(d)$. Therefore, Algorithm 5.1 has a time $O(d2^d)$. □

We give the complexities of our LS-values.

Theorem 5.6. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ such that $\underline{\mathcal{B}}$ is a totally strict level structure of fixed degree d, it requires to compute $Sh_i^L(N, v, \underline{\mathcal{B}})$ for all players $i \in N$ a time $O(n^d \log n)$.*

Proof. We give a pseudocode algorithm based on Definition 4.2.

Algorithm 5.2. *Compute* $Sh_i^L(N, v, \underline{\mathcal{B}})$

Input: *A level structure* $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, *a player* $i \in N$, *and* $v(S)$ *for all* $S \in \mathcal{R}^i_{\underline{\mathcal{B}}}$.

1: **for all** $S \in \mathcal{R}^i_{\mathcal{B}}$ **do** // the relevant coalitions for player i

2: $\quad \bar{v}^{h+1}(S) := v(S)$

3: **end for**

4: **for** $k = h$ **to** 0 **do** // the descending levels

5: $\quad$ **for all** $T \in \Omega^{\mathcal{B}^k(i)} \cap \mathcal{R}^i_{\mathcal{B}}$ **do** // all subsets of component $\mathcal{B}^k(i)$ which are relevant coalitions for player i

6: $\quad\quad$ **for all** $\mathcal{Q} \subseteq \mathcal{T}^{\mathcal{B}^k(i)}$ **do** // all subsets from $\mathcal{T}^{\mathcal{B}^k(i)}$, defined in Notation 4.1

7: $\quad\quad\quad \tilde{v}^k_i(\mathcal{Q}) := \bar{v}^{k+1}_i(\bigcup_{S \in \mathcal{Q}} S)$

8: $\quad\quad$ **end for**

9: $\quad\quad \bar{v}^k_i(T) := Sh_T(\mathcal{T}^{\mathcal{B}^k(i)}, \tilde{v}^k_i)$ // calls a method/function that computes Sh before the assignment, e.g. Algorithm 5.1

10: $\quad$ **end for**

11: **end for**

12: $Sh_i^L(N, v, \underline{\mathcal{B}}) := \bar{v}^0_i(\{i\})$

13: **return** $Sh_i^L(N, v, \mathcal{B})$.

Complexity: Let $\underline{\mathcal{B}}$ be a totally strict level structure of degree d. We have, by Proposition 5.1, $h \leq (\log_2 n) - 1$. It follows

$$|\mathcal{R}^i_{\mathcal{B}}| \leq 2^d \cdot \underbrace{2^{d-1} \cdots 2^{d-1}}_{\text{h times}} - 1 \leq 2 \cdot \underbrace{2^{d-1} \cdots 2^{d-1}}_{\log_2 n \text{ times}} - 1 = 2 \cdot 2^{\log_2 n(d-1)} - 1 = 2n^{d-1} - 1. \tag{11}$$

In line 6, we have $|\mathcal{T}^{\mathcal{B}^k(i)}| \leq d$. It follows

$$t(F_6) \leq 2^d. \tag{12}$$

Thus, we have

$$\begin{aligned} t(Algorithm\ 5.2) \quad &= \quad t(F_1) + t(F_4) + 1 \underset{(11)}{\leq} 2n^{d-1} + \textstyle\sum_{k=h}^{0} t(F_5) \\ &\underset{Prop. 5.1, (11)}{\leq} \quad 2n^{d-1} + (\log_2 n) 2n^{d-1} \big[t(F_6) + t(L_9) \big] \\ &\underset{(12)}{\leq} \quad 2n^{d-1} + (\log_2 n) 2n^{d-1} 2^d + (\log_2 n) 2n^{d-1} t(L_9). \end{aligned}$$

By Theorem 5.5, we have $t(L_9) \in O(d2^d)$. Therefore, Algorithm 5.2 has a time $O(n^{d-1} \log n)$. The claim follows by running the algorithm for n players. □

Remark 5.7. *Theorem 5.6 remains valid for arbitrary level structures of degree d as long as h is logarithmic in n.*

Despite this generally positive result, the time complexity of computing the level structure value may be too high in a number of cases. In practice, the degree of $\underline{\mathcal{B}}$ must be small, even if n is not very large. In these cases, using the nested Shapley levels value may be more appropriate.

Theorem 5.8. *For all* $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, *and* $\underline{\mathcal{B}}$ *of fixed degree d, it requires to compute* $Sh_i^{NL}(N, v, \underline{\mathcal{B}})$ *for all players* $i \in N$

(i) *a time* $O(n^2)$ *if* $\underline{\mathcal{B}}$ *is strict,*

(ii) *a time* $O(n \log n)$ *if* $\underline{\mathcal{B}}$ *is totally strict.*

*P*roof. We give a pseudocode algorithm based on Definition 4.4.

Algorithm 5.3. *Compute* $Sh_i^{NL}(N, v, \underline{\mathcal{B}})$

Input: *A level structure* $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, *a player* $i \in N$, *and* $v(S)$ *for all* $S \in \mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}$.

1: $\bar{v}_i^{h+1}(N) := v(N)$
2: ***for*** $k = h$ ***to*** 0 ***do*** // the descending levels
3: $\quad \tilde{v}_i^k\,(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}) := \bar{v}_i^{k+1}(\mathcal{B}^{k+1}(i))$ // the worth for the restricted grand coalition where all children of $\mathcal{B}^{k+1}(i)$ are players
4: $\quad$ ***for all*** $\mathcal{Q} \subsetneq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}$, $\mathcal{Q} \neq \emptyset$, ***do*** // all coalitions that the children of $\mathcal{B}^{k+1}(i)$ as players can form, except $\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}$
5: $\quad\quad \tilde{v}_i^k\,(\mathcal{Q}) := v(\bigcup_{B \in \mathcal{Q}} B)$
6: $\quad$ ***end for***
7: $\quad \bar{v}_i^k(\mathcal{B}^k(i)) := Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{v}_i^k)$ // calls a method/function that computes Sh before the assignment, e.g. Algorithm 5.1
8: ***end for***
9: $Sh_i^{NL}(N, v, \mathcal{B}) := \bar{v}_i^0(\{i\})$
10: ***return*** $Sh_i^{NL}(N, v, \mathcal{B})$.

Complexity: (i) Let $\underline{\mathcal{B}}$ be a strict level structure of degree d. We have

$$\begin{aligned} t(\mathit{Algorithm}\ 5.3) &= 1 + t(F_2) + 1 = \underset{Prop.5.1}{\leq} 2 + (n-1)\big[1 + t(F_4) + t(L_7)\big] \\ &\leq 1 + n + (n-1)(2^d - 2) + (n-1)t(L_7). \end{aligned}$$

By Theorem 5.5, we have $t(L_7) \in O(d2^d)$. Therefore, Algorithm 5.3 has for a strict level structure a time $O(n)$. The claim follows by running the algorithm for n players.

(ii) Let $\underline{\mathcal{B}}$ be a totally strict level structure of degree d. By Proposition 5.1, the FOR-loop, line 2, now runs at most $\log_2 n$ times instead of $(n-2)$ times. Analogous to (i), it follows

$$t(\mathit{Algorithm}\ 5.3) = 2 + \log_2 n + \log_2 n(2^d - 2) + \log_2 n \cdot t(L_7).$$

By Theorem 5.5, we have $t(L_7) \in O(d2^d)$. Therefore, Algorithm 5.3 has for a totally strict level structure a time $O(\log n)$. The claim follows by running the algorithm for n players. □

Remark 5.9. *As long as* h *is linear in* n*, Theorem 5.8 (i) remains valid and as long as* h *is logarithmic in* n*, Theorem 5.8 (ii) remains valid for arbitrary level structures of degree* d*. Again, the impact of* d *is not negligible in practice. Although, at least for small* d*, in Algorithm 5.1, the factorials could be stored directly, resulting in a slightly better runtime of* $O(2^d)$ *for* $Sh_i(D, v)$*, the influence of* d *is still exponential (see Footnote 1).*

For Sh^{NL} only the relationships of the children within the parent are relevant. Ow^{NL} also takes into account the relationships of the children of the parent to the siblings of the parent with a runtime complexity of the same order.

Theorem 5.10. *For all* $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$*, and* $\underline{\mathcal{B}}$ *of fixed degree* d*, it requires to compute* $Ow_i^{NL}(N, v, \underline{\mathcal{B}})$ *for all players* $i \in N$

(i) *a time* $O(n^2)$ *if* $\underline{\mathcal{B}}$ *is strict,*

(ii) *a time* $O(n \log n)$ *if* $\underline{\mathcal{B}}$ *is totally strict.*

Proof. We give a pseudocode algorithm based on Definition 4.9.

Algorithm 5.4. *Compute* $Ow_i^{NL}(N, v, \underline{\mathcal{B}})$

Input: *A level structure* $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, *a player* $i \in N$, *and* $v(S)$ *for all* $S \in \mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}$.

if $h = 0$ ***then***
  $Ow_i^{NL}(N, v, \mathcal{B}) := Sh_i(N, v)$ // calls a method/function that computes Sh before the assignment, e.g. Algorithm 5.1
else // $h \geq 1$
  for all $T \subseteq N$, $T = \bigcup_{B \in \mathcal{B}^h, B \subseteq T} B$ ***do*** // all coalitions that the components of the hth level can form with their own complete player sets among themselves
    $\bar{v}_i^{h+1}(T) := v(T)$
  end for
  for $k = h$ ***to*** 1 ***do*** // the descending levels
    for all $\mathcal{Q} \subseteq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}$ ***do*** // all coalitions that the children of $\mathcal{B}^{k+1}(i)$ as players can form
      $\tilde{v}_i^k(\mathcal{Q}) := \bar{v}_i^{k+1}(\bigcup_{T \in \mathcal{Q}} T)$
    end for
    $\bar{v}_i^k(\mathcal{B}^k(i)) := Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{v}_i^k)$ // calls a method/function that computes Sh before the assignment, e.g. Algorithm 5.1
    for all $S \subseteq \mathcal{B}^k(i)$, $S = \bigcup_{B \in \mathcal{B}^{k-1}, B \subseteq S} B$, ***do*** // all coalitions that the children of $\mathcal{B}^k(i)$ can form with their own complete player sets among themselves
      for all $\mathcal{Q} \subseteq \mathcal{S}^{\mathcal{B}^k(i)}$ ***do*** // all subsets from the set containing all children of $\mathcal{B}^{k+1}(i)$ where $\mathcal{B}^k(i)$ is replaced by coalition S (see Notation 4.8)
        $v_i^k(\mathcal{Q}) := v(\bigcup_{T \in \mathcal{Q}} T)$
      end for
      $\bar{v}_i^k(S) := Sh_S(\mathcal{S}^{\mathcal{B}^k(i)}, v_i^k)$ // calls a method/function that computes Sh before the assignment, e.g. Algorithm 5.1
    end for
  end for
  for all $\mathcal{Q} \subseteq \mathcal{B}^0|_{\mathcal{B}^1(i)}$ ***do*** // all coalitions that the components of the zeroth level (the singletons), restricted to $\mathcal{B}^1(i)$, as players can form
    $\tilde{v}_i^0(\mathcal{Q}) := \bar{v}_i^1(\bigcup_{T \in \mathcal{Q}} T)$
  end for
  $Ow_i^{NL}(N, v, \mathcal{B}) := \bar{v}_i^0(\{i\})$
end if
return $Ow_i^{NL}(N, v, \mathcal{B})$.

Complexity: (i) Let $\underline{\mathcal{B}}$ be a strict level structure of degree d. We have

$t(\mathit{Algorithm}\ 5.4)$

$$
\begin{aligned}
&\leq && t(IF_1) + t(ELSE_3) = 1 + t(L_2) + t(F_4) + t(F_7) + t(F_{19}) + 1 \\
&\underset{Prop.5.1}{\leq} && 2 + t(L_2) + 2^d - 1 + (n-2)\big[t(F_8) + t(L_{11}) + t(F_{12})\big] + 2^d - 1 \\
&\leq && 2^{d+1} + t(L_2) + (n-2)\Big[2^d - 1 + t(L_{11}) + (2^d - 1)\big[t(F_{13}) + t(L_{16})\big]\Big] \\
&\leq && 2^{d+1} + t(L_2) + (n-2)\Big[2^d - 1 + t(L_{11}) + (2^d - 1)\big[2^d - 1 + t(L_{16})\big]\Big].
\end{aligned}
$$

By Theorem 5.5, we have $t(L_2), t(L_{11}), t(L_{16}) \in O(d2^d)$. Therefore, Algorithm 5.4 has for a strict level structure a time $O(n)$. The claim follows by running the algorithm for n players.

(ii) Let $\underline{\mathcal{B}}$ be a totally strict level structure of degree d. By Proposition 5.1, the FOR-loop, line 7, now runs at most $(\log_2 n - 1)$ times instead of $(n-2)$ times. Analogous to (i), it follows

$$
\begin{aligned}
t(\mathit{Algorithm}\ 5.4) \leq\ & 2^{d+1} + t(L_2) \\
& + (\log_2 n - 1)\Big[2^d - 1 + t(L_{11}) + (2^d - 1)\big[2^d - 1 + t(L_{16})\big]\Big].
\end{aligned}
$$

By Theorem 5.5, we have $t(L_2), t(L_{11}), t(L_{16}) \in O(d2^d)$. Therefore, Algorithm 5.4 has for a totally strict level structure a time $O(\log n)$ and the claim follows by running the algorithm for n players. □

Remark 5.11. *Theorem 5.10 (i) remains valid for arbitrary level structures of degree d as long as h is linear in n, Theorem 5.10 (ii) remains valid for arbitrary level structures of degree d as long as h is logarithmic in n. The effect of d is now quadratic to that of d in Algorithm 5.3 (2^{2d} instead of 2^d). Therefore, in practice, the maximum degree d can now only be half as large as that used for Sh^{NL} to compute Ow^{NL} in a reasonable time.*

5.3 Relevant coalition functions

In this subsection, we will look again at the Shapley levels value. For the representation in (5), we need the dividends, whose calculation normally takes exponential time.

Theorem 5.12. *For all $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$, it requires to compute the dividends $\Delta_v(T)$ for all $T \in \Omega^N$ a time $O(3^n)$.*

Proof. For the proof, we adapt the “dividend” algorithm in Algaba et al. (2007):

Algorithm 5.5. *Compute Δ_v*

Input: $(N, v) \in \mathbb{V}^N$.

```
for ℓ = 1 to n do                         // gives the size of the coalitions
    for m = 1 to (n choose ℓ) do          // all coalitions of size ℓ
        Δ_v(T_ℓm) := v(T_ℓm) − Σ_{S⊊T_ℓm, S≠∅} Δ_v(S)     // (1)
    end for
end for
return Δ_v(T) for all T ∈ Ω^N,
```

$$\text{for } \ell = 1 \text{ to } n \text{ do}, \quad \text{for } m = 1 \text{ to } \binom{n}{\ell} \text{ do}, \quad \Delta_v(T_{\ell_m}) := v(T_{\ell_m}) - \sum_{S \subsetneq T_{\ell_m}, S \neq \emptyset} \Delta_v(S) \quad // (1)$$

where T_{ℓ_m} is the mth coalition with $|T_{\ell_m}| = \ell$.

Description: After the algorithm has computed the dividends of all singletons, the dividends of the larger coalitions are computed successively using the dividends of the smaller coalitions.

Complexity: The number of calls of line 3 by the two nested loops, line 1, line 2, is $2^n - 1$. It follows

$$t(Algorithm\ 5.5) = t(F_1) = \sum_{\ell=1}^{n} t(F_2) = \sum_{\ell=1}^{n} \sum_{m=1}^{\binom{n}{\ell}} t(L_3) = \sum_{\ell=1}^{n} \sum_{m=1}^{\binom{n}{\ell}} (2^\ell - 1)$$
$$= \sum_{\ell=1}^{n} \binom{n}{\ell} 2^\ell - (2^n - 1) = \sum_{\ell=0}^{n} \binom{n}{\ell} 2^\ell - 2^n = 3^n - 2^n,$$

where the last equal sign follows by the binomial theorem. Therefore, Algorithm 5.5 has a time $O(3^n)$. □

By Theorem 5.12, an unreflected implementation of (5) in an executable algorithm for the computation of the Shapley levels value Sh^L requires exponential time. In the following, we will propose an explicit expression for the Shapley levels value with a polynomial runtime for totally strict level structures of fixed degree. Therefore, we generalize the concept of relevant coalitions in Subsection 5.1.

Definition 5.13. *For all $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$, $\mathcal{R} \subseteq \Omega^N$, and $v^{\mathcal{R}}$ such that $v^{\mathcal{R}}(T) := v(T)$ for all $T \in \mathcal{R}$ and $\Delta_{v^{\mathcal{R}}}(S) = 0$ for all $S \in \Omega^N \backslash \mathcal{R}$, we call $v^{\mathcal{R}}$ the ($\mathcal{R}$-)**relevant coalition function** on (N, v) and all $T \in \mathcal{R}$ are called ($\mathcal{R}$-)**relevant coalitions**.*

If we know the relevant coalitions and their number is not too large, the computation of dividends for a relevant coalition function can be done efficiently.

Theorem 5.14. *For all $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$, and $\mathcal{R} \subseteq \Omega^N$ a set of relevant coalitions on (N, v), it requires to compute all dividends $\Delta_{v^{\mathcal{R}}}(T)$ a time $O(n^{2k})$ if the number of all $T \in \mathcal{R}$ is bounded by a polynomial of degree k.*

Proof. For the proof, we again adapt the "dividend" algorithm in Algaba et al. (2007).

Algorithm 5.6. *Compute $\Delta_{v^{\mathcal{R}}}$*

Input: *$v^{\mathcal{R}}(T)$ for all $T \in \mathcal{R}$.*

```
for ℓ = 1 to n do                              // gives the size of the coalitions
    for m = 1 to |R_ℓ| do                      // all coalitions from R of size ℓ
        Δ_{v^R}(T_{ℓ_m}) := v^R(T_{ℓ_m}) − Σ_{S⊊T_{ℓ_m}, S∈R} Δ_{v^R(S)}     // (1)
    end for
end for
return Δ_{v^R}(T) for all T ∈ R,
```

where $\mathcal{R}_\ell$ is the set of all coalitions from $\mathcal{R}$ of size ℓ and T_{ℓ_m} is the mth coalition from $\mathcal{R}_\ell$.

Description: As in Algorithm 5.5, first the dividends of all singletons are computed and then, successively, the dividends of larger coalitions using the dividends of the smaller ones.

Complexity: The number of summands in line 3 is bounded by a polynomial of degree k. Thus, we have $t(L_3) \in O(n^k)$. The number of all calls of line 3 by the two nested loops, line 1, line 2, is bounded by a polynomial of degree k. We have

$$t(Algorithm\ 5.6) = t(F_1) = \sum_{i=\ell}^{n} t(F_2) = \sum_{\ell=1}^{n} \sum_{m=1}^{|\mathcal{R}_\ell|} t(L_3).$$

Therefore, Algorithm 5.6 has a time $O(n^{2k})$. □

By (11), for a totally strict level structure of degree d, the number of the $\mathcal{R}^i_{\underline{\mathcal{B}}}$-relevant coalitions is bounded by a polynomial of degree $(d-1)$. That implies that computing the dividends $\Delta_{v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}}(T)$ for all $T \in \mathcal{R}^i_{\underline{\mathcal{B}}}$ requires a time $O(n^{2d-2})$. In fact, if we take advantage of the special structure of a level structure, the time $O(n^{2d-2})$ can still be improved.

Theorem 5.15. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}}$ such that $\underline{\mathcal{B}}$ is a totally strict level structure of fixed degree d and $v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}$ is the $\mathcal{R}^i_{\underline{\mathcal{B}}}$-relevant coalition function on (N, v), it requires to compute all dividends $\Delta_{v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}}$ for all $T \in \mathcal{R}^i_{\underline{\mathcal{B}}}$ a time $O\big(n^{\frac{d-1}{\log_3 2}}\big)$.*

Proof. For the proof, we look at a TU-game where all children of $\mathcal{B}^1(i)$ and all siblings of all ancestors of $\{i\}$ are the players. All coalitions which these players can form have the same worth as the corresponding previous coalitions. Thus, the dividends of these new coalitions also match the corresponding original dividends of the $\mathcal{R}^i_{\underline{\mathcal{B}}}$-relevant coalition function. Since we have, by Proposition 5.1, at most $d + (\log_2 n - 1)(d-1) = (d-1)\log_2 n + 1$ players, we need, by Theorem 5.12, a time $O(3 \cdot 3^{(d-1)\log_2 n}) = O(n^{\frac{d-1}{\log_3 2}})$ to compute all dividends. □

By Remark 5.2 and (5), the following alternative definition of the Shapley levels value Sh^L is obvious.

Remark 5.16. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $i \in N$, be $v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}$ the $\mathcal{R}^i_{\underline{\mathcal{B}}}$-relevant coalition functions on (N, v) and $K_{\underline{\mathcal{B}},T}(i)$ the expressions from (4). Then, the Shapley levels value Sh^L is given by*

$$Sh^L_i(N, v, \underline{\mathcal{B}}) = \sum_{T \in \mathcal{R}^i_{\underline{\mathcal{B}}},\, T \ni i} K_{\underline{\mathcal{B}},T}(i) \Delta_{v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}}(T) \text{ for all } i \in N.$$

Also for an algorithm, based on an explicit expression, we have a polynomial runtime for the Shapley levels value.

Theorem 5.17. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$ such that $\underline{\mathcal{B}}$ is a totally strict level structure of fixed degree d, it requires to compute $Sh^L_i(N, v, \underline{\mathcal{B}})$ for all players $i \in N$ a time $O\big(n^{(\frac{d-1}{\log_3 2}+1)}\big)$ if we use an algorithm based on (5).*

Proof. We give a pseudocode algorithm based on Remark 5.16 and thus based on (5).

Algorithm 5.7. *Compute $Sh^L_i(N, v, \underline{\mathcal{B}})$ with dividends*

Input: *A level structure $\underline{\mathcal{B}} \in \mathbb{L}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, a player $i \in N$, and $v(T)$ for all $T \in \mathcal{R}^i_{\underline{\mathcal{B}}}$.*

Compute $\Delta_{v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}}(T)$ for all $T \in \mathcal{R}^i_{\underline{\mathcal{B}}}$

$sum := 0$

for all *$T \in \mathcal{R}^i_{\mathcal{B}}$, $T \ni i$* ***do*** // the relevant coalitions for player i

4: $\quad K_{\underline{\mathcal{B}},T}(i) := 1$ // initialization
5: $\quad$ ***for*** $r = 0$ ***to*** h ***do*** // the levels
6: $\qquad K_{\underline{\mathcal{B}},T}(i) := K_{\underline{\mathcal{B}},T}(i) \cdot \dfrac{1}{|\{B \in \mathcal{B}^r : B \subseteq \mathcal{B}^{r+1}(i),\ B \cap T \neq \emptyset\}|}$ // (4)
7: $\quad$ ***end for***
8: $\quad sum := sum + K_{\underline{\mathcal{B}},T}(i)\Delta_{v^{\mathcal{R}^i_{\underline{\mathcal{B}}}}}(T)$ // (5)
9: ***end for***
10: $Sh^L_i(N, v, \underline{\mathcal{B}}) := sum$
11: ***return*** $Sh^L_i(N, v, \mathcal{B})$.

Complexity: Let $\underline{\mathcal{B}}$ be a totally strict level structure of degree d. We have, according to the proof of Theorem 5.6, $h \leq (\log_2 n) - 1$ and $|\mathcal{R}^i_{\underline{\mathcal{B}}}| \leq (2n^{d-1} - 1)$. By Theorem 5.15, it follows $t(Line\,1) \in O\big(n^{\frac{d-1}{\log_3 2}}\big)$. We have $t(L_6)$, $t(L_8) \leq c$, $c \in \mathbb{N}$, and obtain

$$\begin{aligned} t(Algorithm\ 5.7) &= t(Line\,1) + 1 + t(F_3) + 1 \\ &\leq t(Line\,1) + 2 + (2n^{d-1} - 1)\big[1 + t(F_5) + t(L_8)\big] \\ &\leq \Big[t(Line\,1) + 2 + 2n^{d-1}\big[1 + \log_2 n \cdot c + c\big]\Big] \in O\big(n^{\frac{d-1}{\log_3 2}}\big). \end{aligned}$$

The claim follows by running the algorithm for n players. $\square$

6 General reflections

In the previous section, an efficient payoff computation was possible because we did not have to consider all coalitions in the LS-values examined. The same payoffs could also be obtained if, in the related coalition functions, the relevant coalitions would receive their original worth and the other coalitions a worth that results in a dividend of zero. Since we can consider for any coalition function all essential coalitions as relevant, a simple relationship emerges.

Remark 6.1. *Let $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$. If we define $\mathcal{R}$ as the set of all essential coalitions in (N, v), we have $v = v^{\mathcal{R}}$.*

For the examined LS-values, a certain perspective on the hierarchical structure was crucial to determine which coalitions were considered being relevant. Apart from a hierarchical structure, in practice, there are often many other restrictions on the formation of coalitions: group size, spatial restrictions such as rooms, buildings, and locations, or specific requirements for certain members within a team such as military units, ship or aircraft crews, or development and programming teams. In networks (no complete graphs), we often have a direct or indirect connection within a fixed number of coalitions. Definition 5.13 allows the formation of any relevant coalition that may actually or even theoretically be formed.

Relevant coalition functions have a close connection to graph and hypergraph games in Deng and Papadimitriou (1994). A (undirected) **hypergraph** $G = (N, E)$ consists of a set N of **nodes** and a set E of non-empty subsets of N, called **hyperedges**. If the hyperedges are only 2-element subsets of nodes, we call G a **graph**. Since the set of nodes N corresponds to a player set $N \in \mathcal{N}$, we can also interpret each hyperedge $S \in E$ as a coalition $S \in \Omega^N$ of players.

Deng and Papadimitriou define for a given undirected graph $G = (N, E)$ with an integer weight $v_G(S)$ on each edge $S \in E$ a TU-game (N, v_G) by $v_G(T) := \sum_{S \subseteq T,\, S \in E} v_G(S)$ for all $T \subseteq N$. They show that for such games the Shapley value for a player $i \in N$ is to compute by $Sh_i(N, v_G) = \frac{1}{2} \sum_{S \in E,\, S \ni i} v_G(S)$, which results in time $O(n^2)$ to compute the Shapley value for the complete player set. In a first extension, the authors allow games with an underlying hypergraph with weighted hyperedges of a fixed size $k \geq 2$. The coalition function v_G is still given by $v_G(T) := \sum_{S \subseteq T,\, S \in E} v_G(S)$ for all $T \subseteq N$. Since the number of edges for a fixed k is polynomial in n, the Shapley value can be computed by $Sh_i(N, v_G) = \frac{1}{k} \sum_{S \in E,\, S \ni i} v_G(S)$ in polynomial-time. These games coincide with the k-games in van den Nouweland et al. (1996) (see also Section 3).

In the last extension, the size of the hyperedges can vary as long as the number of hyperedges is polynomial in n. This extension is mentioned only rudimentarily. Therefore, a small but for our further considerations significant lack of clarity in Deng and Papadimitriou (1994) should be pointed out. As long as we have no proper subset relationship between hyperedges, by (1), the worth of a hyperedge in v_G is equal to the Harsanyi dividend of the corresponding coalition, all other coalitions have a dividend of zero, and the worth of any coalition is equal to the sum of the worths of all hyperedges contained in that coalition. But, if a hyperedge $T \in E$ contains another hyperedge $S \subsetneq T$ with a non-zero weight as a proper subset, the worth of T cannot be the sum of the worths of S and T simultaneously. Therefore, in the following, we define the weights on each hyperedge $S \in E$ as the dividend $\Delta_{v_G}(S)$ and v_G is given by $v_G(T) := \sum_{S \in E} \Delta_{v_G}(S)$ for all $T \subseteq N$. We believe that this is what Deng and Papadimitriou had in mind.

We make a small generalization to hyperedges with arbitrary weights, allow that singletons can also be hyperedges, and the number of hyperedges no longer has to be polynomial in n. Then, we have for each TU-game (N, v_G) with $G = (N, E)$ a corresponding TU-game $(N, v^{\mathcal{R}})$ and vice versa such that $E = \mathcal{R}$, $\Delta_{v_G}(S) = \Delta_{v^{\mathcal{R}}}(S)$ for all $S \subseteq N$, and thus $v_G = v^{\mathcal{R}}$, where $\mathcal{R}$ is the set of all essential coalitions in (N, v). In particular, we have $v_G = v$ if $v = v^{\mathcal{R}}$ (see Remark 6.1). While the work of Deng and Papadimitriou was, in many respects, groundbreaking for the following literature, this relationship seems to be little or not at all known in the literature so far[10]. This correlation means that the representation in our small generalization is fully expressive, i.e., it can model any TU-game!

In Deng and Papadimitriou (1994), the coalition function is given (in our generalization) by the weights of the hyperedges and thus by the dividends of the relevant coalitions. It follows, by the same arguments as in Deng and Papadimitriou (1994) and Remark 6.1, that the Shapley value can be computed for all TU-games (N, v) in polynomial-time as long as the number of all essential coalitions in v is polynomial in n and we know the essential coalitions and their dividends. If the dividends for the essential coalitions are not explicitly given, we can compute them in advance in polynomial-time according to Theorem 5.14 using Algorithm 5.6. It follows an alternative definition for the Shapley value.

Remark 6.2. *For all $N \in \mathcal{N}$, $(N, v) \in \mathbb{V}^N$, let $\mathcal{R}_{(N,v)}$ be the set of all essential coalitions in (N, v). Then the Shapley value Sh is given by*

$$Sh_i(N, v) := \sum_{S \in \mathcal{R}_{(N,v)},\, S \ni i} \frac{\Delta_v(S)}{|S|} \quad \text{for all } i \in N.$$

It is clear from the outset which coalitions we consider as relevant for k-games and games on hypergraphs. For level structures, we have determined which coalitions are relevant by

[10] Ieong and Shoham, (2005, p. 194) and Michalak et al, (2013, p. 614/615), for example, only look at graphs, which naturally are not fully expressive.

selecting an LS-value. We do this indirectly when we select a value for TU-games. The equal surplus division value (Driessen and Funaki, 1991) is nothing else than the Shapley value, computed with the relevant coalition function where only the singletons and the grand coalition are considered as relevant. The same relationship applies between the proportional rule (Moriarity, 1975) and the proportional Shapley value (Béal et al., 2018; Besner, 2019a). That is, if in fact only the singletons and the grand coalition are essential, we can still use the axiomatizations of, say, the Shapley value to select a value, but then use the simple formula of the equal surplus division value for the calculation. A very similar relationship exists between the Shapley levels value and the other two LS-values examined.

Theorem 6.3. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, we have*

(i) $Sh_i^{NL}(N, v, \underline{\mathcal{B}}) = Sh_i^L(N, v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}}, \underline{\mathcal{B}})$ and

(ii) $Ow_i^{NL}(N, v, \underline{\mathcal{B}}) = Sh_i^L(N, v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}}, \underline{\mathcal{B}})$ for all $i \in N$,

where $\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}$ is the set of relevant nested Shapley coalitions and $\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}$ the set of relevant nested Owen coalitions for player i.

By Theorem 6.3 and Remark 5.16, we immediately obtain the following corollary.

Corollary 6.4. *For all $N \in \mathcal{N}$, $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $i \in N$, be $v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}}$ the $\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}$-relevant coalition function, $v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}}$ the $\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}$-relevant coalition function on (N, v), and $K_{\underline{\mathcal{B}},T}(i)$ the expressions from (4). Then the nested Shapley Levels value Sh^{NL} and the nested Owen levels value Ow^{NL} are given by*

$$Sh_i^{NL}(N, v, \underline{\mathcal{B}}) = \sum_{T \in \mathcal{R}_{\underline{\mathcal{B}}}^{Sh_i}, T \ni i} K_{\underline{\mathcal{B}},T}(i)\Delta_{v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Sh_1}}}(T) \text{ and}$$

$$Ow_i^{NL}(N, v, \underline{\mathcal{B}}) = \sum_{T \in \mathcal{R}_{\underline{\mathcal{B}}}^{Ow_i}, T \ni i} K_{\underline{\mathcal{B}},T}(i)\Delta_{v^{\mathcal{R}_{\underline{\mathcal{B}}}^{Ow_1}}}(T) \text{ for all } i \in N.$$

Example 6.5. We give a numerical example to compare the distributions of the examined LS-values. Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $N := \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$, and $\underline{\mathcal{B}} = \{\mathcal{B}^0, \mathcal{B}^1, \mathcal{B}^2, \mathcal{B}^3\}$, with $\mathcal{B}^1 := \{\{1, 2\}, \{3, 4\}, \{5, 6\}, \{7\}, \{8, 9\}\}$, $\mathcal{B}^2 := \{\{1, 2, 3, 4\}, \{5, 6, 7\}, \{8, 9\}\}$ (see Figure 3).

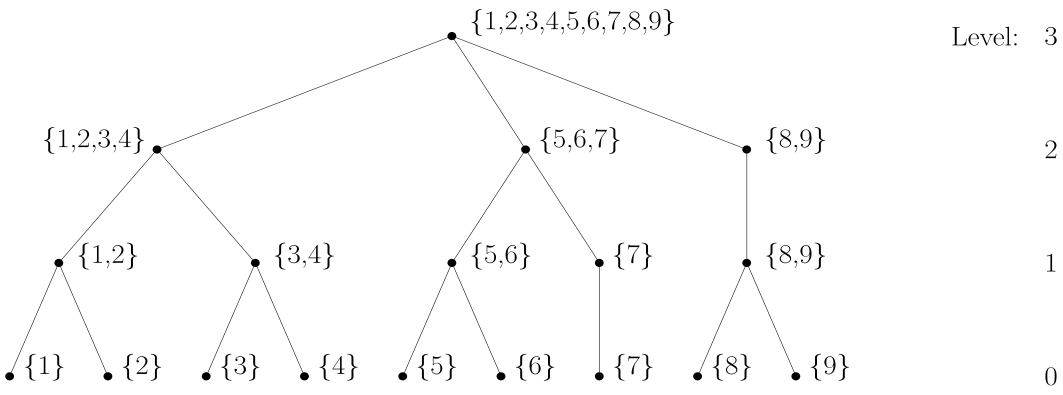

Figure 3: Structure of the components in different levels

The coalition function v is given by

$$v(\{3\}) = 1, \qquad v(\{1,2\}) = 2, \qquad v(\{1,2,3\}) = 9,$$
$$v(\{5,6,7\}) = 6, \qquad v(\{1,2,5,7\}) = 5, \qquad v(\{1,2,3,8,9\}) = 14,$$
$$v(\{5,6,7,8,9\}) = 12, \qquad v(\{1,2,3,5,6,7,8,9\}) = 40,$$

and all other coalitions are not essential in (N,v). By (1), it follows

$$\Delta_v(\{3\}) = 1, \qquad \Delta_v(\{1,2\}) = 2, \qquad \Delta_v(\{1,2,3\}) = 6,$$
$$\Delta_v(\{5,6,7\}) = 6, \qquad \Delta_v(\{1,2,5,7\}) = 2, \qquad \Delta_v(\{1,2,3,8,9\}) = 5,$$
$$\Delta_v(\{5,6,7,8,9\}) = 6, \qquad \Delta_v(\{1,2,3,5,6,7,8,9\}) = 12,$$

and all other coalitions have zero dividends in v. To compute Sh^{NL} and Ow^{NL} with the formulas in Corollary 6.4, we use the dividends of the corresponding relevant coalition functions, which we give, as an example, for player 1, and which also apply to player 2 here:

$$\Delta_{v^{\mathcal{R}^{Sh_1}_{\underline{\mathcal{B}}}}}(\{1,2\}) = 2, \quad \Delta_{v^{\mathcal{R}^{Sh_1}_{\underline{\mathcal{B}}}}}(\{1,2,3,4\}) = 6, \quad \Delta_{v^{\mathcal{R}^{Sh_1}_{\underline{\mathcal{B}}}}}(\{1,2,3,4,5,6,7\}) = 2,$$
$$\Delta_{v^{\mathcal{R}^{Sh_1}_{\underline{\mathcal{B}}}}}(\{1,2,3,4,8,9\}) = 5, \quad \Delta_{v^{\mathcal{R}^{Sh_1}_{\underline{\mathcal{B}}}}}(N) = 12,$$
$$\Delta_{v^{\mathcal{R}^{Ow_1}_{\underline{\mathcal{B}}}}}(\{1,2\}) = 2, \quad \Delta_{v^{\mathcal{R}^{Ow_1}_{\underline{\mathcal{B}}}}}(\{1,2,3,4\}) = 6, \quad \Delta_{v^{\mathcal{R}^{Ow_1}_{\underline{\mathcal{B}}}}}(\{1,2,5,6,7\}) = 2,$$
$$\Delta_{v^{\mathcal{R}^{Ow_1}_{\underline{\mathcal{B}}}}}(\{1,2,3,4,8,9\}) = 5, \quad \Delta_{v^{\mathcal{R}^{Ow_1}_{\underline{\mathcal{B}}}}}(N) = 12.$$

We obtain Table 1. Note that player 4 is a null player, and players 1 and 2 and players 8 and 9 respectively are symmetric in (N,v).

Table 1: Comparison of different values

Value	Payoff to the players (rounded)								
	1	2	3	4	5	6	7	8	9
$Sh(N,v)$	6	6	5.5	0	5.2	4.7	5.2	3.7	3.7
$Sh^L(N,v,\underline{\mathcal{B}})$	4.625	4.625	7.25	0	3.75	3.25	7	4.75	4.75
$Sh^{NL}(N,v,\underline{\mathcal{B}})$	4.375	4.375	4.375	3.375	3.5	3.5	7	4.75	4.75
$Ow^{NL}(N,v,\underline{\mathcal{B}})$	4.625	4.625	5.625	1.625	3.5	3.5	7	4.75	4.75

(End of Example 6.5)

Suppose that the number of essential coalitions is polynomially bounded and we know them and their worths or dividends. Then Remark 6.2 can serve as a blueprint for all TU-values from the Harsanyi set (Vasil'ev, 1978; Vasil'ev and van der Laan, 2002), also called selectope (Hammer et al., 1977; Derks et al., 2000), or for the TU-values from the generalized Harsanyi set (Besner, 2020), for which the coefficients of the related dividends can then be computed in polynomial-time, such as the proportional Shapley value or the proportional Harsanyi solution (Besner, 2020). The representation of the Banzhaf value (Banzhaf, 1965) in van den Brink and van der Laan (1998, Theorem 2.1) is also suitable.

We know the essential coalitions especially in games where not only the grand coalition but also other coalitions (in the same period) are actually formed. Here, the dividend of the larger coalition that is formed is only added to the dividends of already formed coalitions, which are part of this coalition. Not formed coalitions receive a zero dividend.

We are thinking, for example, of a cost function in which specific costs can be assigned to a unit or cost center (dividends), and the coalitional worth of the cost center comprises the sum of these costs and all costs of its sub-cost centers. Such situations are likely to occur often in totally positive games (Vasil'ev, 1975), i.e., games in which all coalitions have a non-negative dividend (see also the example in Besner (2020)).

7 Conclusion and discussion

In this paper, we have examined three different LS-values. Based on corresponding algorithms, we could obtain polynomial runtimes for each value, depending on the structure of the level structure. Note that our algorithms are particularly efficient for models of hierarchical structures, which in practice also require good communication between the individual members within the structure, whereby the individual groups (components) must not become too large. We would like to point out here, e.g., the agile framework Scrum (Takeuchi and Nonaka, 1986), which helps teams to work together. Each team has three to nine members. To cope with the complexity of larger projects, we can scale to Scrum of Scrums (Sutherland, 2001) or Scrum of Scrum of Scrums etc..

In principle, the results shown for the runtimes can also be transferred to weighted variants of our LS-values such as the weighted Shapley hierarchy levels values. For the nested Owen levels value, we have only considered coalitions of children of a parent with the siblings of the parent as relevant coalitions. Further extensions would be if we would allow relevant coalitions also with siblings of the ancestors on any number of levels. As long as this number of levels is logarithmic in n, we get polynomial runtimes.

All offered algorithms for LS-values can be executed for each player independently of the others, so that parallel computing can improve the runtime in practice by up to a factor n. However, the degree of the level structure remains the limiting factor. Of course, the runtimes of LS-values that coincide with a value from the Harsanyi set for a fixed level structure or use such a value in an intermediate game can also benefit from the restriction to a set of relevant coalitions.

Sparse matrices require significantly less storage space than dense ones in numerical analysis and scientific computing and can help to use more efficient algorithms. Similarly, relevant coalition functions can be regarded as advantageous for the values presented here. On the one hand, we can solve problems caused by the huge representations[11], which are completely useless in practice, and on the other hand, much shorter runtimes are available for payoff computations. Just as there are specialized computers and algorithms for sparse matrices, used especially in the field of artificial intelligence, the use of relevant coalition functions could open up new areas of research and application for cooperative game theory.

The values, in this case for the relevant coalition functions, still satisfy their axioms, such as efficiency, null player, additivity, etc., depending on the value, including perhaps the most important axiom for practice, computational ease.

Even if the number of players is not too large, the worths of the coalitions of all possible coalitions may not be known or determinable in a reasonable time, and it may not be possible to store them appropriately. Although we would use approximation methods for

[11] Even in our small Example 6.5, we were able to reduce the storage space from $2^9 - 1 = 511$ entries for the worths of feasible coalitions to only 8 entries for the dividends of essential coalitions!

TU-values, we would finally have to agree on certain coalitions or subsets of the $n!$ orderings of the players and related worths of coalitions somehow, for example, based on Monte Carlo simulation (see, e.g., Mann and Shapley (1960) and Stanojevic et al. (2010)), the normal distribution function (see, e.g., Owen (1972)), or other in some way randomized algorithms and sampling methods (see, e.g., Fatima et al. (2008), Castro et al. (2009), and van Campen et al. (2018))

Based on Theorem 5.14, new approximation methods, which still need to be developed, may offer some advantages when using dividends and relevant coalition functions. On the one hand, values for which only a definition with dividends is known or practicable, such as most values from the Harsanyi set, can then be approximated; on the other hand, we can specifically influence which coalitions are relevant. For example, all coalitions that result from the restrictions, listed after Remark 6.1, such as group size, spatial restrictions, and so on, are suitable. We can also consider relevant coalition functions, which define as relevant coalitions only those for which data already exist or for which data are available in a certain time period. The aim should be to agree on a set of relevant coalitions whose number is limited by a polynomial in n. We assume that the grand coalition N is actually forming. However, other situations are not excluded in principle but may require special treatment to receive efficient payoffs.

Even if it seems inexact to use only a certain number of coalitions for the computation, it is often better to use the important or actually forming coalitions additionally for the payoff computation than to do without them completely when applying the equal division value or the proportional rule, for example. We can interpret the value that uses the relevant coalition function as a new value that considers only the relevant coalitions as the important ones. The worths or dividends of non-relevant coalitions have not disappeared, they have just been included in the coalition worths or dividends of the coalitions which are the relevant supersets of them.

Such a superset always exists when the grand coalition is among the relevant coalitions. For example, if we compute the proportional rule, the dividends of all coalitions with at least two players are summarized in the dividend of the grand coalition if the singletons and the grand coalition are the relevant coalitions. If we compute the Shapley levels value for a player $i \in N$ with Algorithm 5.7, the dividend of a coalition $S \subseteq N$ that i forms with other players outside the parent is always included in the dividend of the coalition that consists of all children of the smallest component containing S, where each child itself contains at least one player of S.

Altogether, the algorithms and methods presented in this study should give new impulses for the practical application of methods of cooperative game theory, e.g., in supply chain management, cost allocation, resource allocation to processes in operating systems, resource allocation of virtual machines, network analysis, etc..

Acknowledgements We are grateful to Winfried Hochstättler for pointing out to us that it should be possible to use level structures to obtain polynomial runtimes for values for cooperative games and his helpful comments.

8 Appendix

8.1 Proof of Proposition 4.6

- **E** and **LG** but not **N**: Since Sh meets $\mathbf{E}^0$, it is easy to see that Sh^{NL} satisfies **E**. **LG** also follows directly from the top-down distribution mechanism and since Sh satisfies $\mathbf{E}^0$. With the help of a small example, we can see that **N** is not satisfied: Let $(N, u_S, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $N = \{1,2,3\}$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_1}$ such that $\mathcal{B}^1 := \big\{\{1,2\},\{3\}\big\}$ and u_S be the unanimity game with carrier $S := \{2,3\}$. Player 1 is a null player in u_S but we have $Sh_1^{NL}(N, u_S, \underline{\mathcal{B}}) = \frac{1}{4} \neq 0$.
- **NBC**: Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, and $B_k, B_\ell \in \mathcal{B}^r$, $0 \leq r \leq h$, such that $B_\ell \subseteq \mathcal{B}^{r+1}(B_k)$. It is well-known that Sh satisfies $\mathbf{BC}^0$ and thus for each TU-game restricted to a component of the $(k+1)$th level where the components of the kth level are the players. Therefore, by **LG**, **NBC** is satisfied for $r = h$. By induction on the size $m := h - r$, (8), and **LG**, the claim follows immediately. $\square$

8.2 Proof of Theorem 4.7

The existence part follows by Proposition 4.6. Therefore, we only have to show uniqueness.

Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, and φ and ψ be two LS-values which satisfy **E** and **NBC**. It is sufficient to show

$$\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B} \psi_i(N, v, \underline{\mathcal{B}}) \text{ for all } B \in \mathcal{B}^r \text{ and all } r,\ 0 \leq r \leq h+1. \tag{13}$$

If $r = h+1$, (13) is satisfied by **E**. We use a first induction I_1 on the size m, $0 \leq m \leq h$, for all levels r, $0 \leq r \leq h$, where $m := h - r$.

Induction basis I_1: Let $m = 0$ and thus $r = h$.

If $|\{B : B \in \mathcal{B}^h\}| = 1$, (13) is satisfied by **E**. We use a second induction I_2 on the size $t := |\{B : B \in \mathcal{B}^h\}|, t \geq 2$.

Induction basis I_2: Let $t = 2$. We have exactly two components $B_k, B_\ell \in \mathcal{B}^h$. By **E**, it follows

$$\sum_{i \in B_k} \varphi_i\big(N \backslash B_\ell, v, \underline{\mathcal{B}_h}|_{N \backslash B_\ell}\big) = \sum_{i \in B_k} \psi_i\big(N \backslash B_\ell, v, \underline{\mathcal{B}_h}|_{N \backslash B_\ell}\big)$$

and

$$\sum_{i \in B_\ell} \varphi_i\big(N \backslash B_k, v, \underline{\mathcal{B}_h}|_{N \backslash B_k}\big) = \sum_{i \in B_\ell} \psi_i\big(N \backslash B_k, v, \underline{\mathcal{B}_h}|_{N \backslash B_k}\big).$$

We obtain, by **NBC**,

$$\sum_{i \in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_k} \psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B_\ell} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_\ell} \psi_i(N, v, \underline{\mathcal{B}}).$$

It follows

$$2 \cdot \Big[\sum_{i \in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_k} \psi_i(N, v, \underline{\mathcal{B}})\Big] = \sum_{i \in N} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in N} \psi_i(N, v, \underline{\mathcal{B}}) \underset{\mathbf{E}}{=} 0$$

and therefore, (13) is satisfied.

Induction step I_2: Assume that (13) holds for $t' \geq 2$ and all t'', $1 \leq t'' < t'$, (IH_2). Let

$t := t' + 1$. We choose two different components $B_k, B_\ell \in \mathcal{B}^h$. It follows

$$\sum_{i\in B_k} \varphi_i\big(N\backslash B_\ell, v, \underline{\mathcal{B}_h}|_{N\backslash B_\ell}\big) \underset{(IH_2)}{=} \sum_{i\in B_k} \psi_i\big(N\backslash B_\ell, v, \underline{\mathcal{B}_h}|_{N\backslash B_\ell}\big)$$

and

$$\sum_{i\in B_\ell} \varphi_i\big(N\backslash B_k, v, \underline{\mathcal{B}_h}|_{N\backslash B_k}\big) \underset{(IH_2)}{=} \sum_{i\in B_\ell} \psi_i\big(N\backslash B_k, v, \underline{\mathcal{B}_h}|_{N\backslash B_k}\big)$$

We obtain, by **NBC**,

$$\sum_{i\in B_k}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_k}\psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i\in B_\ell}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_\ell}\psi_i(N, v, \underline{\mathcal{B}}). \tag{14}$$

(14) holds for all $B \in \mathcal{B}^h$. It follows for an arbitrary $B \in \mathcal{B}^h$,

$$t \cdot \Big[\sum_{i\in B}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B}\psi_i(N, v, \underline{\mathcal{B}})\Big] = \sum_{i\in N} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in N} \psi_i(N, v, \underline{\mathcal{B}}) \underset{\mathbf{E}}{=} 0.$$

Therefore, (13) is satisfied. Note, since N and h were arbitrary, we have also shown, for all $0 \le r \le h$ and two siblings $B_k, B_\ell \in \mathcal{B}^r$,

$$\sum_{i\in B_k}\varphi_i\big(\mathcal{B}^{r+1}(B_k)\backslash B_\ell, v, \underline{\mathcal{B}_r}|_{\mathcal{B}^{r+1}(B_k)\backslash B_\ell}\big) = \sum_{i\in B_k}\phi_i\big(\mathcal{B}^{r+1}(B_k)\backslash B_\ell, v, \underline{\mathcal{B}_r}|_{\mathcal{B}^{r+1}(B_k)\backslash B_\ell}\big). \tag{15}$$

Induction step I_1: Assume that (13) holds for m', $0 \le m' < h$, and all m'', $0 \le m'' < m'$, (IH_1). Let $m = m' + 1$, $r = h - m' - 1$, $B^{r+1} \in \mathcal{B}^{r+1}$, and $t := |\{B \in \mathcal{B}^r : B \subseteq B^{r+1}\}|$. If $t = 1$, we have only one $B \in \mathcal{B}^r$, $B \subseteq B^{r+1}$. It follows

$$\sum_{i\in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i\in B^{r+1}} \varphi_i(N, v, \underline{\mathcal{B}}) \underset{(IH_1)}{=} \sum_{i\in B^{r+1}} \psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i\in B} \psi_i(N, v, \underline{\mathcal{B}}).$$

Let now $t \ge 2$. We choose two siblings $B_k, B_\ell \in \mathcal{B}^r$. We have

$$\sum_{i\in B_k} \varphi_i\big(B^{r+1}\backslash B_\ell, v, \underline{\mathcal{B}_r}|_{B^{r+1}\backslash B_\ell}\big) \underset{(15)}{=} \sum_{i\in B_k} \psi_i\big(B^{r+1}\backslash B_\ell, v, \underline{\mathcal{B}_r}|_{B^{r+1}\backslash B_\ell}\big)$$

and

$$\sum_{i\in B_\ell} \varphi_i\big(B^{r+1}\backslash B_k, v, \underline{\mathcal{B}_r}|_{B^{r+1}\backslash B_k}\big) \underset{(15)}{=} \sum_{i\in B_\ell} \psi_i\big(B^{r+1}\backslash B_k, v, \underline{\mathcal{B}_r}|_{B^{s+1}\backslash B_k}\big).$$

By **NBC**, we obtain

$$\sum_{i\in B_k}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_k}\psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i\in B_\ell}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B_\ell}\psi_i(N, v, \underline{\mathcal{B}}). \tag{16}$$

(16) holds for all $B \in \mathcal{B}^r$, $B \subseteq \mathcal{B}^{r+1}$. It follows for an arbitrary $B \in \mathcal{B}^r$, $B \subseteq B^{r+1}$,

$$t \cdot \Big[\sum_{i\in B}\varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B}\psi_i(N, v, \underline{\mathcal{B}})\Big] = \sum_{i\in B^{r+1}} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i\in B^{r+1}} \psi_i(N, v, \underline{\mathcal{B}}) \underset{(IH_1)}{=} 0.$$

Thus, we have $\sum_{i\in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i\in B} \psi_i(N, v, \underline{\mathcal{B}})$ for all $B \in \mathcal{B}^r$, $B \subseteq B^{r+1}$, which proves the claim. □

8.3 Proof of Proposition 4.12

- **E** and **LG** but not **N**: Since Sh meets $\mathbf{E}^0$ and Ow meets **E**, it is easy to see that Ow^{NL} satisfies **E**. **LG** also follows directly from the top-down distribution mechanism and since Sh meet $\mathbf{E}^0$ and Ow meet **E**. The following example shows that **N** is not satisfied.

Let $(N, u_S, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} := \underline{\mathcal{B}_2}$, $N = \{1,2,3,4\}$, with $\mathcal{B}^1 := \{\{1,2\},\{3\},\{4\}\}, \mathcal{B}^2 := \{\{1,2,3\},\{4\}\}$, and be u_S the unanimity game with carrier $S := \{2,3,4\}$. Player 1 is a null player in u_S but we have $Ow_1^{NL}(N, u_S, \underline{\mathcal{B}}) = \frac{1}{8} \neq 0$.

- **NBOC**: Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $B_k, B_\ell \in \mathcal{B}^r$, $0 \leq r \leq h$, such that $B_\ell \subseteq \mathcal{B}^{r+1}(B_k)$. If $r = h$, (10) is satisfied by **LG** and since Sh meets $\mathbf{BC}^0$. Let now $r < h$. By (9), we have
$$Ow_{B_k}^{NL}(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}) - Ow_{B_k}\big(\mathcal{B}^r|_{\mathcal{B}^{r+2}(B_k)}, v^r, \underline{\mathcal{B}^r_{r+1}}|_{\mathcal{B}^{r+2}(B_k)}\big)$$
$$= Ow_{B_\ell}^{NL}(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}) - Ow_{B_\ell}\big(\mathcal{B}^r|_{\mathcal{B}^{r+2}(B_\ell)}, v^r, \underline{\mathcal{B}^r_{r+1}}|_{\mathcal{B}^{r+2}(B_\ell)}\big).$$
Since Ow, as a special case of the Shapley levels value, satisfies **BC**, it follows
$$Ow_{B_k}^{NL}(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}) - Ow_{B_k}\big(\mathcal{B}^r|_{\mathcal{B}^{r+2}(B_k)\backslash B_\ell}, v^r, \underline{\mathcal{B}^r_{r+1}}|_{\mathcal{B}^{r+2}(B_k)\backslash B_\ell}\big)$$
$$= Ow_{B_\ell}^{NL}(\mathcal{B}^r, v^r, \underline{\mathcal{B}^r}) - Ow_{B_\ell}\big(\mathcal{B}^r|_{\mathcal{B}^{r+2}(B_\ell)\backslash B_k}, v^r, \underline{\mathcal{B}^r_{r+1}}|_{\mathcal{B}^{r+2}(B_\ell)\backslash B_k}\big).$$
By Remark 4.11 and **LG**, the claim follows immediately. □

8.4 Proof of Theorem 4.13

The existence part follows by Proposition 4.12. Therefore, we only have to show uniqueness. Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, and φ and ψ be two LS-values which satisfy **E** and **NBOC**. It is sufficient to show
$$\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B} \psi_i(N, v, \underline{\mathcal{B}}) \text{ for all } B \in \mathcal{B}^r \text{ and all } 0 \leq r \leq h+1. \tag{17}$$
We use a first induction I_1 on the levels starting with level $h+1$.

Induction basis I_1: Let $r = h+1$. Then (17) is satisfied by **E**.

Induction step I_1: Assume that (17) is satisfied for all r, $0 \leq s < r \leq h+1$, (IH_1). Let $B^{s+1} \in \mathcal{B}^{s+1}$. We use a second induction I_2 on the size $t := |\{B \in \mathcal{B}^s : B \subseteq B^{s+1}\}|$.

Induction basis I_2: Let $t = 1$. We have only one $B \in \mathcal{B}^s$, $B \subseteq B^{s+1}$, and, by **E**, it follows
$$\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B^{s+1}} \varphi_i(N, v, \underline{\mathcal{B}}) \underset{(IH_1)}{=} \sum_{i \in B^{s+1}} \psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B} \psi_i(N, v, \underline{\mathcal{B}}). \tag{18}$$
Note that (18) holds also for all restricted cuts as in (10) with $r = s$.

Induction step I_2: Assume that (17) holds for $t' \geq 1$ and all $1 \leq t'' < t'$, (IH_2). Let $t := t' + 1$. We choose two siblings $B_k, B_\ell \in \mathcal{B}^s$, $B_k, B_\ell \subseteq B^{s+1}$. It follows
$$\sum_{i \in B_k} \varphi_i\big(\mathcal{B}^{s+2}(i)\backslash B_\ell, v, \underline{\mathcal{B}_{s+1}}|_{\mathcal{B}^{s+2}(i)\backslash B_\ell}\big) \underset{(IH_2)}{=} \sum_{i \in B_k} \psi_i\big(\mathcal{B}^{s+2}(i)\backslash B_\ell, v, \underline{\mathcal{B}_{s+1}}|_{\mathcal{B}^{s+2}(i)\backslash B_\ell}\big)$$
and
$$\sum_{i \in B_\ell} \varphi_i\big(\mathcal{B}^{s+2}(i)\backslash B_k, v, \underline{\mathcal{B}_{s+1}}|_{\mathcal{B}^{s+2}(i)\backslash B_k}\big) \underset{(IH_2)}{=} \sum_{i \in B_\ell} \psi_i\big(\mathcal{B}^{s+2}(i)\backslash B_k, v, \underline{\mathcal{B}_{s+1}}|_{\mathcal{B}^{s+2}(i)\backslash B_k}\big).$$
We obtain, by **NBOC**,
$$\sum_{i \in B_k} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_k} \psi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B_\ell} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B_\ell} \psi_i(N, v, \underline{\mathcal{B}}). \tag{19}$$
(19) holds for all $B \in \mathcal{B}^s$, $B \subseteq \mathcal{B}^{s+1}$. It follows for an arbitrary $B \in \mathcal{B}^s$, $B \subseteq B^{s+1}$,
$$t \cdot \Big[\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B} \psi_i(N, v, \underline{\mathcal{B}})\Big] = \sum_{i \in B^{s+1}} \varphi_i(N, v, \underline{\mathcal{B}}) - \sum_{i \in B^{s+1}} \psi_i(N, v, \underline{\mathcal{B}}) \underset{(IH_1)}{=} 0.$$
Thus, we have $\sum_{i \in B} \varphi_i(N, v, \underline{\mathcal{B}}) = \sum_{i \in B} \psi_i(N, v, \underline{\mathcal{B}})$ for all $B \in \mathcal{B}^s$, $B \subseteq B^{s+1}$, which proves

the claim. □

8.5 Proof of Theorem 6.3

Let $(N, v, \underline{\mathcal{B}}) \in \mathbb{VL}^N$, $\underline{\mathcal{B}} = \underline{\mathcal{B}_h}$, $u := v^{\mathcal{R}^{Sh_i}_{\underline{\mathcal{B}}}}$, $w := v^{\mathcal{R}^{Ow_i}_{\underline{\mathcal{B}}}}$, and, for $0 \leq k \leq h$, be $\mathcal{T}_i^k$ the set from Notation 4.1 and $\mathcal{S}^{\mathcal{B}^k(i)}$ the set from Notation 4.8.

(i) By Definition 4.2, we have for $Sh_i^L(N, u, \underline{\mathcal{B}})$ and all k, $0 \leq k \leq h$, $\bar{u}_i^k(\mathcal{B}^k(i)) = Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{u}_i^k)$, where $\tilde{u}_i^k$ is given by

$$\tilde{u}_i^k(\mathcal{Q}) = \begin{cases} \bar{u}_i^{k+1}(\mathcal{B}^{k+1}(i)), \text{ if } \mathcal{Q} = \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \\ u(\bigcup_{B \in \mathcal{Q}} B) \text{ if } \mathcal{Q} \subsetneq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)} \text{ by } \mathbf{D}^0. \end{cases}$$

Therefore, the claim follows by Remark 5.3 and (6) and (7) in Definition 4.4.

(ii) We denote $\bar{w}_i^k$ and $\tilde{w}_i^k$ in Definition 4.9 by $\bar{y}_i^k$ and $\tilde{y}_i^k$ respectively to distinguish them from $\bar{w}_i^k$ and $\tilde{w}_i^k$ in Definition 4.2. By Remark 5.4, it is sufficient to show $\bar{w}_i^k(\mathcal{B}^k(i))$ in Definition 4.2 equals $\bar{y}_i^k(\mathcal{B}^k(i))$ in Definition 4.9 for all k, $0 \leq k \leq h$.

For this, we use induction on the size k, $h \geq k \geq 0$, and show additionally for a $c_i^k \in \mathbb{R}$ that

$$\bar{w}_i^k(S) = \begin{cases} Sh_S(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{w}_i^k), \text{ if } S = \mathcal{B}^k(i), \\ Sh_S(\mathcal{S}^{\mathcal{B}^k(i)}, w_i^k)) + c_i^k, \text{ if } S \subsetneq \mathcal{B}^k(i),\, S = \bigcup_{B \in \mathcal{B}^{k-1}, B \subseteq S} B, \\ w(S) + c_i^k, \text{ if } S \subsetneq \mathcal{B}^k(i),\, S \cap \mathcal{B}^{k-1}(i) \neq \emptyset,\, S \neq \mathcal{B}^{k-1}(i), \end{cases} \tag{20}$$

where w_i^k is given by $w_i^k(\mathcal{Q}) = w(\bigcup_{S \in \mathcal{Q}} S)$ for all $\mathcal{Q} \subseteq \mathcal{S}^{\mathcal{B}^k(i)}$, and $\tilde{w}_i^k$ by $\tilde{w}_i^k(\mathcal{Q}) = \bar{w}_i^{k+1}(\bigcup_{S \in \mathcal{Q}} S)$ for all $\mathcal{Q} \subseteq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}$.

Induction basis: Let $k = h$. By Definition 4.2 and for $c_i^k := 0$, (20) is satisfied for $Sh_i^L(N, w, \underline{\mathcal{B}})$ since Sh satisfies $\mathbf{D}^0$. Due to $\tilde{y}_i^h = \tilde{w}_i^h$, the claim is satisfied for $k = h$.

Induction step: Assume that (20) and the claim are satisfied for $k', 1 \leq k' \leq h$, (IH). Let $k := k' - 1$. By Definition 4.2, (20), and (IH), we have $\bar{w}_i^k(\mathcal{B}^k(i)) = Sh_{\mathcal{B}^k(i)}(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \tilde{w}_i^k)$, where $\tilde{w}_i^k(\mathcal{Q})$ is given by

$$\tilde{w}_i^k(\mathcal{Q}) = \begin{cases} \bar{w}_i^{k+1}(\mathcal{B}^{k+1}(i)) = \bar{y}_i^{k+1}(\mathcal{B}^{k+1}(i)) \text{ if } \mathcal{Q} = \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \\ \bar{y}_i^{k+1}(\bigcup_{S \in \mathcal{Q}} S) + c^{k+1} \text{ if } \mathcal{Q} \subsetneq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}. \end{cases}$$

We define a game $(\mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \mathring{w}_i^k)$ by

$$\mathring{w}_i^k(\mathcal{Q}) = \begin{cases} 0 \text{ if } \mathcal{Q} = \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \\ c^{k+1} \text{ if } \mathcal{Q} \subsetneq \mathcal{B}^k|_{\mathcal{B}^{k+1}(i)}, \end{cases}$$

In this game, all players are symmetric. Since Sh satisfies $\mathbf{S}^0$ and $\mathbf{E}^0$, each player gets a payoff of zero in this game with Sh. We have $\tilde{w}_i^k = \tilde{y}_i^k + \mathring{w}_i^k$. By Definition 4.9 and since Sh satisfies $\mathbf{A}^0$, it follows $\bar{y}_i^k(\mathcal{B}^k(i)) = \bar{w}_i^k(\mathcal{B}^k(i))$. By (IH), we have

$$\bar{w}_i^{k+1}(T) = \begin{cases} Sh_T(\mathcal{T}_i^{k+1}, w_i^{k+1}) + c_i^{k+1}, \text{ if } T \subsetneq \mathcal{B}^{k+1}(i),\, T = \bigcup_{B \in \mathcal{B}^k, B \subseteq T} B, \\ w(T) + c_i^k, \text{ if } T \subsetneq \mathcal{B}^{k+1}(i),\, T \cap \mathcal{B}^k(i) \neq \emptyset,\, T \neq \mathcal{B}^k(i). \end{cases}$$

We define $\breve{w}_i^{k+1}$ by

$$\breve{w}_i^{k+1}(T) := \begin{cases} \bar{w}_i^{k+1}(T) - w(T), \text{ if } T \subsetneq \mathcal{B}^{k+1}(i),\, T = \bigcup_{B \in \mathcal{B}^k, B \subseteq T} B, \\ c^{k+1}, \text{ if } T \subsetneq \mathcal{B}^{k+1}(i),\, T \cap \mathcal{B}^k(i) \neq \emptyset,\, T \neq \mathcal{B}^k(i). \end{cases}$$

It follows, $\bar{w}_i^{k+1} = w + \breve{w}_i^{k+1}$. Therefore, by Definition 4.2 and, since Sh satisfies $\mathbf{A}^0$, we have $\bar{w}_i^k(S) = Sh_S(\mathcal{S}^{\mathcal{B}^k(i)}, w_i^k)) + c_i^k$, if $S \subsetneq \mathcal{B}^k(i)$, $S = \bigcup_{B \in \mathcal{B}^{k-1}, B \subseteq S} B$, and, since Sh satisfies

$\mathbf{D}^0$, $\bar{w}_i^k(S) = w(S) + c_i^k$, if $S \subsetneq \mathcal{B}^k(i)$, $S \cap \mathcal{B}^{k-1}(i) \neq \emptyset$, $S \neq \mathcal{B}^{k-1}(i)$. The claim follows by induction. $\square$

References

Algaba, E., Bilbao, J., Fernández, J., Jiménez, N., & López, J. (2007). Algorithms for computing the myerson value by dividends. *Discrete Mathematics Research Progress,* 1–13.

Álvarez-Mozos, M., & Tejada, O. (2011). Parallel characterizations of a generalized Shapley value and a generalized Banzhaf value for cooperative games with level structure of cooperation. *Decision Support Systems, 52*(1), 21–27.

Álvarez-Mozos, M., van den Brink, R., van der Laan, G., & Tejada, O. (2017). From hierarchies to levels: new solutions for games with hierarchical structure. *International Journal of Game Theory,* 1–25.

Aumann, R.J., Drèze, J., 1974. Cooperative games with coalition structures. *International Journal of Game Theory 3,* 217–237.

Banzhaf, J.F. (1965). Weighted voting does not work: a mathematical analysis. *Rutgers Law Review 19*, 317–343.

Béal, S., Ferrières, S., Rémila, E., & Solal, P. (2018) The proportional Shapley value and applications. *Games and Economic Behavior 108*, 93–112.

Besner, M. (2019a). Axiomatizations of the proportional Shapley value. *Theory and Decision, 86*(2), 161–183.

Besner, M. (2019b). Weighted Shapley hierarchy levels values. *Operations Research Letters 47,* 122–126.

Besner, M. (2020). Value dividends, the Harsanyi set and extensions, and the proportional Harsanyi solution. *International Journal of Game Theory,* 1–23.

Brightwell, G., & Winkler, P. (1991, January). Counting linear extensions is #P-complete. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing* (pp. 175–181).

Van den Brink, R., & van der Laan, G. (1998). Axiomatizations of the normalized Banzhaf value and the Shapley value. *Social Choice and Welfare, 15*(4), 567–582.

Calvo, E., Lasaga, J. J., & Winter, E. (1996). The principle of balanced contributions and hierarchies of cooperation, *Mathematical Social Sciences, 31*(3), 171–182.

Van Campen, T., Hamers, H., Husslage, B., & Lindelauf, R. (2018). A new approximation method for the Shapley value applied to the WTC 9/11 terrorist attack. *Social Network Analysis and Mining, 8*(3).

Castro, J., Gómez, D., & Tejada, J. (2009). Polynomial calculation of the Shapley value based on sampling. *Computers & Operations Research, 36*(5),1726–1730.

Chantreuil, F. (2001). Axiomatics of level structure values. In: *Holler M.J., Owen G. (eds) Power Indices and Coalition Formation (pp. 45–62).* Springer, Boston, MA.

Deng, X., & Papadimitriou, C. H. (1994) On the Complexity of Cooperative Solution Concepts, *Mathematics of Operations Research, 19*(2), 257–266.

Derks, J., Haller, H., & Peters, H. (2000). The selectope for cooperative games. *International Journal of Game Theory, 29*(1), 23–38.

Driessen, T. S. H., & Funaki, Y. (1991). Coincidence of and collinearity between game theoretic solutions. *Operations-Research-Spektrum, 13*(1), 15–30.

Faigle, U., & Kern, W. (1992). The Shapley value for cooperative games under precedence constraints. (3), 249-266. *International Journal of Game Theory, 21*(3), 249–266.

Fatima, S. S., Wooldridge, M., & Jennings, N. R. (2008). A linear approximation method for the Shapley value. *Artificial Intelligence, 172*(14), 1673–1699.

Futia, C. (1977). The complexity of economic decision rules. *Journal of Mathematical Economics, 4*(3), 289–299.

Gómez-Rúa, M., & Vidal-Puga, J. (2011). Balanced per capita contributions and level structure of cooperation. *Top, 19*(1), 167–176.

Hammer, P. L., Peled, U. N., & Sorensen, S. (1977). Pseudo-boolean functions and game theory. *I. Core elements and Shapley value. Cahiers du CERO, 19,* 159–176.

Harsanyi, J. C. (1959). A bargaining model for cooperative n-person games. In: *A. W. Tucker & R. D. Luce (Eds.), Contributions to the theory of games IV* (325–355). Princeton NJ: Princeton University Press.

Ieong, S., & Shoham, Y. (2005). Marginal contribution nets: a compact representation scheme for coalitional games. In *Proceedings of the 6th ACM conference on Electronic commerce*, pp. 193–202.

Kalai, E., & Stanford, W. (1988). Finite rationality and interpersonal complexity in repeated games. *Econometrica: Journal of the Econometric Society,* 397–410.

Kamijo, Y. (2009). A two-step Shapley value for cooperative games with coalition structures. *International Game Theory Review, 11*(02), 207–214.

Kamijo, Y. (2013). The collective value: a new solution for games with coalition structures. *Top, 21*(3), 572–589.

Khachiyan, L. G. (1979). A polynomial algorithm in linear programming. *Doklady Akademii Nauk SSSR 244*, 1093–1096 (translated in Soviet Mathematics Doklady 20, 191–194, 1979).

Knuth, D. E. (1976). Big omicron and big omega and big theta. *ACM Sigact News, 8*(2), 18–24.

Littlechild, S. C., & Owen, G. (1973). A simple expression for the Shapley value in a special case. *Management Science, 20*(3), 370–372.

Maafa, K., Nourine, L., & Radjef, M. S. (2018). Algorithms for computing the Shapley value of cooperative games on lattices. *Discrete Applied Mathematics, 249*, 91–105.

Mann, I., & Shapley, L. S. (1960) Values of large games. IV: Evaluating the Electoral College by Monte Carlo Techniques. *The RAND Corporation, Memorandum RM-2651*

Michalak, T. P., Aadithya, K. V., Szczepanski, P. L., Ravindran, B., & Jennings, N. R. (2013). Efficient computation of the Shapley value for game-theoretic network centrality. *Journal of Artificial Intelligence Research, 46*, 607–650.

Moriarity, S. (1975). Another approach to allocating joint costs. *International The Accounting Review, 50*(4), 791–795.

Myerson, R. B. (1980). Conference structures and fair allocation rules. *International Journal of Game Theory, 9*(3), 169–182.

Van Den Nouweland, A., Borm, P., van Golstein Brouwers, W., Groot Bruinderink, R., & Tijs, S. (1996). A game theoretic approach to problems in telecommunication. *Management Science, 42*(2), 294–303.

Owen, G. (1972). Multilinear extensions of games. *Management Science, 18*(5–part–2), 64–79.

Owen, G. (1977). Values of games with a priori unions. In *Essays in Mathematical Economics and Game Theory, Springer, Berlin Heidelberg,* 76–88.

Rubinstein, A. (1986). Finite automata play the repeated prisoner's dilemma. *Journal of economic theory, 39*(1), 83–96.

Sánchez-Sánchez, F., & Vargas-Valencia, M. (2018) Games with nested constraints given by a level structure. *Journal of Dynamics & Games, 5*(2), 95.

Sastre, M., & Trannoy, A. (2002). Shapley inequality decomposition by factor components: Some methodological issues. *Journal of Economics, 9*, 51–89.

Shapley, L. S. (1953a). Additive and non-additive set functions. *Princeton University.*

Shapley, L. S. (1953b). A value for n-person games. H. W. Kuhn/A. W. Tucker (eds.), *Contributions to the Theory of Games,* Vol. 2, Princeton University Press, Princeton, 307–317.

Simon, H. A. (1972). Theories of bounded rationality. *Decision and organization, 1*(1), 161–176.

Stanojevic, R., Laoutaris, N., & Rodriguez, P. (2010) On economic heavy hitters: Shapley value analysis of 95th-percentile pricing. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement,* 75–80.

Sutherland, J. (2001). Inventing and Reinventing SCRUM in five Companies. *Cutter IT journal, 14,* 5–11.

Takeuchi, H., & Nonaka, I. (1986). The new new product development game. *Harvard business review, 64*(1), 137–146.

Vasil'ev, V. A. (1975). The Shapley value for cooperative games of bounded polynomial variation. *Optimizacija Vyp, 17,* 5–27.

Vasil'ev, V. A. (1978). Support function of the core of a convex game. *Optimizacija Vyp, 21,* 30–35.

Vasil'ev, V., & van der Laan, G. (2002). The Harsanyi set for cooperative TU-games. *Siberian Advances in Mathematics 12*, 97–125.

Winter, E. (1989). A value for cooperative games with levels structure of cooperation. *International Journal of Game Theory, 18*(2), 227–240.

Zlotkin, G., & Rosenschein, J. S. (1994). Coalition, cryptography, and stability: Mechanisms for coalition formation in task oriented domains. In *Proceedings of the twelfth national conference on artificial intelligence, vol 1, AAAI '94, Menlo Park, CA, USA. American association for artificial intelligence*, (pp. 432–437)