

The Tractability of the Shapley Value over Bounded Treewidth Matching Games

Gianluigi Greco, Francesco Lupia, and Francesco Scarcello

University of Calabria, Italy

ggreco@mat.unical.it, {lupia,scarcello}@dimes.unical.it

Abstract

Matching games form a class of coalitional games that attracted much attention in the literature. Indeed, several results are known about the complexity of computing over them solution concepts. In particular, it is known that computing the *Shapley value* is intractable in general, formally #P-hard, and feasible in polynomial time over games defined on trees. In fact, it was an open problem whether or not this tractability result holds over classes of graphs properly including acyclic ones. The main contribution of the paper is to provide a positive answer to this question, by showing that the Shapley value is tractable for matching games defined over graphs having *bounded treewidth*. The proposed polynomial-time algorithm has been implemented and tested on classes of graphs having different sizes and treewidth up to three.

1 Introduction

Coalitional games are mathematical models suited to study value distribution problems among rational agents that can obtain higher payoffs by collaborating with each other rather than by acting in isolation [Osborne and Rubinstein, 1994].

In the paper, we focus on the problem of computing the *Shapley value* [Shapley, 1953] in *matching games*, a class of coalitional games where the value that agents guarantee to themselves is defined in terms of *matchings* over an underlying graph. In particular, let $G=(N, E)$ be a graph whose set N of nodes is transparently viewed as a given set of agents. Then, for each coalition $C \subseteq N$, the value $v_G(C)$ that agents in C achieve by collaborating is given by the cardinality of a maximum matching¹ in the subgraph induced by C .

Example 1.1. Consider the graph $G = (N, E)$ in Figure 1, with $N = \{a, b, c, d, i\}$ being the set of its nodes. A matching in this graph is $M = \{\{a, i\}, \{b, c\}\}$, and is depicted in bold. Note that there is no matching including more than two edges, so that $v_G(N) = 2$. The figure also reports the subgraphs induced by the nodes in $\{a, d, c, i\}$ and

¹A generalization of these games can be defined by equipping graphs with edge-weighting functions and by looking for *weighted* matchings. This extension is not considered in the paper.

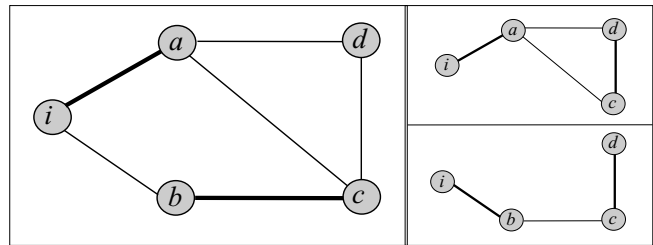


Figure 1: The graph G and the matching M in Example 1.1 (left). The subgraphs induced by two coalitions (right).

$\{b, c, d, i\}$. Again, maximum matchings consisting of two edges are evidenced. Therefore, we can immediately conclude that $v_G(\{a, d, c, i\}) = v_G(\{b, c, d, i\}) = 2$. \triangleleft

Matching games have attracted much attention in the literature, due to their wide spectrum of applicability. For instance, when restricted to bipartite graphs, they go under the name of *assignment games* and form a natural model to study distribution problems in matching markets (see, e.g., [Roth and Sotomayor, 1990; Shapley and Shubik, 1971]). Several results are already known in the literature about the complexity of computing over them *solution concepts*, that is, value distributions embodying desirable fairness and stability requirements. For instance, results have been derived for the core [Deng *et al.*, 1999; Alkan and Gale, 1990; Chalkiadakis *et al.*, 2016], the least core [Kern and Paulusma, 2003], and the nucleolus [Biró *et al.*, 2011; Kern and Paulusma, 2003; Chen *et al.*, 2012; Greco *et al.*, 2014].

Aziz and de Keijzer [2014] observed that computing the Shapley value is generally intractable over matching games, formally it is #P-complete. Moreover, they showed that it is computable in polynomial time when the maximum degree of the underlying graphs is two, and they asked whether this result can be extended to a non-trivial class of graphs of degree at least three. The question has been then addressed by Bousquet [2015], who showed that the Shapley value over trees can be computed in polynomial time, and left open the problem of assessing whether the result can be extended to larger classes of structures, such as the graphs having bounded *treewidth* [Robertson and Seymour, 1984].

In the paper, we provide a positive answer to this question left open in the literature. The result is established via a

polynomial-time algorithm whose design exploits: (i) some useful characterizations we derive for certain properties of matchings, (ii) an encoding for such properties in terms of monadic second order logic (MSO) formulas, and (iii) known tractability results for MSO formulas over graphs having bounded treewidth. As a further (more practical) contribution, the algorithm has been implemented and its scalability has been experimentally validated.

Notably, our analysis and result also apply to the *Banzhaf value* [Banzhaf, 1965], a closely-related solution concept.

2 Preliminaries

Matching Games. Let $G = (N, E)$ be a graph, where N and E are the sets of its nodes and edges, respectively. Any edge $e \in E$ is a subset of nodes with $|e| = 2$. Recall that a set $M \subseteq E$ is a *matching* (in G) if $e_1 \cap e_2 = \emptyset$, for each pair of distinct edges $e_1, e_2 \in M$. We say that M is *maximum* if $|M| \geq |M'|$, for each matching M' . The cardinality of a maximum matching in G is denoted by $\max(G)$.

The *matching game* associated with the graph $G = (N, E)$ is the tuple $\mathcal{G}_G = \langle N, v_G \rangle$, where nodes in N are transparently viewed as agents. Moreover, v_G is the function associating with each *coalition* $C \subseteq N$ the value $v_G(C) = \max(G[C])$, with $G[C]$ being the subgraph induced by the nodes in C .

In the paper, we are interested in the computation of *solution concepts* for matching games, i.e., of ways of dividing the total value $v_G(N)$ over each agent $i \in N$. In particular, we focus on the *Shapley value*, which is a well-known solution concept such that the payoff associated with each agent $i \in N$ is

$$\phi_i(\mathcal{G}_G) = \sum_{C \subseteq N \setminus \{i\}} \frac{|C|!(|N| - |C| - 1)!}{N!} (v_G(C \cup \{i\}) - v_G(C)).$$

Treewidth. A *tree decomposition* of a graph $G = (N, E)$ is a pair $\langle T, \chi \rangle$, where T is a tree, and χ is a labeling function assigning to each vertex p in T a set of nodes $\chi(p) \subseteq N$, such that the following conditions are satisfied: (1) for each node $x \in N$, there exists p in T such that $x \in \chi(p)$; (2) for each edge $\{x, y\} \in E$, there exists p in T such that $\{x, y\} \subseteq \chi(p)$; and, (3) for each node $x \in N$, the subgraph of T induced by all nodes p such that $x \in \chi(p)$ is connected.

The *width* of $\langle T, \chi \rangle$ is the number $\max_{p \in T} (|\chi(p)| - 1)$. The *treewidth* of G , denoted by $tw(G)$, is the minimum width over all its tree decompositions. Treewidth is a true generalization of acyclicity: G is acyclic if, and only if, $tw(G) = 1$. A class of graphs has bounded treewidth if there is some k with $tw(G) \leq k$, for every graph G in the class (see [Gottlob et al., 2016] for a recent survey on decomposition methods).

Example 2.1. Consider the graph G in Figure 1. We have $tw(G) > 1$, for instance because of the cycle over the nodes a, c , and d . In fact, $tw(G) = 2$ holds, as it is witnessed by the tree decomposition reported in Figure 2. \triangleleft

3 Useful Properties on Maximum Matchings

The first ingredient for showing tractability over bounded treewidth matching games is to derive some properties of maximum matchings that simplify the subsequent analysis.

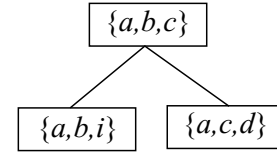


Figure 2: A width-2 tree decomposition of G .

Let M be a matching in a graph $G = (N, E)$. A node $x \in N$ is said *M -matched* if there is some edge $e \in M$ with $x \in e$; otherwise, x is *M -exposed*. A path is a sequence of distinct nodes π_1, \dots, π_m , with $m > 1$, such that $\{\pi_j, \pi_{j+1}\} \in E$, for each $j \in \{1, \dots, m-1\}$. The path is said *M -alternating* if each pair of adjacent edges satisfy $\{\pi_j, \pi_{j+1}\} \in M$ and $\{\pi_{j+1}, \pi_{j+2}\} \notin M$, or vice-versa ($j \in \{1, \dots, m-2\}$). Moreover, the path is *M -augmenting* if it is M -alternating, and π_1 and π_m are M -exposed; in fact, they are the only M -exposed nodes occurring in the path.

Note that M is a maximum matching if, and only if, there is no M -augmenting path [Berge, 1957].

Example 3.1. Consider again the graph G and the matching M of Figure 1. Note that d is M -exposed and, for instance, that i, a, d is M -alternating. However, i is M -matched and thus the path is not M -augmenting. In fact, there is no M -augmenting path, as M is a maximum matching. \triangleleft

The following definition introduces a crucial notion for analyzing matching games.

Definition 3.2. A node $x \in N$ is *blocked* by a matching M if

- x is M -exposed, and
- for each M -alternating path π_1, \dots, π_m with $\pi_1 = x$, it holds that π_m is M -matched.

Example 3.3. Consider node i in our running example. It is easy to check that i is not blocked by the matching $M = \{\{a, i\}, \{b, c\}\}$. Moreover, note that i is not blocked by the matching $\{\{b, c\}\}$, while it is blocked by $\{\{a, d\}, \{b, c\}\}$. \triangleleft

Actually, since any single edge is an alternating path, the following is an equivalent characterization of Definition 3.2.

Fact 3.4. A node $x \in N$ is *blocked* by M if, and only if, for each M -alternating path π_1, \dots, π_m with $\pi_1 = x$, it holds that π_m is M -matched and $\{\pi_1, \pi_m\} \notin M$.

We next show that, for this notion, it is not necessary to focus on maximum matchings.

Lemma 3.5. A node $i \in N$ is *blocked* by some maximum matching if, and only if, it is *blocked* by some matching.

Proof Sketch. Assume that i is blocked by a matching M_1 for which there exists an M_1 -augmenting path π'_1, \dots, π'_h . Consider the matching M_2 obtained from M_1 by removing all edges traversed by π'_1, \dots, π'_h and by adding all edges in $E \setminus M_1$ traversed by π'_1, \dots, π'_h . That is, M_2 is the matching obtained by “augmenting” M_1 via the path π'_1, \dots, π'_h . Clearly, $|M_2| > |M_1|$. Moreover, note that each node π'_j , with $j \in \{1, \dots, h\}$, is not M_2 -exposed. Now, let $\bar{\pi}_1, \dots, \bar{\pi}_q$ be any M_2 -alternating path with $\bar{\pi}_1 = i$. We shall show that the node $\bar{\pi}_q$ is not M_2 -exposed and $\{\bar{\pi}_1, \bar{\pi}_q\} \notin M_2$.

Claim. $\bar{\pi}_1, \dots, \bar{\pi}_q$ and π'_1, \dots, π'_h do not share any edge. By contradiction, consider the smallest index $k \in \{1, \dots, q-1\}$ and an index $j \in \{1, \dots, h-1\}$ such that $\pi'_j = \bar{\pi}_k$ and $\pi'_{j+1} = \bar{\pi}_{k+1}$. Note that if $k = 1$, then the path $\bar{\pi}_k, \pi'_{j+1}, \dots, \pi'_h$ would witness that i is not blocked by M_1 , which is impossible. Indeed, we would have that $\bar{\pi}_k = i$ and that $\bar{\pi}_k, \pi'_{j+1}, \dots, \pi'_h$ is M_1 -alternating, with π'_h being M_1 -exposed (for this latter property, just recall that π'_1, \dots, π'_h is M_1 -augmenting). This means that we necessarily have $k > 1$, and that $\bar{\pi}_1, \dots, \bar{\pi}_k$ is an M_1 -alternating path. Now, in the case where $j = 1$ holds, we have $\bar{\pi}_k = \pi'_1$. Therefore, the path $\bar{\pi}_1, \dots, \bar{\pi}_k$ would again witness that i is not blocked by M_1 , which is impossible. So, we have $j > 1$. Hence, the node $\bar{\pi}_k = \pi'_j$ is adjacent to the nodes $\bar{\pi}_{k-1}, \pi'_{j-1}$, and $\bar{\pi}_{k+1} = \pi'_{j+1}$. Recall that π'_1, \dots, π'_h is M_1 -alternating and assume first that the edge $\{\pi'_j, \pi'_{j-1}\}$ is in M_1 . Then, $\{\bar{\pi}_{k-1}, \pi'_j\} \notin M_1$ because $\pi'_j (= \bar{\pi}_k)$ is matched to someone else. Because M_1 and M_2 are equal for edges in the path $\bar{\pi}_1, \dots, \bar{\pi}_k$ below k , it follows that $\bar{\pi}_1, \dots, \bar{\pi}_{k-1}, \pi'_j, \pi'_{j-1}, \dots, \pi'_1$ is M_1 -alternating. The same holds if we assume instead that the edge $\{\pi'_j, \pi'_{j+1}\}$ is not in M_1 . Indeed, in this case $\{\pi'_j, \pi'_{j+1}\} \in M_1$ and thus, by definition of M_2 , $\{\bar{\pi}_k, \bar{\pi}_{k+1}\} \notin M_2$ and $\{\bar{\pi}_{k-1}, \bar{\pi}_k\} \in M_2$ by alternation, that is, $\{\bar{\pi}_{k-1}, \pi'_j\} \in M_1$, because the two matchings are equal below k . Again, we get that $\bar{\pi}_1, \dots, \bar{\pi}_{k-1}, \pi'_j, \pi'_{j-1}, \dots, \pi'_1$ is M_1 -alternating. Since π'_1 is M_1 -exposed, this contradicts that i is blocked by M_1 . \diamond

Given the above claim, we derive that $\bar{\pi}_1, \dots, \bar{\pi}_q$ is also M_1 -alternating; hence, $\{\bar{\pi}_1, \bar{\pi}_q\} \notin M_1$. It then follows that $\{\bar{\pi}_1, \bar{\pi}_q\} \notin M_2$, because we have already observed that $\bar{\pi}_1 = i$ cannot occur in the path π'_1, \dots, π'_h and thus cannot become M_2 -matched. To conclude, we have then to show that $\bar{\pi}_q$ is M_2 -matched. This is trivial if $\bar{\pi}_q \in \{\pi'_1, \dots, \pi'_h\}$. Otherwise, if $\bar{\pi}_q \notin \{\pi'_1, \dots, \pi'_h\}$, then $\bar{\pi}_q$ is M_2 -matched because it is M_1 -matched and it does not occur in the path where the two matchings may differ.

Now, either M_2 is a maximum matching, or we can repeat the above construction, until we eventually get a sequence M_1, M_2, \dots, M_ℓ of matchings such that i is blocked by each of them and $|M_1| < |M_2| < \dots < |M_\ell|$, with the latter being a maximum matching of the given graph. \square

Example 3.6. Recall from Example 3.3 that node i is blocked by the matching $\{\{a, d\}, \{b, c\}\}$. This is actually a maximum matching. In fact, it can be checked that $\{\{a, d\}, \{b, c\}\}$ is the only matching blocking i . \triangleleft

4 Characterization of Marginal Contribution

Our second step is to relate the fact that an agent i is blocked (cf. Definition 3.2) with the value of the quantity $v_G(C \cup \{i\}) - v_G(C)$, for each given $C \subseteq N$, occurring in the definition of the Shapley value. Note that this value is known as the *marginal contribution* of agent i to coalition $C \cup \{i\}$.

Assume hereinafter that $C \subseteq N$ is a coalition and let i be a node in $N \setminus C$. We show that the condition of i being blocked by some matching in the subgraph $G[C \cup \{i\}]$ precisely determines the marginal contribution of i to $C \cup \{i\}$ —note in the proof the role played by Lemma 3.5.

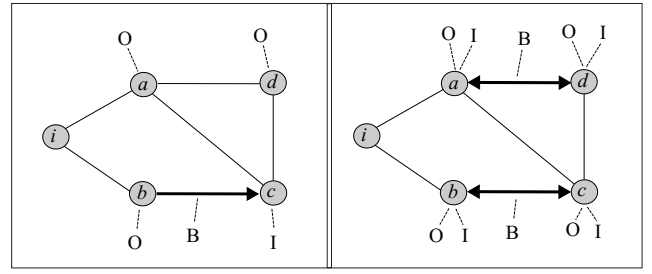


Figure 3: A matching that does not block node i (left). A maximum matching that blocks i (right).

Lemma 4.1. *The following statements are equivalent:*

- (1) $v_G(C \cup \{i\}) - v_G(C) = 0$;
- (2) i is blocked by some matching M in $G[C \cup \{i\}]$.

Proof. (1) \Rightarrow (2) Let M be a maximum matching in $G[C]$. Since $v_G(C \cup \{i\}) - v_G(C) = 0$, M is also a maximum matching in $G[C \cup \{i\}]$. Consider, then, any M -alternating path π_1, \dots, π_m in $G[C \cup \{i\}]$ with $\pi_1 = i$. Since M is a matching in $G[C]$ and $i \notin C$, then π_1 is clearly M -exposed in $G[C \cup \{i\}]$. In particular, $\{\pi_1, \pi_m\} \notin M$. Assume now, by contradiction, that π_m is M -exposed, too. Then, π_1, \dots, π_m is M -augmenting in $G[C \cup \{i\}]$, which is impossible.

(2) \Rightarrow (1) Because of Lemma 3.5, we can assume that M is a maximum matching in $G[C \cup \{i\}]$. Because i is blocked by M , we know that i must be M -exposed. Then, M is also a maximum matching in $G[C]$ and we have concluded, since $v_G(C \cup \{i\}) - v_G(C) = 0$ holds. \square

We continue the chain of equivalences of Lemma 4.1, by providing a number of conditions that can easily be checked.

In the following, given the graph $G = (N, E)$, we define r_E as the set of pairs $\bigcup_{\{x,y\} \in E} \{(x, y), (y, x)\}$.

Lemma 4.2. *The following statements are equivalent:*

- (2) i is blocked by some matching M in $G[C \cup \{i\}]$;
- (3) There are sets $B \subseteq r_E$, $I \subseteq N$, and $O \subseteq N$ satisfying on $G[C \cup \{i\}]$ the conditions (a), ..., (f) of Figure 4.

Before providing the proof of the result, consider the coalition $C = \{a, b, c, d\}$ and Figure 3, which shows two examples of sets B , I , and O . Intuitively, B encodes some matching, whose edges are however oriented according to the way they are traversed along some alternating path starting at node i . Any edge can actually be traversed in both directions. For instance, for the matching $\{\{a, d\}, \{b, c\}\}$, the paths i, a, d, c, b and i, b, c, d, a show that the two edges $\{a, d\}$ and $\{b, c\}$ can be traversed in both directions.

Note that the sets on the left part of Figure 3, with $B = \{(b, c)\}$ violate Condition (d). Instead, the sets on the right part of Figure 3, with $B = \{(a, d), (b, c), (d, a), (c, b)\}$ satisfy all conditions and, by Lemma 4.2, they witness that i is blocked by some matching (in fact, precisely by the maximum matching $\{\{a, d\}, \{b, c\}\}$).

Proof Sketch of Lemma 4.2. (2) \Rightarrow (3) Assume that i is blocked by some matching M in $G[C \cup \{i\}]$, and consider

- (a) $i \notin I$;
The distinguished node i does not belong to the set I .
- (b) $\forall x, y \in C \cup \{i\}, ((x, y) \in B) \rightarrow (y \in I)$;
The set B is meant to encode all the edges of the matching M in $G[C \cup \{i\}]$ (cf. Lemma 4.2.(2)), by adding an orientation to them. The condition states that for each element $(x, y) \in B$, the destination node y belongs to I .
- (c) $\forall x \in C \cup \{i\}, \forall y \in C, ((x, y) \in r_E \wedge (x \in I)) \rightarrow (y \in O)$;
If the source node of an edge in $G[C \cup \{i\}]$ belongs to I , then the destination node must belong to O .
- (d) $\forall x \in C \cup \{i\}, x \in O \rightarrow (\exists z \in C \text{ such that } (x, z) \in B)$;
Every node $x \in O$ must have an outgoing edge that belongs to B .
- (e) $\forall x, y, y' \in C \cup \{i\}, (((x, y) \in B \wedge (x, y') \in B) \vee ((y, x) \in B \wedge (y', x) \in B) \vee ((y, x) \in B \wedge (x, y') \in B)) \rightarrow (y = y')$;
By looking at the undirected version of B , every node cannot have more than one incident edge.
- (f) $\forall x \in C, (i, x) \in r_E \rightarrow ((i, x) \notin B) \wedge (x \in O)$;
For the distinguished node i , every outgoing edge does not belong to B and the destination node belongs to O .

 Figure 4: Properties in Lemma 4.2.(3) that must hold on sets $B \subseteq r_E$, $I \subseteq N$, and $O \subseteq N$.

the sets B , I , and O built as follows. For each edge $\{x, y\}$ in $G[C \cup \{i\}]$ if there is an M -alternating path π_1, \dots, π_m in this subgraph with $\pi_1 = i$ and such that $x = \pi_j$ and $y = \pi_{j+1}$, for some index $j \in \{1, \dots, m-1\}$, and $\{x, y\}$ is in M , then (x, y) is in B , $x \in O$, and $y \in I$. For each edge $\{i, y\} \in E \setminus M$, we set $y \in O$. No other element is in B , I , or O . Note that the sets trivially satisfy (a), (b), and (e).

Concerning (c), let π_1, \dots, π_m be an M -alternating path in $G[C \cup \{i\}]$ with $\pi_1 = i$ such that, for some index $j \in \{2, \dots, m\}$, $\pi_j = x$ with $x \in I$ and $\{\pi_{j-1}, \pi_j\} \in M$ hold. Note that $\pi_{j-1} \in O$. Then, assume that $\{x, y\}$ is in E with $y \in C$ and $y \neq \pi_{j-1}$. We have that $\{x, y\} \notin M$ and π_1, \dots, π_j, y is M -alternating. Since i is blocked by M , y is M -matched and $\{\pi_1, y\} \notin M$. Therefore, there is an edge $\{y, z\} \in M$ and the path $\pi_1, \dots, \pi_j, y, z$ is M -alternating, too. By construction, this entails that $y \in O$.

Concerning (d), assume first that there is an M -alternating path π_1, \dots, π_m with $\pi_1 = i$ such that, for some index $j \in \{1, \dots, m-1\}$, $\pi_j = x$ with $x \in O$ and where $\{\pi_j, \pi_{j+1}\} \in M$ holds. In this case, $(x, \pi_{j+1}) \in B$ holds, by construction. The other possibility is that $\{i, x\} \in E \setminus M$. In this case, since i is blocked by M , there must be a node $z \neq i$ such that $\{x, z\} \in M$. Again, by construction, we have $(x, z) \in B$.

Concerning (f), note that there is no $x \in N$ with $\{i, x\} \in M$; otherwise, the path i, x would witness that i is blocked by M . By construction, we hence have $(i, x) \notin B$ and $x \in O$.

(3) \Rightarrow (2) Assume that B , I , and O satisfy all the properties listed in the statement. Consider the set $M = \{\{x, y\} \mid (x, y) \in B, x \in C \cup \{i\}, y \in C \cup \{i\}\}$. Because of (e), it is immediate that M is a matching in $G[C \cup \{i\}]$. Moreover, by (a), (b), and (f), we derive that there is no node $y \in N$ such that $\{i, y\}$ is in M , that is, i is M -exposed. Consider then any M -alternating path π_1, \dots, π_m of $G[C \cup \{i\}]$ with $\pi_1 = i$. We shall show that π_m is M -matched.

To this end, we first claim that, for each natural number $h \geq 1$ such that $2h+1 \leq m$, it holds that $\{\pi_{2h-1}, \pi_{2h}\} \notin M$,

$(\pi_{2h}, \pi_{2h+1}) \in B$, $\pi_{2h} \in O$. For $h = 1$, we have just to observe that $\{\pi_1, \pi_2\} \notin M$ and, by (f) $\pi_2 \in O$. Moreover, from (d) we necessarily have $(\pi_2, z) \in B$ with $z = \pi_3$, because $\{\pi_2, \pi_3\} \in M$, by alternation, and (e) must hold. So, let us assume that the property holds at some natural number h , and we show that it holds at $h+1$ with $2(h+1)+1 \leq m$, too. Indeed, we know that $(\pi_{2h}, \pi_{2h+1}) \in B$ and hence $\{\pi_{2h}, \pi_{2h+1}\} \in M$, by construction. Moreover, by (b), $\pi_{2h+1} \in I$. By (c), $\pi_{2h+2} \in O$. By alternation, $\{\pi_{2h+1}, \pi_{2h+2}\} \notin M$. Then, as in the base case, by (d) and (e), $(\pi_{2h+2}, \pi_{2h+3}) \in B$ and $\{\pi_{2h+2}, \pi_{2h+3}\} \in M$.

In the light of this result, if m is odd, then we can immediately conclude that π_m is M -matched. Otherwise, we know that $(\pi_{m-2}, \pi_{m-1}) \in B$. By (b), $\pi_{m-1} \in I$. By (c), $\pi_m \in O$. By (d), there is a node z such that $(\pi_m, z) \in B$. Then, we conclude that π_m is M -matched. \square

5 Tractability Results

We are now ready to discuss the third ingredient of our elaboration, namely the use of tractability results for the evaluation of certain logic formulas bounded-treewidth graphs.

5.1 MSO Formulas

Recall that we viewed any graph $G = (N, E)$ as a structure whose domain is N and where the set of its edges is encoded as a binary relation $r_E = \bigcup_{\{x, y\} \in E} \{(x, y), (y, x)\}$.

Over structures of this kind, we can build *monadic second order* (MSO) logic formulas by using the relation r_E , individual variables, the logical connectives \vee , \wedge , and \neg , and the quantifiers \exists and \forall . Moreover, these formulas allow the use of *node* and *edge* variables, ranging over the possible subsets of N and r_E , respectively, of the membership relation \in , and of the quantifiers \exists and \forall over node and edge variables. In addition, it is often convenient to use symbols like $=$, \subseteq , \subset ,

\cap , \cup , and \rightarrow with their usual meaning, as abbreviations. The fact that a MSO formula ϕ holds over G is denoted by $G \models \phi$.

Example 5.1. The well-known 3-coloring problem on $G = (N, E)$ can be encoded via the following MSO formula ϕ :

$$\begin{aligned} \exists R, B, Y, \quad & R \cup B \cup Y = N \wedge \\ & R \cap B = \emptyset \wedge R \cap Y = \emptyset \wedge B \cap Y = \emptyset \wedge \\ & \forall x, x \in B \rightarrow (\forall y, (x, y) \in r_E \rightarrow \neg(y \in B)) \wedge \\ & \forall x, x \in R \rightarrow (\forall y, (x, y) \in r_E \rightarrow \neg(y \in R)) \wedge \\ & \forall x, x \in Y \rightarrow (\forall y, (x, y) \in r_E \rightarrow \neg(y \in Y)) \end{aligned}$$

Indeed, $G \models \phi$ if, and only if, there exists a partition of the nodes in N into three disjoint sets of nodes R , B , and Y (corresponding to nodes colored red, blue, and yellow) and such that no adjacent nodes take the same color. \triangleleft

A basic relationship between treewidth and MSO is illustrated in the following master theorem.

Theorem 5.2 (cf. [Courcelle, 1990]). *Let ϕ be a fixed MSO sentence, let k be a fixed constant, and let \mathcal{C}_k be a class of graphs having treewidth bounded by k . Then, for each $G \in \mathcal{C}_k$, deciding whether $G \models \phi$ is feasible in linear time.*

For instance, from the above theorem and Example 5.1, we can immediately conclude that 3-colorability is a property that can be checked in linear time on classes of graphs having bounded treewidth—in general, the problem is known to be NP-complete (see, e.g., [Garey and Johnson, 1979]).

To our ends, however, we need a recently proposed histogram version of Courcelle’s theorem, which allows us to count solutions of a given size to MSO formulas on graphs having bounded treewidth. The result it recalled below.

Let $\phi(C)$ be a MSO sentence, where C is a free, i.e., not quantified, node set variable and, for each $\hat{C} \subseteq N$, let $\phi(\hat{C})$ denote the formula where C is fixed to be the set \hat{C} . Then, for each graph $G = (N, E)$ and each natural number $d \in \{1, \dots, |N|\}$, define $\text{histogram}(\phi(C), G, d)$ as the number of subsets $\hat{C} \subseteq N$ with $|\hat{C}| = d$ and such that $G \models \phi(\hat{C})$.

Theorem 5.3 (simplified from [Elberfeld *et al.*, 2010]). *Let $\phi(C)$ be a fixed MSO sentence, let k be a fixed constant, and let \mathcal{C}_k be a class of graphs having treewidth bounded by k . Then, for each $G = (N, E) \in \mathcal{C}_k$ and for each natural number $d \in \{1, \dots, |N|\}$, computing $\text{histogram}(\phi(C), G, d)$ is feasible in deterministic logspace (hence, polynomial time).*

For an illustration, let $\phi(R)$ be the formula obtained from the one discussed in Example 5.1 by dropping the existential quantification of R . Then, $\text{histogram}(\phi(R), G, 3)$ returns the number of triples of nodes that can be colored with the same color (red) in some valid coloring of G . According to the above result, this number can be computed in polynomial time, whenever G is taken from some class of graphs having bounded treewidth.

5.2 Putting It All Together

The above result is the basis for our complexity analysis of the Shapley value. The characterization illustrated in Section 3 for the marginal contribution will also play a crucial role here.

Theorem 5.4. *The Shapley value can be computed in polynomial time on matching games defined over any class of graphs having bounded treewidth.*

Proof. Let us start by noticing that, for each agent $i \in N$, the Shapley value can be rewritten as follows:

$$\phi_i(\mathcal{G}_G) = \sum_{h=0}^{n-1} \frac{h!(n-h-1)!}{n!} \beta_i(\mathcal{G}_A, h),$$

where $n = |N|$ and, for each $h \in \{0, \dots, n-1\}$, it holds that

$$\beta_i(\mathcal{G}_G, h) = \sum_{C \subseteq N \setminus \{i\}, |C|=h} (v_G(C \cup \{i\}) - v_G(C)). \quad (1)$$

We show that the value of the expression in Equation 1 can be computed in polynomial time.

The basic observation is that we can build a *Monadic Second Order* (MSO) formula $\phi_i(C)$ encoding conditions **(a)**–**(f)** in Lemma 4.2, where G is viewed as a relational structure over domain N and relation r_E . In particular, B is an existentially-quantified edge-set variable, I and O are existentially-quantified node-set variables, and C is a free node-set variable. Then, given Lemma 4.1 and Lemma 4.2, the number of coalitions $C \subseteq N \setminus \{i\}$ for which $v_G(C \cup \{i\}) - v_G(C) = 0$ holds and with $|C| = h$ coincides with the number of coalitions $\hat{C} \subseteq N \setminus \{i\}$ with $|\hat{C}| = h$ for which $G \models \phi_i(\hat{C})$. This number can be computed in polynomial time because of Theorem 5.3 and, based on it, the value of the expression in Equation 1 can be easily computed (in polynomial time, too) by observing that $v_G(C \cup \{i\}) - v_G(C) \in \{0, 1\}$, for each $C \subseteq N \setminus \{i\}$. \square

A similar approach can be used to prove the tractability on graphs having bounded treewidth of the *Banzhaf value*,² for which the payoff of i is

$$\beta_i(\mathcal{G}_G) = \frac{1}{2^{|N|-1}} \sum_{C \subseteq N \setminus \{i\}} (v_G(C \cup \{i\}) - v_G(C)).$$

In this case we are even able to derive a linear time algorithm, rather than a polynomial-time one.

Theorem 5.5. *The Banzhaf value can be computed in linear time on matching games defined over any class of graphs having bounded treewidth.*

Proof Sketch. We shall show that, for each node $i \in N$, we can compute in linear time the value of the expression

$$\sum_{C \subseteq N \setminus \{i\}} (v_G(C \cup \{i\}) - v_G(C)). \quad (2)$$

Indeed, we can proceed as in the proof of Theorem 5.4, by eventually noticing that the number of coalitions $C \subseteq N \setminus \{i\}$ for which $v_G(C \cup \{i\}) - v_G(C) = 0$ holds coincides with the number of coalitions $\hat{C} \subseteq N \setminus \{i\}$ for which $G \models \phi_i(\hat{C})$.

In this case, however, a linear-time algorithm follows and thus we conclude, by recalling that counting all instantiations of a free node variable leading to satisfy a given MSO formula was shown to be feasible in linear time over graphs having bounded treewidth [Courcelle *et al.*, 2001]. \square

²The question of whether this value is intractable over arbitrary matching games is not addressed in the literature, and is an interesting avenue of research, though outside the scope of this paper.

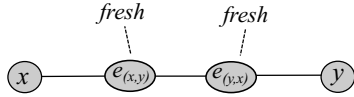


Figure 5: Illustration of directed edge encoding.

6 MSO in Practice

An interesting feature of our approach to show the tractability of the Shapley value over bounded treewidth games is that the MSO encoding provides us with a declarative, yet “executable” specification for its computation. Indeed, we directly followed the steps shown in the proof of Theorem 5.4 by implementing them in a JAVA system prototype and by delegating to the MSO solver *Sequoia* [Langer, 2013] the computation of the histogram of the formula in that proof

Implementation Issues. Note that *Sequoia* supports the evaluation of MSO formulas defined over graphs whose nodes can be equipped with some input labels. However, only undirected graphs can be dealt with. This is not problematic when checking the existence of an edge $(x, y) \in r_E$, since r_E is a symmetric relation. However, the set B in Lemma 4.2 is a set of directed edges. In order to deal with the orientation of edges in B , we transformed every undirected edge $\{x, y\}$ in the input graph by adding two auxiliary nodes labeled “fresh”, as depicted in Figure 5. Then, in the specifications reported in Figure 4, a set of “fresh” nodes B_N can be used to simulate the set B , with the intended meaning that $(x, y) \in B$ (resp., $(y, x) \in B$) if, and only if, there are two fresh nodes $e_{(x,y)}$ and $e_{(y,x)}$ as in Figure 5 and where $e_{(x,y)} \in B_N$ (resp., $e_{(y,x)} \in B_N$).

Benchmark. In order to assess the performances of the prototype implementation, we considered a benchmark of synthetic instances taken from three distinct families of graphs, namely (complete binary) trees, ladder graphs, and Halin graphs. These are well-known classes of graphs (whose treewidth is 1, 2, and 3, respectively) which can be used to evaluate the behavior of the prototype for different treewidth values and for growing sizes of the input graphs. In addition, we considered some real graphs corresponding to author relationships of scientific works in a university (see [Greco and Scarcello, 2013; 2014]).

Results. Experiments have been performed on a dedicated machine equipped with an Intel Core i7-3770k 3.5 GHz processor, 12 GB (DDR 1600 MHz) of RAM, and operating system Linux Debian Jessie. Our Java algorithms were executed on the JDK 1.8.0 05-b13 and the GNU g++-4.9 compiler was used to compile *Sequoia*. Results are reported in Table 1.

It can be checked that, for any fixed treewidth value, the prototype nicely scales w.r.t. the size considered of the input graphs. As expected, however, performances deteriorate at the growing of the treewidth. In addition to the timings, it is very interesting to consider the overall number of blocked coalitions that *Sequoia* has discovered, for each given instance of the benchmark. In many cases, this number is impressive, thereby witnessing the effectiveness of the symbolic computation exploited by the MSO solver. In fact, we remark that results are reported for graphs up to 100 nodes, for which a naïve computation method based on explicitly enumerating all possible (2^{100}) coalitions is clearly unfeasible.

INSTANCE	N. OF NODES	N. OF EDGES	TW	TIME [SEC]	N. OF BLOCKING COALITIONS
TREE	10	9	1	1.9	301
TREE	20	19	1	2.7	242451
TREE	30	29	1	2.7	344506149
TREE	40	39	1	3.0	$3.28E+11$
TREE	50	49	1	3.0	$2.96E+14$
TREE	60	59	1	3.2	$2.70E+17$
TREE	70	69	1	3.2	$1.46E+20$
TREE	80	79	1	3.3	$3.65E+20$
TREE	90	89	1	3.3	$4.96E+20$
TREE	100	99	1	3.9	$6.01E+20$
LADDER-3	6	7	2	37.1	16
LADDER-6	12	16	2	144.2	1024
LADDER-9	18	22	2	57.8	65536
LADDER-12	24	34	2	19.6	4194304
LADDER-15	30	43	2	22.7	268435456
LADDER-18	36	52	2	20.0	17179869184
LADDER-21	42	61	2	458.3	$1.09E+12$
LADDER-24	48	70	2	20.1	$7.03E+13$
LADDER-27	54	79	2	612.6	$4.50E+15$
LADDER-30	60	88	2	33.7	$2.88E+17$
HALIN-6	6	9	3	195.7	14
HALIN-8	8	12	3	930	58
HALIN-10	10	16	3	194.4	237
HALIN-12	12	18	3	3510.2	867
HALIN-15	15	23	3	539.6	6990
COAUTH-1	10	12	2	7.5	230
COAUTH-2	15	15	2	20.3	5040
COAUTH-3	20	27	3	435.4	193998
COAUTH-4	25	28	2	95.4	5909504
COAUTH-5	30	37	3	477.6	211235477
COAUTH-7	35	42	2	70.1	9817601112
COAUTH-8	40	45	2	95.4	$2.37E+11$
COAUTH-9	45	50	2	84.7	$1.10E+13$

Table 1: Summary of experimental results.

7 Conclusion

Computing the Shapley value has been shown to be #P-complete on different classes of games (see, e.g., [Deng and Papadimitriou, 1994; Nagamochi *et al.*, 1997; Bachrach and Rosenschein, 2009]), including *matching games* [Aziz and de Keijzer, 2014]. Moving from these bad news, we have faced the questions left open by [Aziz and de Keijzer, 2014; Bousquet, 2015], by showing that the Shapley value (and the Banzhaf value) of matching games can be computed in polynomial time over graphs having bounded treewidth.

Note that for the related class of *allocation games* [Moulin, 1992], the Shapley value is also known to be #P-complete, but it is tractable when restricted over graphs having bounded treewidth [Greco *et al.*, 2015]. Despite the similarities with allocation games, inspection reveals that the approach of [Greco *et al.*, 2015] cannot be extended to matching games. Moreover, our approach differs from the one used by [Bousquet, 2015] to show tractability of matching games over trees.

An interesting avenue of further research is to assess whether, on bounded treewidth games, the Shapley value is not only tractable, but even *fixed-parameter* tractable, that is, solvable in time $f(k) \cdot \|G\|^{O(1)}$, for each $G \in \mathcal{C}_k$, for some function f depending only on the width k . Moreover, it is open whether tractability still holds over *weighted* graphs.

References

- [Alkan and Gale, 1990] Ahmet Alkan and David Gale. The core of the matching game. *Games and Economic Behavior*, 2(3):203–212, 1990.
- [Aziz and de Keijzer, 2014] Haris Aziz and Bart de Keijzer. Shapley meets shapley. In *Proc. of STACS'14*, pp. 99–111.
- [Bachrach and Rosenschein, 2009] Yoram Bachrach and Jeffrey S. Rosenschein. Power in threshold network flow games. *Autonomous Agents and Multi-Agent Systems*, 18(1):106–132, 2009.
- [Banzhaf, 1965] John F. Banzhaf. Weighted Voting Doesn't Work: A Mathematical Analysis. *Rutgers Law Rev.*, 19:317–343, 1965.
- [Berge, 1957] Claude Berge. Two Theorems in Graph Theory. *Proceedings of the National Academy of Sciences of the United States of America*, 43(9):842–844, 1957.
- [Biró *et al.*, 2011] Péter Biró, Walter Kern, and Daniël Paulusma. Computing solutions for matching games. *International Journal of Game Theory*, 41(1):75–90, 2011.
- [Bousquet, 2015] Nicolas Bousquet. The shapley value of matching games on trees. <http://www.math.mcgill.ca/nbousquet/shapley.pdf>, 2015.
- [Chalkiadakis *et al.*, 2016] Georgios Chalkiadakis, Gianluigi Greco, and Evangelos Markakis. Characteristic function games with restricted agent interactions: Core-stability and coalition structures. *Artificial Intelligence*, 232:76–113, 2016.
- [Chen *et al.*, 2012] Ning Chen, Pinyan Lu, and Hongyang Zhang. Computing the nucleolus of matching, cover and clique games. In *Proc. of AAAI'12*, pp. 1319–1325.
- [Courcelle, 1990] Bruno Courcelle. Graph rewriting: An algebraic and logic approach. In Jan van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics*. The MIT Press, 1990.
- [Courcelle *et al.*, 2001] Bruno Courcelle, Johann A. Makowsky, and Udi Rotics. On the fixed parameter complexity of graph enumeration problems definable in monadic second-order logic. *Discrete Applied Mathematics*, 108:23–52, 2001.
- [Deng and Papadimitriou, 1994] Xiaotie Deng and Christos H. Papadimitriou. On the complexity of cooperative solution concepts. *Mathematics of Operations Research*, 19:257–266, 1994.
- [Deng *et al.*, 1999] Xiaotie Deng, Toshihide Ibaraki, and Hiroshi Nagamochi. Algorithmic aspects of the core of combinatorial optimization games. *Mathematics of Operations Research*, 24(3):751–766, 1999.
- [Elberfeld *et al.*, 2010] Michael Elberfeld, Andreas Jakoby, and Till Tantau. Logspace Versions of the Theorems of Bodlaender and Courcelle. In *Proc. of FOCS'10*, pp. 143–152.
- [Garey and Johnson, 1979] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., 1979.
- [Gottlob *et al.*, 2016] Georg Gottlob, Gianluigi Greco, Nicola Leone, and Francesco Scarcello. Hypertree decompositions: Questions and answers. In *Proc. of PODS'16*, pp. 57–74.
- [Greco *et al.*, 2015] Gianluigi Greco, Francesco Lupia, and Francesco Scarcello. Structural tractability of shapley and banzhaf values in allocation games. In *Proc. of IJCAI'15*, pp. 547–553.
- [Greco *et al.*, 2014] Gianluigi Greco, Enrico Malizia, Luigi Palopoli, and Francesco Scarcello. The complexity of the nucleolus in compact games. *Transactions on Computation Theory*, 7(1):3:1–3:52, 2014.
- [Greco and Scarcello, 2013] Gianluigi Greco and Francesco Scarcello. Fair division rules for funds distribution: The case of the italian research assessment program (VQR 2004-2010). *Intelligenza Artificiale*, 7(1):45–56, 2013.
- [Greco and Scarcello, 2014] Gianluigi Greco and Francesco Scarcello. Mechanisms for fair allocation problems: No-punishment payment rules in verifiable settings. *Journal of Artificial Intelligence Research*, 49:403–449, 2014.
- [Kern and Paulusma, 2003] Walter Kern and Daniel Paulusma. Matching games: The least core and the nucleolus. *Mathematics of Operations Research*, 28(2):294–308, 2003.
- [Langer, 2013] Alexander Langer. *Fast algorithms for decomposable graphs*. PhD thesis, RWTH Aachen University, 2013.
- [Moulin, 1992] Herve Moulin. An application of the Shapley value to fair division with money. *Econometrica*, 60(6):1331–1349, 1992.
- [Nagamochi *et al.*, 1997] Hiroshi Nagamochi, Dao-Zhi Zeng, Naohisa Kabutoya, and Toshihide Ibaraki. Complexity of the minimum base game on matroids. *Mathematics of Operations Research*, 22:146–164, 1997.
- [Osborne and Rubinstein, 1994] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory*. The MIT Press, 1994.
- [Robertson and Seymour, 1984] Neil Robertson and Paul D. Seymour. Graph minors iii: Planar tree-width. *Journal of Combinatorial Theory, Series B*, 36(1):49–64, 1984.
- [Roth and Sotomayor, 1990] Alvin E. Roth and Marilda A. Oliveira Sotomayor. *Two-Sided Matching: A Study in Game Theoretic Modelling and Analysis*. Cambridge University Press, 1990.
- [Shapley, 1953] Lloyd S. Shapley. A value for n-person games. *Contributions to the theory of games*, 2:307–317, 1953.
- [Shapley and Shubik, 1971] Lloyd S. Shapley and Martin Shubik. The assignment game I: The core. *International Journal of Game Theory*, 1(1):111–130, 1971.