Complexity of Computing the Shapley Value in Games with Externalities

Oskar Skibski University of Warsaw, Poland

September 5, 2019

Abstract

We study the complexity of computing the Shapley value in games with externalities. We focus on two representations based on marginal contribution nets (embedded MC-nets and weighted MC-nets). Our results show that while weighted MC-nets are more concise than embedded MC-nets, they have slightly worse computational properties when it comes to computing the Shapley value.

1 Introduction

Coalitional games are a standard model of cooperation in multi-agent systems (Chalkiadakis et al., 2011). In the classic form the profit of a coalition is assumed to be independent of the coalitions formed by the other players. However, this simplifying assumption does not hold in many settings. For example, if agents in a system have conflicted goals or limited resources, then coalitions naturally affect each other (Dunne, 2005). There are also examples of externalities in economics, in oligopolistic markets in particular, where cooperation of some companies affect the profits of the competitors (Yi, 2003). That is why, in the last decade *coalitional games with externalities* have gain attention both in economic (Kóczy, 2018; Abe & Funaki, 2017) and AI literature (Rahwan et al., 2012; Michalak et al., 2010).

Externalities present new challenges both conceptual and computational. On the conceptual side, it is unclear how to extend most solution concepts to games with externalities. In particular, there are several non-equivalent well-established methods of extending the Shapley value to games with externalities proposed by De Clippel & Serrano (2008) (EF-value), McQuillin (2009) (MQ-value), Hu & Yang (2010) (HY-value), Feldman (1996) (SS-value) and Myerson (1977b) (MY-value). On the computational side, the existence of externalities significantly increases the size of the game itself.

To cope with the extensive space requirement of games with externalities, three different representations were proposed in the literature. The first two, called *embedded MC-nets* (Michalak et al., 2010) and *weighted MC-nets* (Michalak et al.,

	Embedded MC-nets	Weighted MC-nets				
MQ-value	in P (*)	in P (Th.6)				
EF-value	in P (Th.8)	#P-complete (Th.9)				
HY-value	#P-complete (Th.10)	#P-complete (Th.10)				
SS-value	#P-complete (Th.13)	#P-complete (Th.13)				
MY-value	#P-complete (Th.14)	#P-complete (Th.14)				

Table 1: Summary of complexity results for computing extended Shapley value in games represented as embedded and weighted MC-nets. (*) Proved by Michalak et al. (2010).

2010), are extensions of the well-known logic-based representation: marginal contribution nets (Ieong & Shoham, 2005). For the former representation, the authors proved that one extension of the Shapley value can be computed in polynomial time. For the latter one, only partial results have been proposed for three extensions. More recently, Skibski et al. (2015) proposed a new representation, partition decision trees, and proved that all five extensions of the Shapley value listed above can be computed in polynomial time under this representation.

In this paper, we fill the gap in the literature by determining what is the complexity of computing all the five extensions of the Shapley value in games represented as embedded and weighted MC-nets. Specifically, we show that only two out of five extensions can be computed in polynomial time for embedded MC-nets and only one can be computed in polynomial time for weighted MC-nets (unless P = NP). For all other values we show that computation is #P-complete (see Table 1).

Interestingly, our results are strongly based on graph theory techniques. Specifically, we show that every embedded/weighted MC-nets rule can be represented as (one or more) graphs and that each extended Shapley value can be expressed as the weighted sum over all proper vertex coloring in these graphs. Building upon these general results, for each value we analyze the resulting weighted sum. That is, we show that MQ-value is a weighted sum over 2-colorings and EF-value—over independent sets in a part of the graph. In turn, SS-value under some assumptions is proved to be equal to the number of matchings in a bipartite graph.

2 Preliminaries

In this section, we introduce basic notation and definitions.

2.1 Coalitional games with externalities

Let $N = \{1, ..., n\}$ be a set of n players, which will be fixed throughout the paper. A *coalition* is any nonempty subset of N. The set of all possible partition of N is denoted by \mathcal{P} and the set of all embedded coalitions, i.e., coalitions in partitions, by EC: $EC = \{(S, P) : P \in \mathcal{P}, S \in P\}$.

In this paper, by *game* we mean a coalitional game with externalities in a partition function form: formally, for a fixed set of players, a game, is a function that assigns a real value to every embedded coalition: $q: EC \to \mathbb{R}$. We say that a game has no externalities if the value of every coalition does not depend on the partition, i.e., g(S, P) = g(S, P') for every coalition $S \subseteq N$ and $(S, P), (S, P') \in EC.$

A value of a player in a game is a real number that represents player's importance or expected outcome. The Shapley value (Shapley, 1953) is defined for games without externalities. Assume g is such a game, i.e., there exists $\hat{g}: 2^N \to \mathbb{R}$ such that $g(S, P) = \hat{g}(S)$ for every $(S, P) \in EC$ and $\hat{g}(\emptyset) = 0$. Now, the Shapley value is defined as follows:

$$SV_i(\hat{g}) = \sum_{S \subseteq N} \zeta_S^i \cdot \hat{g}(S),$$

where:

$$\zeta_S^i = \begin{cases} \frac{(|S|-1)!(n-|S|)!}{n!} & \text{if } i \in S, \\ \frac{-|S|!(n-|S|-1)!}{n!} & \text{otherwise.} \end{cases}$$

Note that $|\zeta_S^i|$ is the probability that in a random permutation players from $S \setminus \{i\}$ appear before i and $N \setminus (S \cup \{i\})$ —after.

In this paper, we focus on five extensions of the Shapley value to games with externalities (see (Kóczy, 2018) for a recent overview). In their definitions, we use Iverson brackets: $[\varphi] = 1$ if statement φ is true, and $[\varphi] = 0$, otherwise.

Definition 1. Extended Shapley values are defined as:

$$ESV_i(g) = \sum_{(S,P)\in EC} \omega_i(S,P) \cdot g(S,P)$$
 (1)

for some weights $\omega : EC \times N \to \mathbb{R}$:

McQuillin value (MQ-value) (McQuillin, 2009): $\omega_i(S, P) = \zeta_S^i \cdot [|P| \le 2].$

Externality-free value (EF-value) (Pham Do & Norde, 2007; De Clippel & Serrano,

$$\omega_i(S, P) = \zeta_S^i \cdot [|P| - 1 = n - |S|].$$

Hu-Yang value (HY-value) (Hu & Yang, 2010): $\omega_i(S,P) = \zeta_S^i \cdot \theta(S,P)/|\mathcal{P}|, \text{ where } \theta(S,P) = |\{P' \in \mathcal{P} : \{T \setminus S : T \in P'\} = \}|$ $P \setminus \{S\}\}|.$

Stochastic Shapley value (SS-value) (Feldman, 1996; Macho-Stadler et al., 2007; Skibski et al., 2018):

$$\omega_i(S, P) = \zeta_S^i \cdot (\prod_{T \in P \setminus \{S\}} (|T| - 1)!) / (n - |S|)!.$$

Myerson value (MY-value) (Myerson, 1977b):
$$\omega_i(S,P) = (-1)^{|P|} ((\sum_{T \in P \setminus \{S\}, i \notin T} \frac{(|P|-2)!}{(n-|T|)}) - \frac{(|P|-1)!}{n}).$$

	[4]	[3,1]	[2,2]	[2,1,1]	[1,1,1,1]	×
MQ	1	0	0	0	0	1/30
EF	0	0	0	0	1	1/30
HY	5	10	10	17	26	1/6090
SS	6	2	1	1	1	1/720
MY	10	-6	-5	9	-24	1/30

Table 2: Weights $\omega_i(S, P)$ from Definition 1 for different extended Shapley value with |N| = 6, |S| = 2 and $i \in S$. Columns are labeled with the integer partitions corresponding to sizes of coalitions in $P \setminus \{S\}$. For each value, the last column contains a common multiplier (e.g., for HY-value and $P \setminus \{S\}$ of the form $\{\{j,k\},\{l\},\{m\}\}\}$ it holds $w_i(S,P) = 17/6090$).

XY-value for player i in game g will be denoted by $XY_i(g)$.

Let us provide intuition behind these definitions. All values except for MY-value can be interpreted as a composition of two functions: first function maps a game with externalities to a game without externalities and the second function applies the original Shapley value. In the case of MQ-value and EF-value, the mapping is obtained simply by taking a value of each coalition from one specific partition. In turn, HY-value and SS-value for every coalition take a weighted average over all partitions it is in; the difference is HY-value assigns greater weights to partitions with more coalitions, while SS-value—to partitions with larger coalitions. Finally, weights of MY-value comes from the inclusion-exclusion principle caused by a strong carrier property.

See Table 2 for an example. In each row, we marked cells that will be most important in our complexity results.

2.2 Representations

Game represented as a single rule γ is denoted by g^{γ} and as the set of rules $\Gamma = \{\gamma_1, \ldots, \gamma_k\}$ by g^{Γ} .

2.2.1 MC-nets (Ieong & Shoham, 2005)

Marginal contribution nets (MC-nets) are a representation for games without externalities. The game is represented as a set of MC-nets rules of the form: $(\alpha \to c)$. Here, $c \in \mathbb{R}$ is the weight of a rule and α is a boolean expression over N of the form:

$$(a_1^+ \wedge \dots \wedge a_m^+ \wedge \neg a_1^- \wedge \dots \wedge \neg a_l^-), \tag{2}$$

where $a_1^+, \ldots, a_m^+ \in N$ are called *positive literals* and $a_1^-, \ldots, a_l^- \in N$ are called *negative literals*. We denote sets of positive and negative literals by $\oplus(\alpha)$ and $\ominus(\alpha)$, respectively, and assume $\oplus(\alpha) \cap \ominus(\alpha) = \emptyset$ and $\oplus(\alpha) \neq \emptyset$. A coalition S

 $^{^1}$ Ieong & Shoham (2005) allows rules without positive literals which entails that empty coalition may have non-zero value. As standard in the literature, we do not allow such situation.

satisfies α if it contains all positive literals and does not contain any negative literal, i.e., $\oplus(\alpha) \subseteq S$ and $\ominus(\alpha) \cap S = \emptyset$. Now, in a game represented as a set of MC-nets rules the value of coalition S is the sum of weights of all satisfied rules.

2.2.2 Embedded MC-nets (Michalak et al., 2010)

An embedded MC-nets rule is of the form:

$$(\alpha_1|\alpha_2,\alpha_3,\ldots,\alpha_k)\to c,$$

where $c \in \mathbb{R}$ is the weight of a rule and $\alpha_1, \ldots, \alpha_k$ are boolean expressions as in Eq. (2). An embedded coalition (S, P) satisfies the rule if S satisfies α_1 and for every α_i with i > 1 there exists a coalition $T \in P \setminus \{S\}$ that satisfies it. Now, in a game represented as a set of embedded MC-nets rules the value of embedded coalition (S, P) is the sum of weights of all satisfied rules.

2.2.3 Weighted MC-nets (Michalak et al., 2010)

A weighted MC-nets rule is of the form:

$$(\alpha_1^1 \to c_1^1) \dots (\alpha_{k_1}^1 \to c_{k_1}^1) | \dots | (\alpha_1^m \to c_1^m) \dots (\alpha_{k_m}^m \to c_{k_m}^m),$$

where $(\alpha_j^i \to c_j^i)$ for every $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, k_i\}$ is an MC-nets rule. A partition P satisfies the rule if it can be partitioned into m disjoint subsets $P = P_1 \dot{\cup} \ldots \dot{\cup} P_m$ such that for every $i \in \{1, \ldots, m\}$ and $j \in \{1, \ldots, k_i\}$ rule $(\alpha_j^i \to c_j^i)$ is satisfies by some coalition from P_i . Now, in a game represented as a set of weighted MC-nets rules the value of embedded coalition (S, P) is the sum of weights of all MC-nets rules $(\alpha \to c)$ that S satisfies in all (weighted MC-nets) rules satisfied by P.

Size: we will assume that the size of a rule is the number of literals plus the number of weights c.

2.3 Graphs

A graph is a pair (V, E) where V is the set of nodes and E is the set of undirected edges, i.e., subsets of nodes of size 2.

Two nodes are *adjacent* if there is an edge connecting them. A *clique* is a subset of nodes every two of which are adjacent. An *independent set* is a subset of nodes no two of which are adjacent.

A (proper vertex) k-coloring of a graph is a function, $f: V \to \{1, \ldots, k\}$, that assigns colors $\{1, \ldots, k\}$ to nodes in a way that every two adjacent nodes have different colors, i.e., $f(v) \neq f(u)$ for every $\{v, u\} \in E$. In other words, nodes colored with the same color are an independent set. The set of all k-colorings of a graph G is denoted by $C_k(G)$.

A k-coloring f results in the partition of nodes:

$$P_f = \{f^{-1}(i) : i \in \{1, \dots, k\}, f^{-1}(i) \neq \emptyset\}.$$

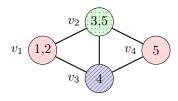


Figure 1: Graph G=(V,E) and a 4-coloring f with colors: 1 (blue/striped), 2 (yellow/checked), 3 (red/plain), 4 (green/dotted). Note that color 2 is not used.

A set in P_f that contains node v_i is denoted by S_f^i . We say that f uses exactly p colors if $|P_f| = p$. Finally, we will denote by $\theta(f)$ the number of all k-colorings that result in the same partition P_f ; note that $\theta(f) = k(k-1)\cdots(k-|P_f|+1)$.

We will consider graphs labeled with subsets of players: $l: V \to 2^N$. We will denote the label of node v by l(v) and define $l(U) = \bigcup_{v \in U} l(v)$ for a subset of nodes $U \subseteq V$ and $l(P) = \{l(U) : U \in P\}$ for a partition P of nodes V.

Example 1. Figure 1 presents an example graph G = (V, E) and a 4-coloring f. Coloring f results in the following partition of nodes: $P_f = \{\{v_1, v_4\}, \{v_2\}, \{v_3\}\}\}$ (there are $\theta(f) = 4 \cdot 3 \cdot 2$ different 4-colorings that result in this partition). We have $S_f^1 = \{v_1, v_4\}$.

Note that labels of nodes in G form a partition of the set of players $N = \{1, \ldots, 6\}$: $\{l(v_1), l(v_2), l(v_3), l(v_4)\} \in \mathcal{P}$. Hence, also $l(P_f)$ is a partition of N and $(l(S_f^1), l(P)) = (\{1, 2, 6\}, \{\{1, 2, 6\}, \{3, 5\}, \{4\}\})$ is an embedded coalition.

3 Mapping MC-Nets into Graphs

The goal of this section is to show that (1) every embedded and weighted MC-nets rule can be represented as (one or more) graphs, and (2) game represented as embedded and weighted MC-nets can be defined based on (proper vertex) colorings in such graphs.

To this end, we begin by introducing a subclass of weighted MC-nets rules under the name *hybrid rules*. The name comes from the fact that hybrid rules, while they are formally weighted MC-nets rules, have a form almost identical to embedded MC-nets rules.

Definition 2. (Hybrid rules) A *hybrid rule* is a weighted MC-nets rule of the form:

$$\gamma = (\alpha_1 \to c)(\alpha_2 \to 0) \dots (\alpha_k \to 0),$$

with $\oplus(\alpha_i)\cap\oplus(\alpha_j)=\emptyset$ for $i,j\in\{1,\ldots,k\},\ i\neq j$ and $\bigcup_{i=1}^k\oplus(\alpha_i)=N$. We will call $c\in\mathbb{R}$ the weight of rule γ .

Note that for every hybrid rule $\{\oplus(\alpha_1), \ldots, \oplus(\alpha_k)\}$ is a partition of N. Based on the definition of weighted MC-nets, embedded coalition (S, P) satisfies hybrid

rule γ if S satisfies α_1 and for every α_i with i > 1 there exists a coalition in P that satisfies it (note that, unlike embedded MC-nets, S may also satisfy α_i for i > 1).

In the following lemma we show that every weighted MC-nets rule can be expressed using polynomially many hybrid rules.

Lemma 1. Every weighted MC-nets rule of size S is equivalent to a set of hybrid rules of size poly(n, S).

Proof. All proofs can be found in the appendix.

In turn, in Lemma 2, we show that embedded MC-nets rules are equivalent to a subset of hybrid rules. We will use a notion of compatibility: we say that expressions α_i, α_j are *compatible*, denoted by $\text{comp}(\alpha_i, \alpha_j)$, if there exists a coalition that satisfies both of them, i.e., if $(\oplus(\alpha_i)\cup\oplus(\alpha_j))\cap(\ominus(\alpha_i)\cup\ominus(\alpha_j))=\emptyset$.

Lemma 2. Every embedded MC-nets rule of size S is equivalent to a hybrid rule of size poly(n, S) satisfying:

$$\forall_{1 < i, j \le k} ((\text{comp}(\alpha_1, \alpha_i) \land \text{comp}(\alpha_1, \alpha_j)) \to (\text{comp}(\alpha_i, \alpha_i) \land (| \oplus (\alpha_i)| = | \oplus (\alpha_i)| = 1)). \tag{*}$$

Moreover, every hybrid rule of size S satisfying (*) is equivalent to an embedded MC-nets rule of size poly(n, S).

Lemmas 1 and 2 will be crucial in our complexity analysis as they allow us to focus on hybrid rules.

So far, we have shown the mapping from weighted MC-nets and embedded MC-nets rules to hybrid rules. In what follows, we show that every hybrid rule can be represented as a graph.

Definition 3. (Graph G^{γ}) For a hybrid rule γ , graph $G^{\gamma} = (V, E)$ is a graph where nodes represent expressions $\alpha_1, \ldots, \alpha_k$ and are labeled with sets $\oplus(\alpha_1), \ldots, \oplus(\alpha_k)$ and edges connect incompatible expressions α_i, α_j :

- $V = \{v_1, \ldots, v_k\}$ with $l(v_i) = \bigoplus (\alpha_i)$ for every $v_i \in V$;
- $E = \{\{v_i, v_j\} \subseteq V : \neg comp(\alpha_i, \alpha_j)\}.$

We note that a similar construction of a graph for the right-hand side part of the embedded MC-nets was proposed by Skibski et al. (2016).

Example 2. Consider a hybrid rule $\gamma = (\alpha_1 \to 1)(\alpha_2 \to 0)(\alpha_3 \to 0)(\alpha_4 \to 0)$ with:

$$\begin{array}{ll} \alpha_1 = (1 \wedge 2 \wedge \neg 3), & \alpha_2 = (3 \wedge 5), \\ \alpha_3 = (4 \wedge \neg 1 \wedge \neg 3 \wedge \neg 6), & \alpha_4 = (6 \wedge \neg 5). \end{array}$$

Note that $comp(\alpha_1, \alpha_4)$ and $\neg comp(\alpha_i, \alpha_j)$ for other $i, j \in \{1, ..., 4\}, i \neq j$. Graph G^{γ} is depicted in Figure 1.

Since α_4 is the only expression compatible with α_1 (other than α_1 itself) and $| \oplus (\alpha_4)| = 1$, we get that γ satisfies condition (*). Hence, from Lemma 2 we know that it is equivalent to some embedded MC-nets rule. One such a rule is: $(1 \wedge 2)|(3 \wedge 5 \wedge \neg 6)(4 \wedge \neg 3 \wedge \neg 6) \rightarrow 1$.

Definition 3 shows that every hybrid rule can be represented as a graph in which labels of nodes form a partition of players. The natural question arises: does every such graph represent some hybrid rule? The answer is "yes", which we prove in the following lemma.

Lemma 3. For a graph G = (V, E) with labels l there exists a hybrid rule γ such that $G = G^{\gamma}$ if and only if $\{l(v) : v \in V\}$ is a partition of N.

The next lemma states the necessary and sufficient conditions for the graph to represent a hybrid rule that satisfies condition (*).

Lemma 4. For a graph G = (V, E) with labels l there exists a hybrid rule γ satisfying condition (*) such that $G = G^{\gamma}$ if and only if:

- (i) $\{l(v): v \in V\}$ is a partition of N;
- (ii) $\{u \in V \setminus \{v_1\} : \{v_1, u\} \notin E\}$ (nodes not adjacent to v_1) form an independent set; and
- (iii) |l(u)| = 1 for every node u not adjacent to v_1 ($u \neq v_1$).

Based on Lemmas 1–4 we know that every weighted MC-nets can be represented as (one or more) graphs and every embedded MC-nets can be represented as a graph satisfying conditions (i)–(iii).

Let us now explain how a game represented as a hybrid rule γ can be defined based on graph G^{γ} . Fix a hybrid rule γ and consider a partition P that satisfies it. Since every node in graph G^{γ} is labeled with a set of players which is equal to the set of positive literals in some expression α_i , it is clear that all these players must appear in the same coalition in P. This observation combined with the fact that every player appears in exactly one node implies that P can be associated with a partition of nodes in graph G^{γ} .

Consider partition P^* of nodes that correspond to P. Note that two adjacent nodes cannot belong to the same coalition in P^* , because they represent incompatible expressions that cannot be satisfied by one coalition. Hence, every set in P^* is an independent set. In result, we get that P^* corresponds to some coloring of a graph.

This analysis is formalized in the following lemma.

Lemma 5. Partition P satisfies a hybrid rule γ if and only if there exists k-coloring $f \in C_k(G^{\gamma})$ such that $P = l(P_f)$. Moreover:

$$g^{\gamma}(S, P) = c \cdot \left[\exists_{f \in C_k(G^{\gamma})} (P = l(P_f) \land S = l(S_f^1)) \right]. \tag{3}$$

Example 3. Consider a hybrid rule γ from Example 2 with graph G^{γ} depicted in Figure 1. Let us discuss all possible 4-colorings of G^{γ} :

- There are no colorings of G^{γ} that uses 2 or 3 colors.
- There are 24 colorings that uses 3 colors: $f(v_1) = f(v_4) = a$, $f(v_2) = b$ and $f(v_3) = c$ where $a, b, c \in \{1, ..., 4\}$ are different colors. Note that for every such coloring f we have: $P_f = \{\{1, 2, 6\}, \{3, 5\}, \{4\}\}.$

• There are 24 colorings that uses 4 colors; in these colorings all nodes have different colors. For every such coloring $f: P_f = \{\{1,2\}, \{3,5\}, \{4\}, \{6\}\}\}$.

Overall, 48 colorings results in two partitions of players. Now, from Lemma 5, game q^{γ} is defined as follows:

$$\begin{array}{l} g^{\gamma}(\{1,2,6\},\{\{1,2,6\},\{3,5\},\{4\}\})=1,\\ g^{\gamma}(\{1,2\},\{\{1,2\},\{3,5\},\{4\},\{6\}\})=1, \end{array}$$

and $g^{\gamma}(S, P) = 0$ for the remaining embedded coalitions.

In the following section, we consider computing extended Shapley values in games defined with Eq. (3).

4 Computing extended Shapley values

Recall Definition 1. From Eq. (1) we know that all extended Shapley values considered by us satisfy linearity, i.e., ESV(g+g')=ESV(g)+ESV(g') and $ESV(c\cdot g)=c\cdot ESV(g)$ for every two games g,g' and $c\in\mathbb{R}$. Thus, in our computational analysis we can focus on games represented as a single rule and, based on Lemmas 1 and 2, as a single hybrid rule. Moreover, we can assume the weight of this rule is 1 (i.e., c=1). Hence, from now on, we will assume that game is represented as a hybrid rule with weight 1.

Fix such a hybrid rule γ . In Lemma 5, we showed that the value of an embedded coalition (S, P) is non-zero if and only if there exists a coloring f in graph G^{γ} such that $P = P_f$. Since there are $\theta(f)$ colorings that results in the same partition as f, from Lemma 5 and Eq. (1) we get the following formula for extended Shapley values:

$$ESV_i(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \omega_i(l(S_f^1), l(P_f)) / \theta(f). \tag{4}$$

To put it in words, extended Shapley value in game g^{γ} is a weighted sum over all colorings in graph G^{γ} . Weights depend on $l(P_f)$ (partition of players resulting from the coloring f), $l(S_f^1)$ (union of labels of all nodes colored with the same color as node v_1), and player $i \in N$. For an example we refer the reader to Table 3 in the appendix.

More generally, we can consider the following counting problem that we name Weighted Coloring Counting. The problem is parametrized with weights $\omega^*: C_k(G) \to \mathbb{R}$ that for each coloring assigns some real value.

Definition 4. ω^* -Weighted Coloring Counting

Input: graph G=(V,E), labels $l:V\to 2^N$ s.t. $\{l(v):v\in V\}$ is a partition of players N

Output:
$$\sum_{f \in C_{|V|}(G)} \omega^*(f)$$
.

This problem in general is computationally challenging, as it generalizes the problem of counting all k-colorings which is #P-complete and allows us to determine whether a graph is 3-colorable which is NP-complete.

Based on Eq. (4), computing each extended Shapley value for a fixed player can be considered a special case of WEIGHTED COLORING COUNTING. In the following sections, we analyze these problems one by one. Values are ordered in ascending order by the complexity of their formula:

- First two values, MQ-value and EF-value, take into account only one partition P for every coalition S; hence, they can be computed by traversing all subsets, not all partitions of players.
- In HY-value, considered third, the weight of an embedded coalition (S, P) depends solely on |S| and |P|; this allows us to group all colorings that use the same number of colors.
- Finally, in the last two values, SS-value and MY-value, weights depend on sizes of all coalitions in a partition.

Before we move to the next section, let us roughly explain a technique that we use in the proofs of Theorem 9, 10 and 14. Assume we want to compute x_1, \ldots, x_k and we have an algorithm that computes the sum $f(j) = \sum_{m=1}^k a_{j,m}x_m$ for some weights a that depend on m and some external parameter $j \in \{1, \ldots, k\}$. To this end, we can construct a system of linear equations with the following matrix form:

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \dots & a_{1,k} \\ a_{2,1} & a_{2,2} & \dots & a_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{k,1} & a_{k,2} & \dots & a_{k,k} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_k \end{bmatrix} = \begin{bmatrix} f(1) \\ f(2) \\ \vdots \\ f(k) \end{bmatrix}.$$
 (5)

Now, if the matrix $(a_{j,m})_{1 \leq j,m \leq k}$ has non-zero determinant, then it is invertible. Hence, if we know $f(1), \ldots, f(k)$, then using Gaussian elimination we can compute x_1, \ldots, x_k .

In our case, f(j) will be an extended Shapley value and x_m will be the number of independent sets of size m (Theorem 9), k-coloring that uses m colors (Theorem 10) or matchings in a bipartite graph of size m (Theorem 14). Hence, based on the fact that computing $\sum_{m=1}^k x_m$ is #P-hard we will get that computing EF-value, HY-value and MY-value is also #P-hard.

Note that based on Eq. (4) each extended Shapley value is a sum over exponentially many colorings and, in general, two colorings that result in different partitions of nodes may have different weights. Hence, the main challenge with this approach is to (1) express an extended Shapley value as a weighted sum over polynomial number of elements and (2) creating a system of linear equations that results in a matrix which is invertible.

4.1 Computing MQ-value

We begin with the analysis of MQ-value. Consider Eq. (4) for weights of MQ-value (Definition 1):

$$MQ_i(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \frac{\zeta_{l(S_f^1)}^i}{\theta(f)} \cdot [|l(P_f) \le 2].$$

As we can see, only colorings that uses 1 or 2 colors have non-zero weights. It is easy to verify that instead of going through all k-colorings it is enough to consider 2-colorings of a graph. Using the fact that for every 2-coloring f we have $\theta(f) = 2$, we get:

$$MQ_i(g^{\gamma}) = \frac{1}{2} \sum_{f \in C_2(G^{\gamma})} \zeta_{l(S_f^1)}^i.$$
 (6)

While in a connected graph there are at most two 2-colorings, in a disconnected graph it can be exponentially many. Nevertheless, in Theorem 6, we show that this sum can be computed in polynomial time for every graph.

Theorem 6. For a game represented as weighted MC-nets, MQ-value can be computed in polynomial time.

Sketch of proof. From Lemma 1 and linearity of MQ-value we can focus on hybrid rules with weight 1. Fix such a hybrid rule γ . We will focus on the simplest case: $i \in l(v_1)$. Let us define a table T[1...n] as follows: $T[s] = |\{f \in C_2(G^{\gamma}) : |l(S_f^1)| = s\}|$. To put it in words, T[s] is the number of 2-colorings of G^{γ} in which there are s players in nodes colored with the same color as node v_1 . From Eq. (6) we have:

$$MQ_i(g^{\gamma}) = \sum_{s=1}^n \frac{(s-1)!(n-s)!}{(2n!)} \cdot T[s].$$

Thus, it is enough to compute table T.

Table T can be computed with the following dynamic programming algorithm. Assume T is filled with zeros at the beginning. Let A_1, \ldots, A_m be connected components of graph G^{γ} , and let $\{B_j, C_j\}$ be a unique partition of A_j into two independent sets. Components and partitions can be found by performing several breadth-first searches. If partition $\{B_j, C_j\}$ does not exist for at least one component A_j , then the calculation is complete (T is a zero table). Assume otherwise and assume $v_1 \in B_1$. We begin by assigning $T[|l(B_1)|] = 2$, since there are two 2-colorings of A_1 . Now, we consider components A_2, \ldots, A_m , one by one, and for each component consider two cases: either B_j or C_j is colored with the same color as node v_1 . Thus, in each step, we update table T by replacing it with a new table T' defined as follows: $T'[s] = T[s - |l(B_j)|] + T[s - |l(C_j)|]$ for $1 \le s \le n$ (assuming T[s] = 0 for $s \le 0$). After the m-th step, the calculation is complete.

Theorem 6 implies polynomial computation also for embedded MC-nets.

Corollary 7. For game represented as embedded MC-nets, MQ-value can be computed in polynomial time.

Proof. Directly from Theorem 6. This result was also proved by (Michalak et al., 2010).

4.2 Computing EF-value

EF-value, considered by us next, is complementary to MQ-value. Consider Eq. (4) for weights of EF-value (Def. 1):

$$EF_i(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \frac{\zeta_{l(S_f^1)}^i}{\theta(f)} [|P_f| - 1 = n - |l(S_f^1)|].$$

Note that condition $|P_f| - 1 = n - |l(S_f^1)|$ holds if and only if every player $i \in N \setminus l(S_f^1)$ form a singleton coalition $\{i\}$ in $l(P_f)$, i.e., node v such that $i \in l(v)$ has a label of size one and is colored with a different color than all other nodes. Hence, P_f is uniquely defined by set S_f^1 .

Let us analyze the conditions on set S_f^1 . Let V^* be the set of nodes with non-singleton labels. For an arbitrary set of nodes $S \subseteq V$, there exists a coloring f with non-zero weight such that $S = S_f^1$ if and only if: (1) S is an independent set; (2) S contains node v_1 ; and (3) S contains set V^* . This implies that if $V^* \cup \{v_1\}$ does not form an independent set, then the formula evaluates to zero. Assume otherwise.

We get that S is the union of $V^* \cup \{v_1\}$ and an independent set of nodes which are not adjacent to $V^* \cup \{v_1\}$. Let us denote these nodes by U (formally, $U = \{u \in V \setminus (V^* \cup \{v_1\}) : \{u, w\} \in E \to w \notin V^* \cup \{v_1\}\}$). Also, if $S = S_f^1$ for some coloring f, then there are $\theta(f)$ colorings with the same set S_f^1 . Hence, we get the following formula for EF-value:

$$EF_i(g^{\gamma}) = \sum_{S \in I(G^{\gamma}): S \subseteq U} \zeta^i_{l(S \cup V^* \cup \{v_1\})}$$
(7)

where $I(G^{\gamma})$ is the set of all independent sets in graph G^{γ} .

In the following two theorems, we show that this sum is hard to compute in general, but it is easy to compute if graph satisfies condition stated in Lemma 4.

Theorem 8. For game represented as embedded MC-nets, EF-value can be computed in polynomial time.

Sketch of proof. From Lemma 2 it is enough to consider hybrid rules that satisfy condition (*) with weight 1. Fix such a hybrid rule γ and consider G^{γ} . If $V^* \setminus \{v_1\} \neq \emptyset$, i.e., there exists a node, other than v_1 , with the size of a label larger than one, then from Lemma 4 it must be adjacent to v_1 ; hence, $EF_i(g^{\gamma}) = 0$ for every $i \in N$. Assume otherwise. We get that $V^* \cup \{v_1\} = \{v_1\}$, and U

is the set of nodes not adjacent to v_1 . From Lemma 4 we know that U is an independent set. Thus, $EF_i(g^{\gamma}) = \sum_{S \subseteq U} \zeta^i_{l(S \cup \{v_1\})}$ which can be computed in polynomial time.

Theorem 9. For game represented as weighted MC-nets, computing EF-value is #P-complete.

Sketch of proof. To show that the problem is #P-complete, we use the (Turing) reduction from the problem of counting all independent sets in a graph which is #P-complete (Valiant, 1979). We use a technique described at the beginning of the section (see Eq. (5)). Specifically, for an arbitrary graph G = (V, E), we label each node with one player and add a new node, v_1 , labeled with j + 1 new players. We get that U = V and for $i \in l(v_1)$ Eq. (7) iterates over independent sets in G and weights depends only their size and parameter j.

4.3 Computing HY-value

HY-value is the first value considered by us with non-zero weights of every embedded coalition. Let us recall these weights from Definition 1: $\omega_i(S, P) = \zeta_S^i \cdot \theta(S, P)/|\mathcal{P}|$, where $\theta(S, P) = |\{P' \in \mathcal{P} : \{T \setminus S : T \in P'\} = P \setminus \{S\}\}|$. To put it in word, $\theta(S, P)$ is the number of partitions that can be obtained from $P \setminus \{S\}$ by inserting players from S.

Let us introduce a notion of *Bell numbers*. The *n*-th *Bell number*, denoted B_n , is the number of all possible partitions of n elements. Now, r-Bell numbers are a generalization of Bell numbers: $B_{n,r}$ is the number of partitions of n+r elements such that the first r elements are in distinct subsets (Mezo, 2011). In particular, $B_{n,0} = B_n$ and $B_{n,1} = B_{n+1}$.

Now, observe that $\theta(S, P) = B_{|S|, |P|-1}$. Thus, HY-value weights combined with Eq. (4) yields:

$$HY_i(g^{\gamma}) = \frac{1}{B_n} \sum_{f \in C_k(G^{\gamma})} \frac{\zeta_{l(S_f^1)}^i}{\theta(f)} \cdot B_{|l(S_f^1)|,|l(P_f)|-1}.$$
 (8)

Thus, for a fixed player i and size of $l(S_f^1)$, the weight of a coloring depends solely on the number of colors it uses (see Table 2 for an illustration).

We will prove that computing this sum is #P-complete.

Theorem 10. For game represented as embedded MC-nets or weighted MC-nets, computing HY-value is #P-complete.

Sketch of proof. To show that the problem is #P-complete, we use the (Turing) reduction from the *chromatic polynomial problem*, i.e., counting m-colorings in a graph, which is #P-complete (Jaeger et al., 1990).

Let G = (V, E) be an arbitrary graph and c_m be the number of k-colorings that use exactly m colors. For every $j \in \{1, \ldots, k\}$ we construct a graph $G^{\gamma_j} = (V \cup \{v_1\}, E \cup \{\{v_1, v_i\} : v_i \in V) \text{ with } l(v_i) = \{i\} \text{ for } v_i \in V \text{ and } l(v_1) = \{1\} \cup \{k+2, \ldots, k+j\}$. Since node v_1 is connected to all other nodes we know

that G^{γ_j} satisfies conditions from Lemma 4 and there exists a hybrid rule γ_j equivalent to some embedded MC-nets rule such that G^{γ_j} is the corresponding graph. Now, it can be shown that

$$HY_1(g^{\gamma_j}) = \frac{(j-1)!}{(k+j)!B_{k+j}} \sum_{m=1}^k (k-m)!B_{j,m}c_m.$$

In result, we get a system of linear equations from Eq. (5) where $a_{j,m} = (k - m)!B_{j,m}$, $x_m = c_m$ and $f(j) = ((k+j)!/(j-1)!)B_{k+j}HY_1(g^{\gamma_j})$.

To show that the determinant of $A = (a_{j,m})_{1 \leq j,m \leq k}$ is non-zero we first prove a general formula for r-Bell numbers (Lemma 11) and then, using this formula, we prove that the determinant of matrix $(B_{j,m})_{1 \leq j,m \leq k}$ is equal to the known determinant of matrix $(B_{j+m})_{1 \leq j,m \leq k}$ (Lemma 12).

Lemma 11. For every $n, r \in \mathbb{N}$, $B_{n+r} = \sum_{i=1}^{r} \{r, i\} B_{n,i}$, where $\{r, i\}$ is the Stirling number of the second kind, i.e., the number of partitions of r elements into i subsets.

Lemma 12. The determinant of matrix $B = (B_{j,m})_{1 \leq j,m \leq k}$ equals $(\prod_{i=0}^k i!) \cdot (\sum_{i=0}^k 1/i!)$.

This concludes the proof.

4.4 Computing SS-value

SS-value, considered next, is probably the most popular extended Shapley value. Weights of SS-value (see Definition 1) combined with Eq. (4) yields:

$$SS_{i}(g^{\gamma}) = \sum_{f \in C_{b}(G^{\gamma})} \frac{\zeta_{l(S_{f}^{1})}^{i}}{\theta(f)} \frac{\prod_{T \in P_{f} \setminus \{S_{f}^{1}\}} (|l(T)| - 1)!}{(n - |l(S_{f}^{1})|)!}.$$

In what follows, let us focus on graphs in which every node is labeled with a single player: |l(v)| = 1 for every $v \in V$. In such a case, we have n = k = |V| and |l(T)| = |T|. Under this assumption, formula for SS-value of player $i \in l(v_1)$ is as follows:

$$SS_i(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \frac{\prod_{T \in P_f} (|T| - 1)!}{\theta(f) \cdot n!}.$$
 (9)

For a fixed partition $P \in \mathcal{P}$, value $\prod_{T \in \mathcal{P}} (|T|-1)!$ is the number of all permutations in which P is the partition obtained from a cycle decomposition (for a permutation $h: N \to N$ such partition is defined as follows: $\{\{i, h(i), h(h(i)), \dots\}: i \in N\}$). Hence, Eq. (9) is the probability that the partition obtained from a cycle decomposition of a random permutation corresponds to a (proper vertex) coloring in graph G^{γ} .

Let us consider a complement of graph $G^{\gamma} = (V, E)$:

$$\overline{G^{\gamma}} = (V, \{\{i, j\} : i, j \in V, i \neq j\} \setminus E).$$

For every coloring $f \in C_k(G^{\gamma})$, sets of nodes in P_f are independent sets in G^{γ} . Hence, they are cliques in $\overline{G^{\gamma}}$. As a consequence, we get that $SS_i(g^{\gamma})$ from Eq. (9) is equivalently a weighted sum over *clique covers* (i.e., partitions of the nodes in a graph into cliques):

$$SS_i(g^{\gamma}) = \frac{1}{n!} \sum_{P \in QC(\overline{G^{\gamma}})} \prod_{T \in P} (|T| - 1)!, \tag{10}$$

where QC(G) is the set of all clique covers in graph G.

Now, assume that $\overline{G^{\gamma}}$ is a bipartite graph. In such a case, there are no cliques of size larger than 2 and each partition into cliques is equivalent to a matching (not necessary perfect or maximal) in this graph. Moreover, for each such a partition we have $\prod_{T\in P}(|T|-1)!=1$ (see also Table 2). Hence, Eq. (10) is equal to the number of matchings in $\overline{G^{\gamma}}$ divided by n!. We formalize this reasoning in the following theorem.

Theorem 13. For game represented as embedded MC-nets or weighted MC-nets, computing SS-value is #P-complete.

Sketch of proof. To show that the problem is #P-complete, we use a reduction from the problem of counting all matchings in a bipartite graph which is #P-complete (Valiant, 1979). For an arbitrary bipartite graph G = (V, E) we construct a graph $\overline{G^{\gamma}}$ by adding a new node, v_1 , and we label each node with one player. Since node v_1 does not have any edges in $\overline{G^{\gamma}}$, then it is connected to all nodes in G^{γ} ; hence, G^{γ} satisfies conditions of Lemma 4 and from Lemma 2 there exist a hybrid rule γ equivalent to some embedded MC-nets rule such that G^{γ} is the corresponding graph.

Now, from Eq. (10) for $i \in l(v_1)$ we get that $SS_i(g^{\gamma})/n!$ equals the number of matchings in \overline{G}^{γ} , so also in G.

4.5 Computing MY-value

The last value that we consider is MY-value which is the first chronologically proposed extension of the Shapley value to games with externalities. Eq. (4) for weights of MY-value (Definition 1) gives:

$$MY_i(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \frac{(-1)^{|P_f|} (|P_f| - 2)!}{\theta(f)} \cdot h_i(f),$$

with $h_i(f) = \left(\sum_{T \in P_f \setminus \{S_f^1\}, i \notin l(T)} \frac{1}{(n-|l(T)|)}\right) - \frac{|P_f|-1}{n}$. From Table 2 it is visible that both techniques used for HY-value and SS-value does not work in this case.

To cope with this problem, we will exploit the fact that weights of MY-value have a form of a sum over all coalitions. Specifically, we will consider a difference between the MY-value of two players. Let us denote such difference for players i and j in game g by $\Delta_i^j(g)$: $\Delta_i^j(g) = MY_i(g) - MY_j(g)$. Now, for $i \in l(S_f^1)$ and

 $j \in l(T)$ for some $T \in P_f \setminus \{S_f^1\}$ we get $h_i(f) - h_j(f) = 1/(n - |T|)$ and

$$\Delta_i^j(g^{\gamma}) = \sum_{f \in C_k(G^{\gamma})} \frac{(-1)^{|P_f|} (|P_f| - 2)!}{\theta(f) \cdot (n - |l(T)|)}.$$
 (11)

Note that if j is in a label of a node adjacent to all other nodes, then the weight of a coloring depends solely on the number of colors it uses. Hence, we can use a technique described at the beginning of this section (see Eq. (5)).

Theorem 14. For game represented as embedded MC-nets or weighted MC-nets, computing MY-value is #P-complete.

Sketch of proof. To show that the problem is #P-complete, again we use the reduction from the problem of counting all matching in a bipartite graph which is #P-complete (Valiant, 1979). With the same reasoning as in SS-value, instead of considering colorings in graph G^{γ} we will focus on clique covers in the complement graph $\overline{G^{\gamma}}$. For an arbitrary bipartite graph G = (V, E) for $j \in \{1, \ldots, k\}$, we construct a graph $\overline{G^{\gamma_j}}$ by adding j+2 isolated nodes and consider Δ_j^i for two players from newly added nodes. In this way, based on Eq. (11) we build a system of linear equations, as in Eq. (5).

5 Conclusions

In this paper, we studied the complexity of computing extended Shapley value in games represented as embedded and weighted MC-nets. Our results show that weighted MC-nets, which are more concise than embedded MC-nets, are slightly worse when it comes to the Shapley value computation. Also, combined with the work by Skibski et al. (2015), we get that computational properties of partition decision trees are significantly better than both MC-nets representations.

There are many possible directions of further research. The extended Shapley value proposed by Bolger (1989), as well as other solution concepts can be considered. Also, it would be interesting to analyze hybrid rules and corresponding graphs not as a representation, but as a graph-restriction scheme for games with externalities (Myerson, 1977a).

References

- Abe, T. & Funaki, Y. (2017). The non-emptiness of the core of a partition function form game. *International Journal of Game Theory*, 46(3), 715–736.
- Aigner, M. (1999). A characterization of the Bell numbers. *Discrete mathematics*, 205(1-3), 207–210.
- Bacher, R. (2002). Determinants of matrices related to the Pascal triangle. Journal de théorie des nombres de Bordeaux, 14(1), 19–41.
- Bolger, E. M. (1989). A set of axioms for a value for partition function games. *International Journal of Game Theory*, 18(1), 37–44.
- Chalkiadakis, G., Elkind, E., & Wooldridge, M. (2011). Computational aspects of cooperative game theory. Synthesis Lectures on Artificial Intelligence and Machine Learning, 5(6), 1–168.
- De Clippel, G. & Serrano, R. (2008). Marginal contributions and externalities in the value. *Econometrica*, 76(6), 1413–1436.
- Dunne, P. E. (2005). Multiagent resource allocation in the presence of externalities. In *Multi-Agent Systems and Applications IV* (pp. 408–417). Springer.
- Feldman, B. E. (1996). Bargaining, coalition formation, and value. PhD thesis, State University of New York at Stony Brook.
- Hu, C.-C. & Yang, Y.-Y. (2010). An axiomatic characterization of a value for games in partition function form. *SERIEs*, 1(4), 475–487.
- 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 (ACM-EC)*, (pp. 193–202).
- Jaeger, F., Vertigan, D. L., & Welsh, D. J. (1990). On the computational complexity of the Jones and Tutte polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, 108(1), 35–53.
- Kóczy, L. Á. (2018). The Shapley-value. In *Partition Function Form Games:* Coalitional Games with Externalities (pp. 173–200). Springer.
- Macho-Stadler, I., Pérez-Castrillo, D., & Wettstein, D. (2007). Sharing the surplus: An extension of the Shapley value for environments with externalities. *Journal of Economic Theory*, 135(1), 339–356.
- McQuillin, B. (2009). The extended and generalized Shapley value: Simultaneous consideration of coalitional externalities and coalitional structure. *Journal of Economic Theory*, 144(2), 696–721.
- Mezo, I. (2011). The r-Bell numbers. Journal of Integer Sequences, 14(2), 3.

- Michalak, T. P., Marciniak, D., Szamotulski, M., Rahwan, T., Wooldridge, M., McBurney, P., & Jennings, N. R. (2010). A logic-based representation for coalitional games with externalities. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, (pp. 125–132).
- Michalak, T. P., Rahwan, T., Marciniak, D., Szamotulski, M., & Jennings, N. R. (2010). Computational aspects of extending the Shapley value to coalitional games with externalities. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI)*, (pp. 197–202).
- Myerson, R. B. (1977a). Graphs and cooperation in games. *Mathematical Methods of Operations Research*, 2(3), 225–229.
- Myerson, R. B. (1977b). Values of games in partition function form. *International Journal of Game Theory*, 6, 23–31.
- Pham Do, K. H. & Norde, H. (2007). The Shapley value for partition function form games. *International Game Theory Review*, 9(02), 353–360.
- Rahwan, T., Michalak, T. P., Wooldridge, M., & Jennings, N. R. (2012). Anytime coaliton structure generation in multi-agent systems with positive or negative externalities. *Artificial Intelligence*, 186(0), 95–122.
- Shapley, L. S. (1953). A value for n-person games. In H. Kuhn & A. Tucker (Eds.), *Contributions to the Theory of Games*, volume II (pp. 307–317). Princeton University Press.
- Skibski, O., Matejczyk, S., Michalak, T., Wooldridge, M., & Yokoo, M. (2016). k-Coalitional cooperative games. In *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, (pp. 177–185).
- Skibski, O., Michalak, T. P., Sakurai, Y., Wooldridge, M., & Yokoo, M. (2015). A graphical representation for games in partition function form. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence (AAAI)*, (pp. 1036–1042).
- Skibski, O., Michalak, T. P., & Wooldridge, M. (2018). The Stochastic Shapley value for games with externalities. *Games and Economic Behavior*, 108, 65–80.
- Valiant, L. G. (1979). The complexity of enumeration and reliability problems. SIAM Journal on Computing, 8(3), 410–421.
- Yi, S.-S. (2003). Endogenous formation of economic coalitions: A survey on the partition function approach. In *The Endogenous Formation of Economic Coalitions* (pp. 80–127). London, UK: Edward Elgar.

f		$\theta(f)$	$l(S_f^1)$	$l(P_f) \setminus \{l(S_f^1)\}$	$\omega_1(l(S_f^1), l(P_f))$				
	$ P_f $	- (3)	,	-(1) (((()))	MQ	EF	HY	SS	MY
1 - 2 - 3 4.6 - 5	2	20	{1,3,4,6}	$\{\{2,5\}\}$	1	0	52	6	5
1 - 2 - 3 4.6 - 5	2	20	$\{1, 3, 5\}$	$\{\{2,4,6\}\}$	1	0	15	4	10
1 - 2 - 3 4,6 - 5	3	60	{1,3,4,6}	$\{\{2\}, \{5\}\}$	0	1	151	6	-4
1 - 2 - 3 (4.6) - 5	3	60	$\{1, 3, 5\}$	$\{\{2\},\{4,6\}\}$	0	0	37	2	-7
1 3 4,6 5	3	60	$\{1, 4, 6\}$	$\{\{2,5\},\{3\}\}$	0	0	37	2	-7
1 3 4,6 5	3	60	$\{1, 4, 6\}$	$\{\{2\},\{3,5\}\}$	0	0	37	2	-7
1 - 2 - 3 4.6 - 5	3	60	$\{1, 3\}$	$\{\{2,5\},\{4,6\}\}$	0	0	20	1	-10
1 - 2 - 3 4.6 - 5	3	60	$\{1, 3\}$	$\{\{2,4,6\},\{5\}\}$	0	0	20	2	-12
1 - 2 - 3 4.6 - 5	3	60	$\{1, 5\}$	$\{\{2,4,6\},\{3\}\}$	0	0	20	2	-12
1 - 2 - 3 4,6 - 5	3	60	{1,5}	$\{\{2\},\{3,4,6\}\}$	0	0	20	2	-12
1 - 2 - 3 4.6 - 5	3	60	{1}	$\{\{2,4,6\},\{3,5\}\}$	0	0	30	2	-15
1 3 4.6 5	3	60	{1}	$\{\{2,5\},\{3,4,6\}\}$	0	0	30	2	-15
1 - 2 - 3 4,6 - 5	4	120	{1,4,6}	{{2},{3},{5}}	0	1	77	2	12
1 - 2 - 3 4.6 - 5	4	120	{1,3}	{{2}, {4,6}, {5}}	0	0	34	1	18
1 3 4.6 5	4	120	{1,5}	{{2},{3},{4,6}}	0	0	34	1	18
1 - 2 - 3 4.6 - 5	4	120	{1}	$\{\{2,5\},\{3\},\{4,6\}\}$	0	0	40	1	24
1 - 2 - 3 4.6 - 5	4	120	{1}	{{2},{3,5},{4,6}}	0	0	40	1	24
1 - 2 - 3 4.6 - 5	4	120	{1}	{{2,4,6},{3},{5}}	0	0	40	2	28
1 3 4,6 5	4	120	{1}	{{2},{3,4,6},{5}}	0	0	40	2	28
1 3 4.6 5	5	120	{1}	{{2},{3},{4,6},{5}}	0	0	40	1	-66
				-	$\times \frac{1}{60}$	$\times \frac{1}{60}$	$\times \frac{1}{12180}$	$\times \frac{1}{7200}$	$\times \frac{1}{60}$

Table 3: Example of weights associated with all possible colorings in a simple graph according to extended Shapley values. Graph G=(V,E) has $V=\{v_1,\ldots,v_5\}$ and $E=\{\{v_1,v_2\},\{v_2,v_3\},\{v_4,v_5\}\}$ and labels: $l(v_i)=i$ for $i\in\{1,2,3,5\}$ and $l(v_4)=\{4,6\}$ and we consider all 5-colorings $f:V\to\{1,\ldots,5\}$ (isomorphic colorings are grouped). For each value, the last row contains a common multiplier.

A Proof of Lemma 1

Proof. Consider a weighted MC-nets rule:

$$(\alpha_1^1 \to c_1^1) \dots (\alpha_{k_1}^1 \to c_{k_1}^1) | \dots | (\alpha_1^m \to c_1^m) \dots (\alpha_{k_m}^m \to c_{k_m}^m).$$

Assume $\oplus(\alpha_j^i)\cap\oplus(\alpha_{j'}^{i'})\neq\emptyset$ for some $i,i'\in\{1,\ldots,m\},\ j\in\{1,\ldots,k_i\},\ j'\in\{1,\ldots,k_{i'}\}$. If $i\neq i'$ or i=i' but expressions α_j^i and $\alpha_{j'}^{i'}$ are not compatible, then the weighted MC-nets rule is contradictory; hence it is equivalent to an empty set of hybrid rules. If i=i' and α_j^i and $\alpha_{j'}^{i'}$ are compatible, then rules $(\alpha_j^i\to c_j^i)$ and $(\alpha_{j'}^{i'}\to c_{j'}^{i'})$ can be combined into $(\alpha_j^i\wedge\alpha_{j'}^{i'}\to(c_j^i+c_{j'}^{i'}))$. Hence, in what follows, we assume that $\oplus(\alpha_j^i)\cap\oplus(\alpha_{j'}^{i'})=\emptyset$ for every $i,i'\in\{1,\ldots,m\},\ j\in\{1,\ldots,k_i\},\ j'\in\{1,\ldots,k_{i'}\}$.

Define $\oplus^i = \bigcup_{j=1}^{k_i} \oplus (\alpha_j^i)$ for every $i \in \{1, \dots, m\}$ and $\oplus = \bigcup_{i=1}^m \oplus^i$. Now, define β_j^i as α_j^i with players $\oplus \setminus \oplus^i$ added as negative literals for every $i \in \{1, \dots, m\}$ and $j \in \{1, \dots, k_i\}$. Since β_j^i , $\beta_{j'}^{i'}$ for $i \neq i'$ cannot be satisfied by the same coalition, we get that the following weighted MC-nets rule without bars ("|") is equivalent to the original one:

$$(\beta_1^1 \to c_1^1) \dots (\beta_i^i \to c_i^i) \dots (\beta_{k_m}^m \to c_{k_m}^m).$$

In each MC-nets rule we added at most S literals, so the size of this rule is $O(S^2)$.

Now, we divide the formula for $k_1 + \cdots + k_m$ separate rules in which only one weight c is non-zero:

$$(\beta_1^1 \to c_1^1)(\beta_2^1 \to 0) \dots (\beta_j^i \to 0) \dots (\beta_{k_m}^m \to 0)$$

$$(\beta_1^1 \to 0)(\beta_2^1 \to c_2^1) \dots (\beta_j^i \to 0) \dots (\beta_{k_m}^m \to 0)$$

$$\dots$$

$$(\beta_1^1 \to 0)(\beta_2^1 \to 0) \dots (\beta_j^i \to 0) \dots (\beta_{k_m}^m \to c_{k_m}^m).$$

The size of this set of rules is $O(S^3)$.

Finally, for every player $p \in N \setminus \oplus$ we add an MC-nets rule $((p) \to 0)$ to every rule. We added at most n MC-nets rules to O(S) rules; hence, the total size of the final set of rules is $O(S^3 + nS)$. This concludes the proof.

B Proof of Lemma 2

Proof. Consider an embedded MC-nets rule:

$$(\alpha_1|\alpha_2,\ldots,\alpha_k)\to c.$$

For this rule we will construct an equivalent hybrid rule.

Assume $\oplus(\alpha_i) \cap \oplus(\alpha_j) \neq \emptyset$ for some $1 \leq i < j \leq k$. If i = 1 or α_i and α_j are not compatible, then the embedded MC-nets rule is contradictory; hence

it is equivalent to any hybrid rule with c=0. If i,j>1 and α_i and α_j are compatible, then α_i and α_j can be replaced by $\alpha_i \wedge \alpha_j$. Hence, in what follows, we assume that $\oplus(\alpha_i)\cap\oplus(\alpha_j)=\emptyset$ for every $i,j\in\{1,\ldots,mk\}$.

Let us define expressions β_1 as α_1 with $\oplus(\alpha_2)\cup\cdots\cup\oplus(\alpha_k)$ added as negative literals. Since β_1 and α_i for any i>1 cannot be now satisfied by the same coalition, we get that the following weighted MC-nets rule is equivalent to the original one:

$$(\beta_1 \to c)(\alpha_2 \to 0) \dots (\alpha_k \to 0).$$

Note that this may not be a hybrid rule, since not all players may appear as positive literals. Hence, we add an MC-nets rule $((p) \to 0)$ for every player $p \in N \setminus \bigcup_{i=1}^k \oplus (\alpha_i)$.

Note that the resulting hybrid rule satisfies condition (*): only MC-nets rules of the form $((p) \to 0)$ are compatible with β_1 and indeed each of them has one positive literal and they are all compatible with each other.

For the original embedded MC-nets rule of size S, the size of the resulting hybrid rule is $O(S^2 + N)$.

Now, consider a hybrid rule satisfying (*):

$$(\alpha_1 \to c)(\alpha_2 \to 0) \dots (\alpha_k \to 0).$$

For this rule we will construct an equivalent embedded MC-nets rule.

Without loss of generality assume expressions $\alpha_{m+1}, \ldots, \alpha_k$ are compatible with α_1 . Fix i > m. We know that $| \oplus (\alpha_i)| = 1$. Assume $\oplus(\alpha_i) \neq \emptyset$ and take player $p \in \ominus(\alpha_i)$. From the definition of hybrid rules we know that every player appears as a positive literal somewhere, i.e., there exists α_j such that $p \in \bigoplus(\alpha_j)$. Hence, we can remove $\neg p$ from α_i and add a player from $\bigoplus(\alpha_i)$ as a negative literal in α_j (unless it is already in $\bigoplus(\alpha_j)$ —in such a case addition can be omitted) without changing the satisfiability of the rule. From (*) we get that α_j is not compatible with α_1 ; hence, condition (*) will still be satisfied.

By doing so for every negative literal of every rule compatible with α_1 (other than α_1), we will obtain a rule of the form:

$$(\alpha_1 \to c)(\beta_2 \to 0) \dots (\beta_m \to 0)((p_1) \to 0) \dots ((p_l) \to 0),$$

in which all β_2, \ldots, β_m are not compatible with α_1 . Hence, this rule is equivalent to an embedded MC-nets rule:

$$(\alpha_1|\beta_2,\ldots,\beta_m)\to c.$$

Note that the size of the resulting embedded MC-nets rule is smaller than the size of the original hybrid rule. This concludes the proof. \Box

C Proof of Lemma 3

Proof. Fix a hybrid rule γ . From Definition 2 we know that $\{\oplus(\alpha_i): 1 \leq i \leq k\}$ is a partition of N. Hence, from Definition 3 $\{l(v_i): v_i \in V\}$ is also a partition

of N. In result, if graph G = (V, E) has labels l such that $\{l(v_i) : v_i \in V\}$ is not a partition of N, then G is not equal to G^{γ} for any γ .

On the other hand, let G = (V, E) be a graph with labels l such that $\{l(v_i) : v_i \in V\}$ is a partition of N. We can now construct a hybrid rule γ as follows: for every node $v_i \in V$ we create a boolean expression α_i as in Eq. (2) with $l(v_i)$ as positive literals and $\bigcup_{v_j:\{v_i,v_j\}\in E} l(v_j)$ as negative literals. Now, it is easy to verify that for $\gamma = (\alpha_1 \to 1)(\alpha_2 \to 0) \dots (\alpha_{|V|} \to 0)$ we have $G = G^{\gamma}$.

D Proof of Lemma 4

Proof. Fix a hybrid rule γ that satisfies condition (*). From Lemma 3 we know that G^{γ} satisfies (i). From condition (*) we get that all expressions compatible with α_1 are compatible with each other and have a single positive literal. Hence, nodes in G^{γ} not adjacent to v_1 form an independent set (ii) and have labels of size 1 (iii).

On the other hand, let G = (V, E) be a graph that satisfies conditions (i)—(iii). To construct a hybrid rule γ that satisfies condition (*) we repeat the construction from the proof of Lemma 3: for every node $v_i \in V$ we create a boolean expression α_i as in Eq. (2) with $l(v_i)$ as positive literals and $\bigcup_{v_j:\{v_i,v_j\}\in E} l(v_j)$ as negative literals. Now, it is easy to verify that for $\gamma = (\alpha_1 \to 1)(\alpha_2 \to 0) \dots (\alpha_{|V|} \to 0)$ we have $G = G^{\gamma}$. Clearly, if v_i and v_j are not adjacent, then α_i and α_j are compatible. Hence, from (ii) and (iii) we get that γ satisfies condition (*).

E Proof of Lemma 5

Proof. Assume $P = \{S_1, \ldots, S_m\}$ satisfies γ . We have that:

- (i) every expression α_i is satisfied by exactly one coalition (it is not possible that $\oplus(\alpha_i)$ is a subset of two non-overlapping coalitions);
- (ii) every coalition S_r satisfies at least one expression (for an arbitrary player $p \in S_r$ we know that there exists an expression α_j such that $p \in \oplus(\alpha_j)$; hence only S_r may satisfy α_j).

Let us define a function f as follows: $f(v_i) = r$ such that $\oplus(\alpha_i) \subseteq S_r$ (from (i) we know there exists exactly one such r). Function f is a proper coloring: if $f(v_i) = f(v_j) = r$, then coalition S_r satisfies both α_i and α_j , hence they are compatible and $\{v_i, v_j\} \notin E$. Also, f is a k-coloring, because $m \leq k$ (from (i) and (ii)).

Now, take k-coloring f of G^{γ} and consider $l(P_f)$. Fix $v_i \in V$ and take $l(S_f^i)$ (i.e., set of all players in nodes colored with the same color as node v_i). We claim $l(S_f^i)$ satisfies α_i . Obviously, $\oplus(\alpha_i) \subseteq l(S_f^i)$. On the other hand, $\ominus(\alpha_i) \cap l(S_f^i) = \emptyset$, because players from $\ominus(\alpha_i)$ all appear in nodes which are neighbors of v_i in graph G^{γ} , hence have different colors than v_i .

So far, we have proved that P satisfies a hybrid rule γ if and only if there exists a k-coloring $f \in C_k(G^{\gamma})$ such that $P = l(P_f)$. Now, (S, P) has non-zero value if and only if P satisfies γ and S satisfies α_1 which—in a partition satisfying γ —is equivalent to containing all players from nodes colored with the same color as node v_1 . This proves Eq. (3).

F Proof of Theorem 6

Proof. From Lemma 1 we know that it is enough to show that MQ-value can be computed in polynomial time for a game represented as a single hybrid rule. Fix a hybrid rule γ . We begin by calculating $MQ_i(g^{\gamma})$ for $i \in l(v_1)$. From Eq. (6) we have:

$$MQ_i(g^{\gamma}) = \sum_{f \in C_2(G^{\gamma})} \frac{(|l(S_f^1)| - 1)!(n - |l(S_f^1)|)!}{2n!}.$$

Let us define a table T[1...n] as follows:

$$T[s] = |\{f \in C_2(G^\gamma) : |l(S_f^1)| = s\}|.$$

To put it in words, for $s \in \{1, ..., n\}$, T[s] is the number of 2-colorings in which there are s players in nodes colored with the same color as node v_1 . We have now:

$$MQ_i(g^{\gamma}) = \sum_{s=1}^{n} \frac{(s-1)!(n-s)!}{2n!} T[s].$$

To compute table T, we begin with basic facts about 2-colorings in a graph. Graph is 2-colorable if and only if it is bipartite, i.e., nodes can be partitioned into two groups $V = V_1 \dot{\cup} V_2$, such that V_1 and V_2 are independent sets. If a bipartite graph is connected (i.e., there exists a path between any two nodes), then there exist a unique such partition. It can be found by performing a breadth-first search from any node, $v \in V$, and putting all nodes at even distance from v in set V_1 and all nodes at odd distance—in set V_2 . Note that creating such a partition and checking whether both groups are independent sets is also a good way of checking whether the graph is 2-colorable. Now, there are two 2-colorings: in the first one nodes from V_1 are colored with color 1, and in the second one—with color 2.

On the other hand, if the graph is not connected, then it is 2-colorable if its every connected component (i.e., maximal subset of nodes such that there exists a path between every pair of nodes) is 2-colorable. In such a case, for each connected component there exists a unique partition into independent sets. However, in the whole graph there may be an exponential number of 2-colorings.

Building upon our analysis, let us now describe the algorithm that computes table T. Assume T is filled with zeros at the beginning. Let A_1, \ldots, A_m be connected components of graph G^{γ} , and let $\{B_j, C_j\}$ be a partition of A_j into two independent sets. Components and partitions can be found by performing

several breadth-first searches. If partition $\{B_j, C_j\}$ does not exist for at least one component A_j , then the calculation is complete (T is a zero table). Assume otherwise and assume $v_1 \in B_1$. We begin by assigning $T[|l(B_1)|] = 2$, since there are two 2-colorings of A_1 . Now, we consider components A_2, \ldots, A_m , one by one, and for each component consider two cases: either B_j or C_j is colored with the same color as node v_1 . Thus, in each step, we update table T by replacing it with a new table T' defined as follows:

$$T'[s] = T[s - |l(B_j)|] + T[s - |l(C_j)|] \text{ for } 1 \le s \le n,$$
(12)

assuming T[s] = 0 for $s \leq 0$. After the *m*-th step, the calculation is complete. Using table T we compute $MQ_i(g^{\gamma})$ in polynomial time.

So far, we have shown how to compute $MQ_i(g^{\gamma})$ for $i \in l(v_1)$. Assume $i \in N \setminus l(v_1)$ and let $u \in V$ be such $i \in l(u)$. From Eq. (6) we have:

$$MQ_{i}(g^{\gamma}) = \sum_{\substack{f \in C_{2}(G^{\gamma}) \\ f(v_{1}) = f(u)}} \frac{(|l(S_{f}^{1})| - 1)!(n - |l(S_{f}^{1})|)!}{2n!} - \sum_{\substack{f \in C_{2}(G^{\gamma}) \\ f(v_{1}) \neq f(u)}} \frac{|l(S_{f}^{1})|!(n - |l(S_{f}^{1})| - 1)!}{2n!}.$$

Now, if $u \in B_1$, then always $f(v_1) = f(u)$ and $MQ_i(g^{\gamma})$ is the same as for players from $l(v_1)$. If $u \in C_1$, then always $f(v_1) \neq f(u)$ and $MQ_i(g^{\gamma}) = -\sum_{s=1}^n \frac{s!(n-s-1)!}{2n!} T[s]$.

Finally, assume $u \in A_j$ for some j > 1. Without loss of generality assume $u \in B_j$. Now, assume table T is split into two parts: $T_{=}$ with colorings such that $f(v_1) = f(u)$ and T_{\neq} with colorings such that $f(v_1) \neq f(u)$. This leads to the following formula:

$$MQ_i(g^{\gamma}) = \sum_{s=1}^n \frac{(s-1)!(n-s)!}{2n!} T_{=}[s] - \sum_{s=1}^n \frac{s!(n-s-1)!}{2n!} T_{\neq}[s].$$

Matrices $T_{=}$ and T_{\neq} can be computed in a number of ways. One way is to assume that component A_j was considered last in the construction of table T (obviously, the result of the construction does not depend on the order of components considered) and reverse step from Eq. (12); then, matrices can be obtained with equations: $T_{=}[s] = T[s - |l(B_j)|]$ and $T_{\neq}[s] = T[s - |l(C_j)|]$ for every $s \in \{1, \ldots, n\}$. Alternatively, matrices can be initiated by assigning $T_{=}[|l(B_1)| + |l(B_j)|] = 2$ and $T_{\neq}[|l(B_1)| + |l(C_j)|] = 2$ and then step from Eq. (12) can be repeated for both matrices for components other than A_1 and A_j . Knowing matrices $T_{=}$ and T_{\neq} we can compute $MQ_i(g^{\gamma})$ in polynomial time. This concludes the proof.

G Proof of Theorem 8

Proof. Let γ be a hybrid rule satisfying condition (*) (from Lemma 2 we know that for every embedded MC-nets rule such equivalent hybrid rule exists).

First, assume $V^* \setminus \{v_1\} \neq \emptyset$, i.e., there exists a node, other than v_1 , with the size of a label larger than one. Clearly, this condition can be checked in polynomial time. From Lemma 4, we know that in G^{γ} nodes not adjacent to v_1 have singleton labels. Hence, this node must be adjacent to v_1 and $V^* \cup \{v_1\}$ does not form an independent set. As we already argued, in such a case the formula evaluates to zero: $EF_i(g^{\gamma}) = 0$ for every $i \in N$.

Assume otherwise that $V^* \setminus \{v_1\} = \emptyset$, i.e., nodes other than v_1 are labeled with singleton sets. We get that $V^* \cup \{v_1\} = \{v_1\}$ and U is the set of nodes not adjacent to node v_1 . From Lemma 4, we get that U is an independent set. Thus, from Eq. (7):

$$EF_i(g^{\gamma}) = \sum_{S \subseteq U} \zeta^i_{l(S \cup \{v_1\})}.$$

Note that $|l(v_1)| = n - k + 1$ and $|l(S \cup \{v_1\})| = |S| + n - k + 1$. Therefore, for $i \in l(v_1)$, we get $EF_i(g^\gamma) = \sum_{s=0}^{|U|} \frac{(s+n-k)!(k-s-1)!}{n!}$. For $i \in l(U)$, we get $EF_i(g^\gamma) = \sum_{s=1}^{|U|} \frac{(s+n-k)!(k-s-1)!}{n!} - \sum_{s=0}^{|U|-1} \frac{(s+n-k+1)!(k-s-2)!}{n!} = 0$. Finally, for $i \in N \setminus (l(v_1) \cup l(U))$, we get: $EF_i(g^\gamma) = -\sum_{s=0}^{|U|} \frac{(s+n-k+1)!(k-s-2)!}{n!}$. Each of this sum can be computed in polynomial time. This concludes the proof. \square

H Proof of Theorem 9

Proof. The value $n!EF_i(g^{\gamma})$ can be considered as the number of accepting paths of nondeterministic Turing machine, so the problem is in #P. To show that the problem is #P-hard, we use the (Turing) reduction from the problem of counting all independent sets in a graph which is #P-complete (Valiant, 1979).

Let G = (V, E) be an arbitrary graph with $V = \{v_2, \dots, v_{k+1}\}$ for notational convenience. Let $I_m(G)$ be the set of independent sets of size m in graph G. We will determine $|I_m(G)|$ for every $m \in \{0, \dots, k\}$.

To this end, let us construct k+1 graphs: for $j \in \{0, ..., k\}$, construct graph $G^{\gamma_j} = (V \cup \{v_1\}, E)$ with $l(v_i) = \{i\}$ for $v_i \in V$ and $l(v_1) = \{1\} \cup \{k+2, ..., k+j+1\}$ (based on Lemma 3 we know that for every graph G there exists hybrid rule γ such that $G = G^{\gamma}$). Note that $V^* \cup \{v_1\} = \{v_1\}$ and U = V. Hence, Eq. (7) for i = 1 and graph G^{γ_j} simplifies to:

$$EF_1(g^{\gamma_j}) = \sum_{m=0}^k \frac{(m+j)!(k-m)!}{(k+j+1)!} |I_m(G)|.$$

This system of linear equations is equivalent to the following matrix form:

$$\begin{bmatrix} 0!k! & 1!(k-1)! & \dots & k!0! \\ 1!k! & 2!(k-1)! & \dots & (k+1)!0! \\ \vdots & \vdots & \ddots & \vdots \\ k!k! & (k+1)!(k-1)! & \dots & (2k)!0! \end{bmatrix} \cdot \begin{bmatrix} |I_0(G)| \\ |I_1(G)| \\ \vdots \\ |I_k(G)| \end{bmatrix} = \begin{bmatrix} (k+1)!EF_1(g^{\gamma_0}) \\ (k+2)!EF_1(g^{\gamma_1}) \\ \vdots \\ (2k+1)!EF_1(g^{\gamma_k}) \end{bmatrix}$$

From (Bacher, 2002, Theorem 1.1), we know that the determinant of matrix $A = ((i+j)!)_{0 \le i,j \le k}$ equals $\prod_{i=0}^k (i!)^2$; hence, the determinant of the square matrix is $\prod_{i=0}^k (i!)^3$ (columns of A are multiplied by $k!, \ldots, 0!$). Since the determinant is non-zero, the matrix is invertible and knowing $EF_1(g^{\gamma_0}), \ldots, EF_1(g^{\gamma_k})$ allows us to find IS_0, \ldots, IS_k in polynomial time using Gaussian elimination. \square

I Proof of Lemma 11

Proof. Take r first elements from n+r and consider all their possible partitions. There are $\{r,i\}$ partitions of r elements into i subsets which implies there are $\{r,i\}B_{n,i}$ partitions of n+r elements in which first r elements form i subsets. Summing over all $i \in \{1,\ldots,r\}$ concludes the proof.

J Proof of Lemma 12

Proof. Matrix B looks as follows

$$B = \begin{bmatrix} B_{1,1} & B_{1,2} & \dots & B_{1,k} \\ B_{2,1} & B_{2,2} & \dots & B_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ B_{k,1} & B_{k,2} & \dots & B_{k,k} \end{bmatrix}$$

Consider matrix:

$$C = \begin{bmatrix} \{1,1\} & \{2,1\} & \{3,1\} & \dots & \{k,1\} \\ 0 & \{2,2\} & \{3,2\} & \dots & \{k,2\} \\ 0 & 0 & \{3,3\} & \dots & \{k,3\} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \{k,k\} \end{bmatrix}$$

Since $\{i, i\} = 1$ for every $i \in \mathbb{N}$, matrix C is triangular with diagonal 1, so $\det(C) = 1$. Consider the product $B \cdot C$. Multiplying by C is equivalent to adding to column j columns $1, 2, \ldots, j-1$ with weights $\{j, 1\}, \{j, 2\}, \ldots, \{j, j-1\}$. Consider field (i, j) in the new matrix. From Lemma 11, we know that:

$$B_{i,1} \cdot \{j,1\} + \cdots + B_{i,j} \cdot \{j,j\} = B_{i+j}$$
.

Hence, $B \cdot C = (B_{i+j})_{1 \leq i,j \leq k}$. Now, from (Aigner, 1999, Remark 2) we get that the determinant of matrix $(B_{i+j})_{1 \leq i,j \leq k}$ is $(\sum_{i=0}^k k!/(k-i)!) \cdot (\prod_{i=0}^{k-1} i!)$ which is

equivalent to $(\prod_{i=0}^k i!) \cdot (\sum_{i=0}^k 1/i!)$. Hence, it is also the determinant of matrix B.

K Proof of Theorem 10

Proof. The value $\mathcal{P} \cdot HY_i(g^{\gamma})$ can be considered as the number of accepting paths of nondeterministic Turing machine, so the problem is in #P. To show that the problem is #P-hard, we use the (Turing) reduction from the *chromatic polynomial problem*, i.e., counting *m*-colorings in a graph, which is #P-complete (Jaeger et al., 1990).

Let G = (V, E) be an arbitrary graph with $V = \{v_2, \ldots, v_{k+1}\}$ for notational convenience and m be an arbitrary number. The task is to determine $|C_m(G)|$. Let c_i be the number of k-colorings that uses exactly i colors. We will determine c_i for every $i \in \{1, \ldots, k\}$. From these values it is easy to compute the number of m-colorings with the following formula:

$$|C_m(G)| = \sum_{i=1}^k \frac{m(m-1)\cdots(m-i+1)}{k(k-1)\cdots(k-i+1)} \cdot c_i.$$

To this end, let us construct k graphs: for $j \in \{1, \ldots, k\}$, construct graph $G^{\gamma_j} = (V \cup \{v_1\}, E \cup \{\{v_1, v_i\} : v_i \in V) \text{ with } l(v_i) = \{i\} \text{ for } v_i \in V \text{ and } l(v_1) = \{1\} \cup \{k+2, \ldots, k+j\}.$ Since node v_1 is connected to all other nodes we know that G^{γ_j} satisfies condition from Lemma 4, i.e., there exists hybrid rule γ_j equivalent to some embedded MC-nets rule such that G^{γ_j} is the corresponding graph. Note that $S_f^1 = \{1\}$ for every coloring f. Also, the number of (k+1)-colorings of graph G^{γ_j} that uses m+1 colors is equal to (k+1) (color of node v_1) times the number of k-colorings of graph G that uses m colors:

$$|\{f \in C_{k+1}(G^{\gamma}) : |P_f| = m+1\}| = (k+1) \cdot c_m.$$

Hence, Eq. (8) for i = 1 and graph G^{γ_j} yields:

$$HY_{i}(g^{\gamma_{j}}) = \frac{(j-1)!k!}{(k+j)!B_{k+j}} \sum_{f \in C_{k+1}(G^{\gamma})} \frac{B_{j,|l(P_{f})|-1}}{\theta(f)}$$

$$= \frac{(j-1)!k!}{(k+j)!B_{k+j}} \sum_{m=2}^{k+1} \sum_{f \in C_{k+1}(G^{\gamma}):|P_{f}|=m} \frac{B_{j,(m-1)}}{\binom{k+1}{m}m!}$$

$$= \frac{(j-1)!k! \cdot (k+1)}{(k+j)!B_{k+j}} \sum_{m=2}^{k+1} \frac{B_{j,(m-1)}}{\binom{k+1}{m}m!} c_{m-1}$$

$$= \frac{(j-1)!(k+1)!}{(k+j)!B_{k+j}} \sum_{m=1}^{k} \frac{(k-m)!B_{j,m}}{(k+1)!} c_{m}$$

$$= \frac{(j-1)!}{(k+j)!B_{k+j}} \sum_{m=1}^{k} (k-m)!B_{j,m} c_{m}.$$

This system of linear equations can be presented in the matrix form as follows:

$$\begin{bmatrix} (k-1)!B_{1,1} & (k-2)!B_{1,2} & \dots & 0!B_{1,k} \\ (k-1)!B_{2,1} & (k-2)!B_{2,2} & \dots & 0!B_{2,k} \\ \vdots & \vdots & \ddots & \vdots \\ (k-1)!B_{k,1} & (k-2)!B_{k,2} & \dots & 0!B_{k,k} \end{bmatrix} \cdot \begin{bmatrix} c_1 \\ c_2 \\ \vdots \\ c_k \end{bmatrix} = \begin{bmatrix} \frac{(k+1)!}{0!}B_{k+1}HY_1(g^{\gamma_1}) \\ \frac{(k+2)!}{1!}B_{k+2}HY_1(g^{\gamma_2}) \\ \vdots \\ \frac{(2k)!}{(k-1)!}B_{2k}HY_1(g^{\gamma_k}) \end{bmatrix}$$

From Lemma 12 we get that the determinant of the square matrix equals $(\prod_{i=0}^k i!) \cdot (\sum_{i=0}^k 1/i!)$ multiplied by $(\prod_{i=0}^{k-1} i!)$. Since the determinant is nonzero, the matrix is invertible and knowing $HY_1(g^{\gamma_1}), \ldots, HY_1(g^{\gamma_k})$ allows us to find c_1, \ldots, c_k in polynomial time using Gaussian elimination. This concludes the proof.

L Proof of Theorem 13

Proof. The value $n!SS_i(g^{\gamma})$ can be considered as the number of accepting paths of nondeterministic Turing machine, so the problem is in #P. To show that the problem is #P-hard, we use a reduction from the problem of counting all matching in a bipartite graph which is #P-complete (Valiant, 1979).

Let G = (V, E) be an arbitrary bipartite graph with $V = \{v_2, \ldots, v_k\}$ for notational convenience. Let h_G be the number of all matchings in G (called Hosoya index). Our goal is to determine $\underline{h_G}$.

To this end, let us construct a graph $\overline{G^{\gamma}} = (V \cup \{v_1\}, E)$ with $l(v_i) = \{v_i\}$ for $v_i \in V \cup \{v_1\}$. Consider a complement of graph $\overline{G^{\gamma}}$, denoted by G^{γ} . Since node v_1 does not have any edges in $\overline{G^{\gamma}}$, then it is connected to all nodes in G^{γ} ; hence, G^{γ} satisfies conditions of Lemma 4 and there exist a hybrid rule γ equivalent to some embedded MC-nets rule such that G^{γ} is the corresponding graph.

Now, consider $SS_1(g^{\gamma})$. From Eq. (10) we know that $SS_1(g^{\gamma})$ is a weighted sum over clique covers, i.e., partitions of nodes in $\overline{G^{\gamma}}$ into cliques. However, since the graph is bipartite, there exist no clique with more than 2 nodes. Hence, for every such a partition, P, we have $\prod_{T \in P} (|T| - 1)! = 1$. Thus, from Eq. (10) we get:

$$SS_1(g^{\gamma}) = |QC(G^{\gamma})|/n!.$$

Now, it remains to observe that every partition of nodes into cliques corresponds to exactly one matching. This concludes the proof. \Box

M Proof of Theorem 14

Proof. From the process approach interpretation of MY-value (Skibski et al., 2018) we get that $n!MY_i(g^{\gamma})$ can be considered as the number of accepting paths of nondeterministic Turing machine, so the problem is in #P. To show that the problem is #P-hard, again we use the reduction from the problem

of counting all matching in a bipartite graph which is #P-complete (Valiant, 1979).

With the same reasoning as in SS-value, instead of considering colorings in graph G^{γ} we will focus on partitions of a graph into cliques in the complement graph $\overline{G^{\gamma}}$. Then, Eq. (10) can be rewritten as follows:

$$\Delta_i^j(g^{\gamma}) \sum_{P \in QC(\overline{G^{\gamma}})} \frac{(-1)^{|P|}(|P|-2)!}{n - |l(T)|}.$$
 (13)

Let G = (V, E) be an arbitrary bipartite graph with $V = \{v_3, \ldots, v_{k+2}\}$ for notational convenience. Let h_G^m be the number of all matchings in G of size k - m; note that in such a case m is the number of pairs plus the number of unmatched nodes. We will determine h_G^m for every $m \in \{1, \ldots, k\}$. The sum $\sum_{m=1}^k h_G^m$ is the number of all matchings in G.

 $\sum_{m=1}^k h_G^m \text{ is the number of all matchings in } G.$ To this end, let us construct k graphs: for $j \in \{1,\ldots,k\}$, construct graph $\overline{G^{\gamma_j}} = (V \cup U, E)$ where $U = \{v_1, v_2, v_{k+3}, \ldots, v_{k+j+2}\}$ and $l(v_i) = \{i\}$ for $v_i \in (V \cup U) \setminus \{v_1\}$ and $l(v_1) = \{1, k+j+3, \ldots, 3k+1\}$. Consider a complement of graph $\overline{G^{\gamma_j}}$, denoted by G^{γ_j} . Since node v_1 does not have any edges in $\overline{G^{\gamma_j}}$, then it is adjacent to all nodes in G^{γ_j} ; hence, G^{γ_j} satisfies conditions from Lemma 4, i.e., there exists hybrid rule γ_j equivalent to some embedded MC-nets rule such that G^{γ_j} is the corresponding graph.

Now, consider $\Delta_1^2(g^{\gamma_j})$. Note that $j \in l(v_2)$, hence $T = \{v_2\}$ and $|l(T)| = |\{2\}| = 1$. From Eq. (13) for game g^{γ_j} we get that:

$$\Delta_1^2(g^{\gamma_j}) = \frac{1}{3k} \sum_{m=1}^k (-1)^{m+j} (m+j)! \cdot h_G^m.$$

This system of linear equations can be presented in the matrix form as follows:

$$\begin{bmatrix} 2! & -3! & \dots & \pm (k+1)! \\ -3! & 4! & \dots & \mp (k+2)! \\ \vdots & \vdots & \ddots & \vdots \\ \pm (k+1)! & \mp (k+2)! & \dots & (2k)! \end{bmatrix} \cdot \begin{bmatrix} h_G^1 \\ h_G^2 \\ \vdots \\ h_G^k \end{bmatrix} = \begin{bmatrix} 3k \cdot \Delta_1^2(g^{\gamma_1}) \\ 3k \cdot \Delta_1^2(g^{\gamma_1}) \\ \vdots \\ 3k \cdot \Delta_1^2(g^{\gamma_1}) \end{bmatrix}$$

Note that by multiplying even rows and then even columns by (-1) we can transform the square matrix into matrix $A=((i+j)!)_{1\leq i,j\leq k}$. Moreover, since there are as many even rows as even columns, the determinant of the original matrix is the same as the determinant of A. Now, from (Bacher, 2002, Theorem 1.1), we know that the determinant of A equals $\prod_{i=0}^{k-1}(i!)(i+2)!$; hence, the determinant of the original matrix is the same. Since the determinant is non-zero, the matrix is invertible and knowing $\Delta_1^2(g^{\gamma_1}), \ldots, \Delta_1^2(g^{\gamma_k})$ (i.e., $MY_1(g^{\gamma_1}), MY_2(g^{\gamma_1}), \ldots, MY_1(g^{\gamma_k}), MY_2(g^{\gamma_k})$) allows us to find h_G^1, \ldots, h_G^k in polynomial time using Gaussian elimination.