Journal of the Operations Research Society of Japan
Vol. 63, No. 1, January 2020, pp. 31–40

MONTE CARLO ALGORITHM FOR CALCULATING THE SHAPLEY VALUES OF MINIMUM COST SPANNING TREE GAMES

Kazutoshi Ando   Koichi Takase
Shizuoka University

(Received March 25, 2019; Revised October 9, 2019)

Abstract In this paper, we address a Monte Carlo algorithm for calculating the Shapley values of minimum cost spanning tree games. We provide tighter upper and lower bounds for the marginal cost vector and improve a previous study's lower bound on the number of permutations required for the output of the algorithm to achieve a given accuracy with a given probability. In addition, we present computational experiments for estimating the lower bound on the number of permutations required by the Monte Carlo algorithm.

Keywords: Game theory, algorithm, cooperative game, Shapley value, minimum cost spanning tree

1. Introduction

Consider a joint project in which members plan to connect to an information supplier directly or indirectly by creating links between pairs of members and/or between a member and the supplier. Assuming that there are costs associated with creating the links, the total cost can be minimized by arranging the links to form a tree spanning all members and the supplier. The minimum cost spanning tree game, introduced by Bird [4], is a model for analyzing reasonable ways to allocate the total cost among the members under such circumstances.

Let $N = \{1, \cdots, n\}$ be a set of players and $\mathrm{K}_{N'}$ be the complete graph with vertex set $N' = N \cup \{r\}$. Given a function c that assigns a nonnegative cost $c(v, w)$ to each edge $\{v, w\}$ of $\mathrm{K}_{N'}$, the minimum cost spanning tree game is the following cooperative (cost) game $(N, \tilde{c})$: for all $S \subseteq N$, $\tilde{c}(S)$ is the cost of a minimum cost spanning tree for the subgraph of $\mathrm{K}_{N'}$ induced by $S \cup \{r\}$. The fundamental theory of these games was developed in [4], [7], and [8], and examples of its applications include the problem of building a drainage system that connects every house in a city with a water purifier, and the carpooling problem [12].

In cooperative game theory, a method of sharing the cost among players is called a solution, and one of the most important solution concepts is the Shapley value [15] (see [17] for a survey). Let (N, f) be a cooperative game. For a given permutation π of N, the marginal cost vector $X(\pi)$ is defined as $X(\pi)_{\pi(i)} = f(\{\pi(1), \cdots, \pi(i)\}) - f(\{\pi(1), \cdots, \pi(i-1)\})$ $(i = 1, \cdots, n)$. The Shapley value of the game (N, f) is defined as the average over all its marginal cost vectors. Because computation of the Shapley value of minimum cost spanning tree games is #P-hard [1], when these games are applied to the problems described above, an approximation algorithm is required that can provide a provably good approximation of the Shapley value.

Monte Carlo methods are a natural approach for approximating Shapley values for arbitrary cooperative games, and they have been well-studied in the literature. These algorithms select uniformly random permutations, and output the average of the corresponding marginal cost vectors. Concentration inequalities, such as the Hoeffding inequality [9] and Chebyshev inequality [16], are powerful tools for providing a lower bound on the number of permutations required for such algorithms to achieve a given accuracy with a given probability. See [3] for weighted majority games and [5, 11] for general cooperative games and [10] for supermodular games.

Ando et al. [2] studied the Monte Carlo algorithm applied to the Shapley value of minimum cost spanning tree games and, using the Chebyshev inequality [16], derived a lower bound on the number of permutations for the output of the algorithm to achieve a given accuracy with a given probability. In this study, we provide tighter upper and lower bounds on the marginal cost vector and, using the Hoeffding inequality [9], improve a previous study's lower bound [2] on the number of permutations required for this algorithm. In addition, we estimate the empirical lower bound via computational experiments and compare it with the predicted lower bound.

The remainder of this paper is organized as follows. In Section 2, we review the definition of a minimum cost spanning tree game and several related fundamental results. In Section 3, we present a Monte Carlo algorithm and an improved lower bound on the number of permutations required for the algorithm to achieve a given accuracy with a given probability. In Section 4, we present the results of computational experiments to compare the predicted and true lower bounds. Finally, in Section 5, we summarize our results.

2. Minimum Cost Spanning Tree Games

In this section, we review several definitions from cooperative game theory and define a minimum cost spanning tree game. Here, we denote the set of real numbers by $\mathbb{R}$ and the set of nonnegative real numbers by $\mathbb{R}_+$.

A *cooperative (cost) game* (N, f) is a pair consisting of a finite set $N = \{1, \cdots, n\}$, called the set of *players*, and a function $f : 2^N \to \mathbb{R}$ with $f(\emptyset) = 0$, called the *characteristic function*. For a given permutation π of N and $v \in N$, we denote the set of players that precede v in π by S_v^π, that is,

$$S_v^\pi = \{w \mid w \in N, \pi^{-1}(w) < \pi^{-1}(v)\}.$$

For all permutations π of N, the *marginal cost vector* $X(\pi) : N \to \mathbb{R}$ of (N, f) with respect to π is defined as

$$X(\pi)_v = f(S_v^\pi \cup \{v\}) - f(S_v^\pi) \quad (v \in N). \tag{2.1}$$

The *Shapley value* $\mathrm{Sh}(f) : N \to \mathbb{R}$ of the cooperative game (N, f) is then defined as

$$\mathrm{Sh}(f)_v = \frac{1}{n!} \sum_{\pi \in \Pi_N} X(\pi)_v \quad (v \in N), \tag{2.2}$$

where Π_N is the set of permutations of N.

All graphs considered in this paper are simple undirected graphs; that is, they do not contain self-loops or parallel edges. For graph $G = (V, A)$, subgraph $H = (W, B)$ of G is called a *spanning tree* of G if $V = W$ and H is a tree. We also say that B is a *spanning tree* of $G = (V, A)$ if $H = (W, B)$ is a spanning tree of G.

Let $\mathrm{K}_{N'}$ be the complete graph with vertex set $N' = N \cup \{r\}$ and let $c\colon \binom{N'}{2} \to \mathbb{R}_+$ be a nonnegative-real valued function defined on the edge set of $\mathrm{K}_{N'}$, where r is interpreted as the *source* of a service provided to players. This pair, $(\mathrm{K}_{N'}, c)$, is called a *network*. For all subsets Γ of edges of $\mathrm{K}_{N'}$, the *cost* $c(\Gamma)$ of Γ is defined as

$$c(\Gamma) = \sum_{\{v,w\}\in\Gamma} c(v,w). \tag{2.3}$$

For all $S \subseteq N$, we define $S' = S \cup \{r\}$. The *minimum cost spanning tree game* associated with the network $(\mathrm{K}_{N'}, c)$ is the cooperative game $(N, \tilde{c})$ defined by

$$\tilde{c}(S) = \min\{c(\Gamma) \mid \Gamma \text{ is a spanning tree of } \mathrm{K}_{S'}\} \quad (S \subseteq N), \tag{2.4}$$

where $\mathrm{K}_{S'}$ is the complete subgraph of $\mathrm{K}_{N'}$ with vertex set S'.

For a network $(\mathrm{K}_{N'}, c)$ where c is $\{0,1\}$-valued, we can characterize the characteristic function of the associated minimum cost spanning tree game $(N, \tilde{c})$ as follows.

Lemma 2.1 (Ando [1])**.** *Let* $(\mathrm{K}_{N'}, c)$ *be a network such that* c *is* $\{0,1\}$*-valued. Then, for all* $S \subseteq N$, $\tilde{c}(S)$ *is the number of connected components of graph* $G(c,S) = (S', E(c,S))$ *minus one, where*

$$E(c,S) = \{\{v,w\} \mid \{v,w\} \in \binom{S'}{2}, c(v,w) = 0\}. \tag{2.5}$$

For a general network $(\mathrm{K}_{N'}, c)$, let the distinct positive values $c(v,w)$'s be

$$(0 <)\gamma_1 < \cdots < \gamma_l \tag{2.6}$$

and let $\gamma_0 = 0$. For all $i = 1, \ldots, l$, define $c_i : \binom{N'}{2} \to \{0,1\}$ by

$$c_i(v,w) = \begin{cases} 1 & \text{if } \gamma_i \le c(v,w), \\ 0 & \text{otherwise} \end{cases} \qquad (v, w \in N'). \tag{2.7}$$

Then, we have

$$c = \sum_{i=1}^{l} (\gamma_i - \gamma_{i-1}) c_i. \tag{2.8}$$

Norde, Moretti and Tijs [12] demonstrated that if c is decomposed as in (2.8), then the corresponding characteristic function $\tilde{c}$ is accordingly decomposed as

$$\tilde{c} = \sum_{i=1}^{l} (\gamma_i - \gamma_{i-1}) \widetilde{c_i}. \tag{2.9}$$

The number of summands in (2.9) can potentially be reduced as indicated by the following lemma.

Lemma 2.2. *Let* $(\mathrm{K}_{N'}, c)$ *be a network and suppose that* c *is decomposed as in* (2.8)*. Let* $l^* \le l$ *be the integer such that*

$$\gamma_{l^*} = \max\{c(v,r) \mid v \in N\}. \tag{2.10}$$

Then, we have

$$\tilde{c} = \sum_{i=1}^{l^*} (\gamma_i - \gamma_{i-1}) \widetilde{c_i}. \tag{2.11}$$

Proof. It suffices to demonstrate that if $i > l^*$, then $\widetilde{c_i}(S) = 0$ $(S \subseteq N)$. Suppose that $i > l^*$ and $S \subseteq N$. Then, we have

$$c(v,r) \le \gamma_{l^*} < \gamma_i, \tag{2.12}$$

and thus, by (2.7) $c_i(v,r) = 0$ for $v \in N$. It follows that $T = \{(v,r) \mid v \in S\}$ is a minimum cost spanning tree of $(\mathrm{K}_{S'}, c_i)$ with $c_i(T) = 0$. Therefore, we have $\widetilde{c_i}(S) = 0$. $\square$

3. Monte Carlo Algorithm

Algorithm 1 is a Monte Carlo algorithm for calculating the Shapley values of minimum cost spanning tree games. To obtain a lower bound on the number m of permutations required by the algorithm, we use the Hoeffding inequality [9] in Lemma 3.1. The Hoeffding inequality is often used to analyze the number of samples required for a general sampling-based randomized algorithm to achieve a given accuracy with a given probability. In the context of cooperative games, Bachrach et al. [3] studied a Monte Carlo algorithm for the Shapley-Shubik power index (and the Banzhaf power index) of weighted majority games and used the Hoeffding inequality to analyze the number of permutations (and coalitions) required for the Monte Carlo algorithm. In addition, Maleki et al. [11] used the Hoeffding inequality to derive a lower bound on the number of permutations required by the Monte Carlo algorithm for the Shapley value of general cooperative games.

Input : Network $(\mathrm{K}_{N'}, c)$ and positive integer m
Output: Approximate Shapley value $\widehat{\mathrm{Sh}}$ for the minimum cost spanning tree game $(N, \tilde{c})$

```
for count = 1, ··· , m do
    Choose a permutation π of N uniformly at random;
    Compute the marginal cost vector X(π) of (N, c̃) with respect to π;
    Ŝh_v ← (1/m) X(π)_v   (v ∈ N);
end
```

Algorithm 1: Monte Carlo algorithm for calculating the Shapley value

Lemma 3.1 (Hoeffding inequality [9])**.** *Let $X_1, \cdots, X_m$ be independent random variables such that $l_j \leq X_j \leq u_j$ for $j = 1, \cdots, m$. Then, for all real numbers $\epsilon > 0$, we have*

$$\Pr[|\bar{X} - \mathrm{E}[\bar{X}]| \geq \epsilon] \leq 2 \exp\left(-\frac{2m^2\epsilon^2}{\sum_{j=1}^{m}(u_j - l_j)^2}\right), \tag{3.1}$$

where $\bar{X} = \frac{1}{m}\sum_{j=1}^{m} X_j$.

Estimated upper and lower bounds can be obtained for the marginal costs $X(\pi)_v$ $(v \in N)$ by applying the following lemma.

Lemma 3.2. *Let $(\mathrm{K}_{N'}, c)$ be a network with $c \neq \mathbf{0}$. Suppose that the distinct positive values of $c(v, w)$ are as in (2.6), and let*

$$\Delta_v = \sum\{\gamma_{l^*} - c(v, w) \mid w \in N', w \neq v, c(v, w) \leq \gamma_{l^*}\} \tag{3.2}$$

for all $v \in N$. Then, for all $v \in N$ and $S \subseteq N - v$, we have

$$\gamma_{l^*} - \Delta_v \leq \tilde{c}(S \cup v) - \tilde{c}(S) \leq \gamma_{l^*}. \tag{3.3}$$

Before proving Lemma 3.2, we consider the case in which the cost function c is $\{0, 1\}$-valued. By Lemma 2.1, the lower and upper bounds on the marginal cost can be derived by evaluating the difference in the number of connected components between $G(c, S)$ and $G(c, S \cup v)$ for $v \notin S$.

Lemma 3.3. *Let $(\mathrm{K}_{N'}, c)$ be a network, where c is $\{0, 1\}$-valued and $c \neq \mathbf{0}$. Let $\deg(v)$ be the degree of vertex v in graph $G(c, N) = (N', E(c, N))$, where $E(c, N)$ is defined by (2.5). Then, for all $v \in N$ and $S \subseteq N - v$, we have*

$$1 - \deg(v) \leq \tilde{c}(S \cup v) - \tilde{c}(S) \leq 1. \tag{3.4}$$

Proof. Let $(\mathrm{K}_{N'}, c)$ be a network as described in the statement of the theorem. Let $v \in N$ and $S \subseteq N - v$, and suppose that T is a minimum cost spanning tree of $(\mathrm{K}_{S'}, c)$. Then, $\tilde{c}(S) = c(T)$. Because $T \cup \{v, w\}$ is a spanning tree of $(\mathrm{K}_{(S\cup v)'}, c)$ for any $w \in S'$, we have

$$\tilde{c}(S \cup v) \leq c(T) + c(v, w) \leq \tilde{c}(S) + 1. \tag{3.5}$$

Let $\mathcal{D}(c, S)$ be the vertex sets of the connected components of graph $G(c, S) = (S', E(c, S))$, and let

$$\mathcal{D}' = \{D \mid D \in \mathcal{D}(c, S), c(v, w) = 0 \text{ for some } w \in D\}.$$

Then, by Lemma 2.1, we have $\tilde{c}(S) = |\mathcal{D}(c, S)| - 1$ and $\tilde{c}(S \cup v) = |\mathcal{D}(c, S)| - |\mathcal{D}'|$, from which we obtain

$$\tilde{c}(S \cup v) - \tilde{c}(S) = -|\mathcal{D}'| + 1 \geq -\deg(v) + 1. \tag{3.6}$$

Now, the inequalities (3.4) follow from (3.5) and (3.6). □

Proof of Lemma 3.2. Let $(\mathrm{K}_{N'}, c)$ be a network with $c \neq \mathbf{0}$. Let $v \in N$ and $S \subseteq N - v$, and suppose that c is decomposed as in (2.8). For all $i = 1, \cdots, l$ and $v \in N$, let $\deg_i(v)$ be the degree of v in graph $G(c_i, N)$. Then, because c_i is $\{0, 1\}$-valued and $c_i \neq \mathbf{0}$ for all $i = 1, \cdots, l$, by Lemma 3.3, we have

$$\begin{aligned}
\gamma_{l^*} - \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})\deg_i(v) &= \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})(-\deg_i(v) + 1) \\
&\leq \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})(\widetilde{c_i}(S \cup v) - \widetilde{c_i}(S)) \\
&\leq \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1}) \\
&= \gamma_{l^*}.
\end{aligned} \tag{3.7}$$

Because we also have

$$\tilde{c}(S \cup v) - \tilde{c}(S) = \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})(\widetilde{c_i}(S \cup v) - \widetilde{c_i}(S)) \tag{3.8}$$

by Lemma 2.2, it follows from (3.7) that

$$\gamma_{l^*} - \sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})\deg_i(v) \leq \tilde{c}(S \cup v) - \tilde{c}(S) \leq \gamma_{l^*}. \tag{3.9}$$

Finally, we have $\sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})\deg_i(v) = \Delta_v$ because

$$\begin{aligned}
\sum_{i=1}^{l^*}(\gamma_i - \gamma_{i-1})\deg_i(v) &= \deg_{l^*}(v)\gamma_{l^*} - \sum_{i=1}^{l^*-1}(\deg_{i+1}(v) - \deg_i(v))\gamma_i \\
&= \sum\{\gamma_{l^*} - c(v, w) \mid w \in N', w \neq v, c(v, w) \leq \gamma_{l^*}\} \\
&= \Delta_v.
\end{aligned} \tag{3.10}$$

This completes the proof. □

By Lemma 3.2 and the Hoeffding inequality (Lemma 3.1), we have the following main results of this paper.

Theorem 3.1. *Let* $(\mathrm{K}_{N'}, c)$ *be a network with* $c \neq \mathbf{0}$*. For all* $0 < \epsilon$ *and* $0 < \eta < 1$*, we have the following.*

(i) *For all players* $v \in N$*, the output* $\widehat{\mathrm{Sh}}$ *of Algorithm* 1 *satisfies*

$$\Pr[|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| < \epsilon] \geq 1 - \eta \tag{3.11}$$

if $m \geq \frac{\Delta_v^2 \log\left(\frac{2}{\eta}\right)}{2\epsilon^2}$.

(ii) *Let* $\Delta = \max\{\Delta_v \mid v \in N\}$*. Then, the output* $\widehat{\mathrm{Sh}}$ *of Algorithm* 1 *satisfies*

$$\Pr[\|\widehat{\mathrm{Sh}} - \mathrm{Sh}(\tilde{c})\|_\infty < \epsilon] \geq 1 - \eta \tag{3.12}$$

if $m \geq \frac{\Delta^2 \log\left(\frac{2n}{\eta}\right)}{2\epsilon^2}$.

Proof. Let $(\mathrm{K}_{N'}, c)$ be a network with $c \neq \mathbf{0}$ and $v \in N$, and let M be a set of m uniformly random permutations of N. Because the marginal costs $X(\pi)_v$ $(\pi \in M)$ are defined by

$$X(\pi)_v = \tilde{c}(S_v^\pi \cup \{v\}) - \tilde{c}(S_v^\pi),$$

by Lemma 3.2 we have

$$\gamma_{l^*} - \Delta_v \leq X(\pi)_v \leq \gamma_{l^*} \quad (\pi \in M). \tag{3.13}$$

Because $\widehat{\mathrm{Sh}}_v = \frac{1}{m}\sum_{\pi \in M} X(\pi)_v$ and

$$\mathrm{E}[\widehat{\mathrm{Sh}}_v] = \frac{1}{m}\sum_{\pi \in M} \mathrm{E}[X(\pi)_v] = \frac{1}{m}\sum_{\pi \in M} \mathrm{Sh}(\tilde{c})_v = \mathrm{Sh}(\tilde{c})_v,$$

by the Hoeffding inequality (Lemma 3.1) and (3.13), we have

$$\Pr[|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| \geq \epsilon] \leq 2\exp\left(-\frac{2m^2\epsilon^2}{\sum_{\pi \in M}\Delta_v^2}\right) = 2\exp\left(-\frac{2m\epsilon^2}{\Delta_v^2}\right). \tag{3.14}$$

(i) If $m \geq \frac{\Delta_v^2 \log\left(\frac{2}{\eta}\right)}{2\epsilon^2}$, then, by (3.14), we have

$$\Pr[|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| \geq \epsilon] \leq \eta,$$

and thus, we have (3.11).

(ii) If $m \geq \frac{\Delta^2 \log\left(\frac{2n}{\eta}\right)}{2\epsilon^2}$, then by (3.14) and the union bound we have

$$\begin{aligned}
\Pr[\exists v \in N\colon |\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| \geq \epsilon] &\leq \sum_{v \in N} \Pr[|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| \geq \epsilon] \\
&\leq \sum_{v \in N} 2\exp\left(-\frac{2m\epsilon^2}{\Delta_v^2}\right) \\
&\leq 2n\exp\left(-\frac{2m\epsilon^2}{\Delta^2}\right) \\
&\leq \eta,
\end{aligned}$$

from which the claimed inequality (3.12) follows. □

We can also prove Theorem 3.1(i) using a result of Makeki et al. [11] for the Shapley value of general cooperative games, which was suggested by Bachrach et al. [3]. However, we prove the theorem by using the Hoeffding inequality directly for the sake of completeness.

Ando et al. [2] provided a lower bound $\frac{n^3\gamma_l^2}{4\eta\epsilon^2}$ on the number m of permutations to have (3.12) using the Chebychev inequality [16]. The bound $\frac{\Delta^2 \log\left(\frac{2n}{\eta}\right)}{2\epsilon^2}$ in Theorem 3.1(ii) improves that of Ando et al. because $n\gamma_l \geq \Delta$.

Next, we consider the running time per iteration of Algorithm 1. This time is dominated by the time required to compute the marginal cost vector $X(\pi)$, which is $\mathrm{O}(n^3)$ because $X(\pi)_v$ can be computed in $\mathrm{O}(n^2)$ time by applying Prim's algorithm [13] for all $v = 1, \cdots, n$. However, if c is $\{0,1\}$-valued, then the running time can be reduced to $\mathrm{O}(n^2)$ using a disjoint-set data structure, as illustrated by the following lemma in [2].

Lemma 3.4 (Ando and Tokutake [2]). *Let $(N, \tilde{c})$ be a minimum cost spanning tree game, where c is $\{0,1\}$-valued, and let π be a permutation of N. The marginal cost vector $X(\pi)$ of $(N, \tilde{c})$ with respect to π can be computed in $\mathrm{O}(n^2)$ time.*

Proof. Suppose that c is a $\{0,1\}$-valued function on $\binom{N'}{2}$. For all $k = 0, 1, \cdots, n$ let

$$S_k = \{\pi(1), \cdots, \pi(k)\}$$

and let $\mathcal{D}_k$ be the vertex sets of the connected components of graph $G_k = (S', E_k)$, where

$$E_k = \{\{v, w\} \mid \{v, w\} \in \binom{S'_k}{2}, c(v, w) = 0\}.$$

Then, by Lemma 2.1, we have

$$\begin{aligned} X(\pi)_{\pi(k)} &= \tilde{c}(S_k) - \tilde{c}(S_{k-1}) \\ &= |\mathcal{D}_k| - |\mathcal{D}_{k-1}| \end{aligned} \tag{3.15}$$

for $k = 1, \cdots, n$.

To obtain the required time bound, we dynamically represent the vertex sets $\mathcal{D}_k$ ($k = 0, 1, \cdots, n$) using a linked-list-based implementation of a disjoint-set data structure [6], in which the make-set and find operations take $\mathrm{O}(1)$ time and the union operation takes $\mathrm{O}(n)$ time. Initially, we have $\mathcal{D}_0 = \{\{v\} \mid v = r, 1, \cdots, n\}$. For $k = 1, \cdots, n$, suppose that we have $\mathcal{D}_{k-1}$. $\mathcal{D}_k$ can be computed as follows: for all $w \in S'_{k-1}$ such that $c(\pi(k), w) = 0$, we determine whether $\pi(k)$ and w are in different components using the find operation and, if so, we merge the components into one using the union operation. Thereafter, $X(\pi)_{\pi(k)}$ can be calculated using (3.15).

Because this procedure involves a total of $n(n+1)$ find operations and n union operations, it can be concluded that $X(\pi)$ can be computed in $\mathrm{O}(n^2)$ time. $\square$

Corollary 3.1. *Let $(N, \tilde{c})$ be a minimum cost spanning tree game. In general, the running time of Algorithm 1 is $\mathrm{O}(mn^3)$. If the cost function c is $\{0,1\}$-valued, its running time is $\mathrm{O}(mn^2)$.*

4. Computational Experiments

In Section 3, we consider a Monte Carlo algorithm for calculating the Shapley values of minimum cost spanning tree games and provide an improved lower bound on the number of permutations required for the algorithm to achieve a given accuracy with a given probability. In this section, we compare this lower bound with the true lower bound via computational experiments.

4.1. Problem instances

To experimentally evaluate the accuracy of the output of the Monte Carlo algorithm, we must know the Shapley values of the given minimum cost spanning tree games precisely. Although computing the Shapley values of minimum cost spanning tree games is generally #P-hard, the values can be calculated in polynomial time for certain games. One such class of games is described as follows.

Proposition 4.1 (Ando [1]). *Let* $(\mathrm{K}_{N'}, c)$ *be a network. If graph* $G(c_i, N) = (N', E(c_i, N))$ *is chordal for all* $i = 1, \cdots, l$, *where* c_i *is defined by* (2.7), *then the Shapley value of the minimum cost spanning tree game* $(N, \tilde{c})$ *associated with the network* $(\mathrm{K}_{N'}, c)$ *can be computed in* $\mathrm{O}(ln^2)$ *time.*

We call a cost function $c\colon \binom{N'}{2} \to \mathbb{R}_+$ *chordal* if graph $G(c_i, N) = (N', E(c_i, N))$ is chordal for all $i = 1, \cdots, l$, where c_i is defined by (2.7). It should be noted that tree metrics are a special class of chordal cost functions. A cost function $c\colon \binom{N'}{2} \to \mathbb{R}_+$ is a *tree metric* if there exists a nonnegative-weighted tree $U = (V, E)$ with the leaf set being N' such that for each $v, w \in N'$ the length of the unique path in U from v to w is equal to $c(v, w)$ (see, e.g., [14]).

In this study, we conducted experiments on minimum cost spanning tree games associated with the random chordal cost function $c\colon \binom{N'}{2} \to \mathbb{R}_+$, generated as follows. First, we generated a sequence of chordal graphs G_i $(i = 1, \cdots, l)$ such that $E(G_{i-1}) \subset E(G_i)$ for $i = 2, \cdots, l$. Then, we set $c = \sum_{i=1}^{l} (\gamma_i - \gamma_{i-1}) c_i$, where $c_i\colon \binom{N'}{2} \to \{0, 1\}$ was such that $G(c_i) = G_i$ and γ_i were positive real numbers such that $\gamma_{i-1} < \gamma_i$ for $i = 2, \cdots, l$.

4.2. Experiments

For each random chordal cost function $c\colon \binom{N'}{2} \to \mathbb{R}_+$, we ran the Monte Carlo algorithm (Algorithm 1) 20 times each for $m = 1000, 2000, \cdots, 10000$, counting the ratio of successful runs in each case. Here, we say a run of Algorithm 1 is *successful* if $|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| < \epsilon$, where we set $v = 1$ and $\epsilon = 1/4$. Figure 1 presents the average ratio of successful runs for each of the 10 random chordal cost functions $c\colon \binom{N'}{2} \to \mathbb{R}_+$ with $(n, l, (\gamma_i)) = (100, 100, (i)), (100, 100, (1.5i))$, and $(100, 100, (2i))$, respectively. To achieve $|\widehat{\mathrm{Sh}}_v - \mathrm{Sh}(\tilde{c})_v| < \epsilon$ with probability $1 - \frac{1}{4}$, the required number m of permutations of the Monte Carlo algorithm (Theorem 3.1) is

$$m = \left\lceil \frac{\Delta_v^2 \log\left(\frac{2}{\eta}\right)}{2\epsilon^2} \right\rceil = \left\lceil 24\Delta_1^2 \log 2 \right\rceil, \tag{4.1}$$

where Δ_v is defined by (3.2). For the 10 instances with $(n, l, (\gamma_i)) = (100, 100, (1.5i))$, we found $4,911.0 \leq \Delta_1 \leq 7,570.5$, and we should have thus required $401,214,455 \leq m \leq 953,423,452$ permutations to achieve a probability of success of $1 - \frac{1}{4} = \frac{3}{4}$. However, the experimental results demonstrate that in the case of $\gamma_i = 1.5i$, the ratio of successful runs exceeds $\frac{3}{4}$ if $m \geq 6,000$ (see line of $\gamma_i = 1.5$ in Figure 1). The number $m = 6,000$ is much smaller than the estimate given by Theorem 3.1.

5. Summary and Concluding Remarks

In this study, we examined the Monte Carlo algorithm for calculating the Shapley values of minimum cost spanning tree games. We provided tighter upper and lower bounds on the marginal cost vector and improved a previous study's lower bound [2] on the number of permutations required for the algorithm to achieve a given accuracy with a given probability. We then estimated the true lower bound via computational experiments. The results suggest that the algorithm can output favorable approximations of the Shapley value with fewer

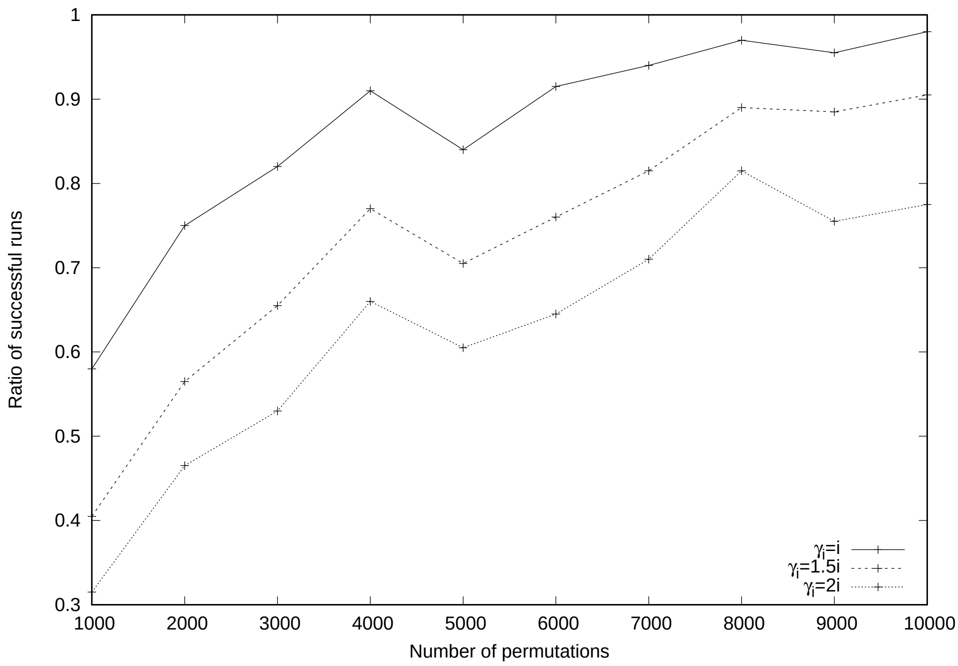

Figure 1: Average ratio of successful runs vs number of permutations

iterations than predicted by Theorem 3.1; therefore, it may be possible to further improve the lower bound.

Acknowledgments

This work was supported by the Japan Society for the Promotion of Science, KAKENHI Grant Numbers 15K00033 and 18K11180. The authors are grateful to the anonymous referees for useful comments which improved the presentation of this paper.

References

[1] K. Ando: Computation of the Shapley value of minimum cost spanning tree games: #P-hardness and polynomial cases. *Japan Journal of Industrial and Applied Mathematics*, **29** (2012), 385–400.

[2] K. Ando and T. Tokutake: Approximation algorithm for the Shapley value of minimum cost spanning tree games based on sampling (in Japanese). *Proceedings of the 2013 Spring National Conference of the Operations Research Society of Japan* (2013), 148–149.

[3] Y. Bachrach, E. Markakis, E. Resnick, A.D. Procaccia, J.S. Rosenschein and A. Saberi: Approximating power indices: theoretical and empirical analysis. *Autonomous Agents and Multi-Agent Systems*, **20** (2010), 105–122.

[4] C.G. Bird: On cost allocation for a spanning tree. A game theoretic approach. *Networks*, **6** (1976), 335–350.

[5] J. Castro, D. Gómez, E. Molina and J. Tejada: Improving polynomial estimation of the Shapley value by stratified random sampling with optimum allocation. *Computers & Operations Research*, **82** (2017), 180–188.

[6] T.H. Cormen, C.E. Leiserson, R.L. Rivest and C. Stein: *Introduction to Algorithms, Third Edition* (MIT Press, 2009).

[7] D. Granot and G. Huberman: Minimum cost spanning tree games. *Mathematical Programming*, **21** (1981), 1–18.

[8] D. Granot and G. Huberman: The relationship between convex games and minimum cost spanning tree games: a case for permutationally convex games. *SIAM Journal on Algebraic and Discrete Methods*, **3** (1982), 288–292.

[9] W. Hoeffding: Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, **58** (1963), 13–30.

[10] D. Liben-Nowell, A. Sharp, T. Wexler and K. Woods: Computing Shapley value in supermodular coalitional games. *Lecture Notes in Computer Science* **7434** (2012), 568–579.

[11] S. Maleki, L. Tran-Thanh, G. Hines, T. Rahwan and A. Rogers: Bounding the estimation error of sampling-based Shapley value approximation. arXiv:1306.4265, February 2014.

[12] H. Norde, S. Moretti and S. Tijs: Minimum cost spanning tree games and population monotonic allocation schemes. *European Journal of Operational Research*, **154** (2004), 84–97.

[13] R.C. Prim: Shortest connection networks and some generalizations. *Bell System Technical Journal*, **36** (1957), 1389–1401.

[14] C. Semple and M. Steel: *Phylogenetics* (Oxford University Press, 2003).

[15] L.S. Shapley: A value for n-person games. In H. W. Kuhn and A. W. Tucker (eds.): *Contributions to the Theory of Games, volume II* (Princeton University Press, 1953), 307–317.

[16] P. Tchebichef: Des valeurs moyennes. *Journal de Mathématiques Pures et Appliquées 2*, **12** (1867), 177–184.

[17] E. Winter: The Shapley value. In R. J. Aumann and S. Hart (eds.): *Handbook of Game Theory with Economic Applications, volume 3* (Elsevier, 2002), 2025–2054.

Kazutoshi Ando
Department of Mathematical and Systems Engineering
Faculty of Engineering, Shizuoka University
Hamamatsu 432-8561, Japan
E-mail: ando.kazutoshi@shizuoka.ac.jp