Minimum Cut Tree Games

Anne M. Schwahn*

Abstract

In this paper we introduce a cooperative game based on the minimum cut tree problem which is also known as multi-terminal maximum flow problem. Minimum cut tree games are shown to be totally balanced and a solution in their core can be obtained in polynomial time. This special core allocation is closely related to the solution of the original graph theoretical problem. We give an example showing that the game is not supermodular in general, however, it is for special cases and for some of those we give an explicit formula for the calculation of the Shapley value.

Keywords: cooperative game, minimum cut tree, core, Shapley value, cactus graph

1 Introduction

Cooperative games on graphs combine the theories of graphs, optimization, and games. One classic problem in graph theory is to find a maximum flow. In combinatorial optimization there is one central decision maker who controls all resources and aims to optimize her objective. For the maximum flow problem several efficient algorithms are known to solve it to optimality. The situation changes if the system is not controlled by one individual but by several decision makers, the so called *players* who may have conflicting objectives. A solution which is optimal for one decision maker or for the system as a whole may not be accepted by the others and thus may not be implementable. Therefore it is desirable to find solutions which are attractive for at least those players needed to realize them. The maximum flow problem has been investigated intensively from this game-theoretic perspective: in maximum flow games the edges are controlled

*University of Kaiserslautern, Department of Mathematics, P.O.Box 3049, 67653 Kaiserslautern, Germany, `schwahn@mathematik.uni-kl.de`

by players who want a reward for providing the capacity of their edges to the transport of a good. Kalai and Zemel [KZ82] defined the maximum flow game and showed that the solution of the corresponding graph theoretical problem yields a core solution of their game, i.e., an allocation of rewards such that every player has an incentive to cooperate. Surveys on the maximum flow game and other cooperative games on graphs are given by Curiel [Cur97] and Borm et al [BHH01].

In this paper we introduce and investigate a game based on the minimum cut tree problem. We start by summarizing the necessary concepts in cooperative game theory and graph theory and define the minimum cut tree game in Section 2. Some properties of the game are discussed in Subsection 3.1 and a variant of the game is defined in Subsection 3.2, we state and prove a core allocation in Subsection 3.3. Finally, explicit formulas for the Shapley value of special cases are given in Subsection 3.4.

2 Preliminaries And Notation

2.1 Cooperative Game Theory

A *cooperative game* with *transferable utility* consists of a set of *players* N and a *characteristic function* $v : 2^N \mapsto \mathbb{R}$ mapping every subset of players (a so called *coalition*) to a real value. The game is *superadditive* if $v(S) + v(T) \leq v(S \cup T)$ for all coalitions $S, T \subseteq N$ with $S \cap T = \emptyset$. Superadditive reward games incite disjoint coalitions to join. In a *supermodular* game $v(S) + v(T) \leq v(S \cup T) + v(S \cap T)$ holds for all coalitions.

A *payoff vector* is a vector in $\mathbb{R}^N$ assigning a value x_i to every player i. An *imputation* is a payoff vector which is feasible and efficient – $\sum_{i \in N} x_i = v(N)$ – as well as individually rational – $x_i \geq v(\{i\})$ for all $i \in N$. The central question of a cooperative reward game is: Once a coalition has been formed, how do we allocate the rewards such that every member of the coalition is satisfied? There are several answers to this question based on different philosophies. The *core* $C(v)$ of a game consists of all imputations satisfying $\sum_{i \in S} x_i \geq v(S)$ for all coalitions $S \subseteq N$, i.e., every coalition is better off joining the grand coalition N and the reward of the grand coalition is completely distributed among the players. The core of a game may be empty, a singleton or a convex polyhedron. A game with non-empty core is *balanced* and vice versa. If all its subgames are balanced the game is *totally balanced*. Another important concept is the Shapley value [Sha53], which is the average over all marginal vectors corresponding to all permutations of players. Let π be a permutation of N and $P(\pi, i)$ the predecessors of i w.r.t. π, then a marginal vector $m^\pi(v)$ is defined by its entries $m_i^\pi(v) := v(P(\pi, i) \cup \{i\}) - v(P(\pi, i))$. Thus, the Shapley value can be calculated as $\phi(v) := \frac{1}{n!} \sum_{\pi \in \Pi_N} m^\pi(v)$. The Shapley value of a general game may not lie in the core, however, in supermodular game it is an element of the core. Therefore, the core of a supermodular game is non-empty. In general, the calculation of the Shapley value is #P-complete and the decision whether the core of a game is non-empty is in NP due to results of Deng and Papadimitriou [DP94].

2.2 Minimum Cut Tree Problem

Let $G = (V, E)$ be an undirected graph and $w : E \mapsto \mathbb{R}_+$ a weight function on its edges. In order to calculate the maximum flow or minimum cut between any pair of vertices one could solve $\frac{n(n-1)}{2}$ single source - single sink problems. Gomory and Hu [GH61] came up with a smarter algorithm solving the problem in $O(n^4)$. Their procedure yields a minimum cut tree $\mathcal{T}$ of G such that every edge e of $\mathcal{T}$ induces a cut t_e in G and the weight of e is equal to the sum of the weights of edges of the original graph G in the cut, denoted by w_{t_e}. Given any pair of vertices $u, v \in V$, the minimum cut in G separating them equals the minimum cut in $\mathcal{T}$ in terms of capacity and vertex partition. An edge of $\mathcal{T}$ is not necessarily an edge of G. In the remainder of this paper, if we minimize an objective function over the set of trees in a graph, these trees may also contain edges which are not edges of the original graph. The problem can be described by $\min_{\mathcal{T}\text{: tree in } G} \sum_{e \in \mathcal{T}} w_{t_e}$ or alternatively by $\min_{\mathcal{T}\text{: tree in } G} \sum_{e \in \mathcal{T}} w_e \cdot l_e^{\mathcal{T}}$ where $l_e^{\mathcal{T}}$ is the number of edges on the path in $\mathcal{T}$ connecting the endvertices of e. The complexity of the Gomory-Hu algorithm can be improved to $O(n\tau)$ using a better maximum flow algorithm with complexity $O(\tau)$, e.g. the algorithm of Goldberg and Rao [GR98] with $O(\min(n^{\frac{2}{3}}, m^{\frac{1}{2}})m \log(\frac{n^2}{m}) \log U)$ where U is the largest weight and all weights are integral. In the following we review the minimum cut tree algorithm of Gomory and Hu:

Algorithm 2.1. *Algorithm of Gomory and Hu*
Input: undirected graph $G = (V, E)$, *weight function* $w : E \mapsto \mathbb{R}_+$
Output: minimum cut tree $\mathcal{T}$

1. *Initialize* $V(\mathcal{T}) := \{V(G)\}$, $E(\mathcal{T}) := \emptyset$

2. *Choose* $X \in V(\mathcal{T})$ *with* $|X| \geq 2$*, if none exists go to 6.*

3. *Choose* $u, v \in X$ *with* $u \neq v$

   Execute for all connected components C *of* $\mathcal{T} \setminus X$*:*

   Let $S_C := \bigcup_{Y \in V(C)} Y$

   Let (G', w') *arise from* (G, w) *by contracting* S_C *to a single vertex* v_C

   (So $V(G') = X \cup \{v_C : C$ *is a connected component of* $\mathcal{T} \setminus X\}$*)*

4. *Find a minimum* $u - v$*-cut* $(U', V(G') \setminus U')$ *in* (G', w')

   Let $W' := V(G') \setminus U'$

   Set $U := (\bigcup_{v_c \in U' \setminus X} S_C) \cup (U' \cap X)$ *and* $W := (\bigcup_{v_c \in W' \setminus X} S_C) \cup (W' \cap X)$

5. *Set* $V(\mathcal{T}) := (V(\mathcal{T}) \setminus \{X\}) \cup \{U \cap X, W \cap X\}$

   For each edge $e = \{X, Y\} \in E(\mathcal{T})$ *incident to vertex* X *do:*

       If $Y \subseteq U$ *then set* $e' := \{U \cap X, Y\}$ *else set* $e' := \{W \cap X, Y\}$

       Set $E(\mathcal{T}) := (E(\mathcal{T}) \setminus \{e\}) \cap \{e'\}$ *and* $w(e') := w(e)$

   Set $E(\mathcal{T}) := E(\mathcal{T}) \cup \{\{U \cap X, W \cap X\}\}$ *and* $w(\{U \cap X, W \cap X\}) := w'(U', W')$

   Go to 2.

6. *Replace all* $\{x\} \in V(\mathcal{T})$ *by* x *and all* $\{\{x\}, \{y\}\} \in E(\mathcal{T})$ *by* $\{x, y\}$

   STOP

An example for the minimum cut tree of a graph is shown in Figure 1 in Subsection 3.1.

2.3 Definition Of The Minimum Cut Tree Game

Let $G = (N \cup r, E)$ be an undirected graph with edge set E and vertex set $\{i_1, \ldots, i_n\} \cup r$. The specified vertex r is the *root vertex*. The other vertices are owned by and will be identified with players. We denote $N_r := N \cup r$. Let $w : N_r \times N_r \mapsto \mathbb{R}_+ \cup \{0\}$ be a weight function mapping pairs of vertices to non-negative numbers. Let $S \subseteq N$ be a coalition, then $S_r := S \cup r$ and G^{S_r} is the subgraph of G induced by S_r.

The characteristic function of a cooperative *minimum cut tree game* is defined as follows:

$$v(S) = \min_{\mathcal{T}:\,\text{tree in}\,G^{S_r}} \sum_{i<j \in S_r} w_{ij} \cdot l_{ij}^{\mathcal{T}}.$$

Here, $i < j \in S_r$ implies that each vertex pair is only considered once.

3 Minimum Cut Tree Game

3.1 Examples And Properties

Two examples will be used to visualize some of the considerations in this section. Let G be the unweighted complete graph with n vertices plus root. We refer to the game it implies as *unitgame*. The minimum cut tree for a coalition $S \subseteq N$ is always a star tree, no matter which vertex is the center, $l_{ij} = 1$ if either i or j is the center and $l_{ij} = 2$ else. As a tree for $|S|$ players has $|S| + 1$ vertices, $|S|$ tree edges and $\frac{(|S|+1)\times|S|}{2} - |S|$ non-tree edges, we get $v(S) = |S| + 2 \cdot (\frac{(|S|+1)\times|S|}{2} - |S|) = |S|^2$.

The game referred to as *second game* is defined by the graph in Figure 1. On the right hand side the minimum cut tree is given for $S = N$. The values of the coalitions are given in Table 1.

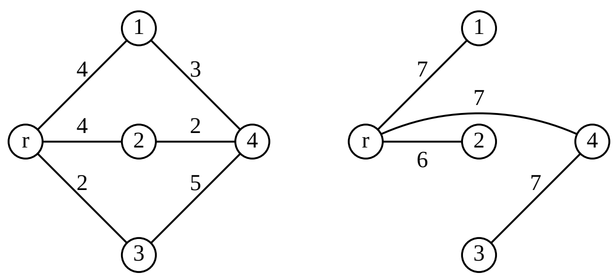

Figure 1: Example of a game with four players

S	$v(S)$	S	$v(S)$	S	$v(S)$	S	$v(S)$	S	$v(S)$
1	4	4	0	1,4	7	3,4	7	1,3,4	18
2	4	1,2	8	2,3	6	1,2,3	10	2,3,4	17
3	2	1,3	6	2,4	6	1,2,4	17	1,2,3,4	27

Table 1: Values of the coalitions

We give a construction scheme to obtain restricted trees which will be used in several proofs later on.

Construction 3.1. *Given a graph $G = (N, E)$, a minimum cut tree $\mathcal{T}^{N_r}$ and a coalition $S \subseteq N$, we construct a tree $\mathcal{T}_{|S_r}$ by restricting $\mathcal{T}^{N_r}$ to S. First, $\mathcal{T}_{|S_r}$ is initialized with all vertices of S_r and no edges. Now, for every vertex $i \in S$ we look at the path from i to*

r in $\mathcal{T}^{N_r}$, denominated by $P_{ir}^{\mathcal{T}^{N_r}}$. Let w_i be the weight of the first edge on $P_{ir}^{\mathcal{T}^{N_r}}$ starting from i and let j be the first vertex of S_r on this path. Observe that j may be equal to r. Connect i and j in $\mathcal{T}_{|S_r}$ by an edge with weight w_i.

Lemma 3.2. *The weight of the restricted tree $\mathcal{T}_{|S_r}$ obtained by Construction 3.1 is not smaller than the weight of a minimum cut tree in G^{S_r}, i.e.,*

$$\sum_{e \in \mathcal{T}_{|S_r}} w_e \geq \min_{\mathcal{T}^{S_r} \in G^{S_r}} \sum_{e \in \mathcal{T}^{S_r}} w_{t_e}^{G^{S_r}}.$$

Proof. First, we have to show that w_e in $\mathcal{T}_{|S_r}$ is not smaller than $w_{t_e}^{G^{S_r}}$, the weight of the cut induced by e in G^{S_r}. Let i and j be the endvertices of e with i being further away from r. Let $U_i := \{k \in N : i \in P_{kr}^{\mathcal{T}^{N_r}}\}$, U_i contains all vertices who are connected to r via i, by definition it also contains i. The cut $(U_i, N_r \setminus U_i)$ has weight w_e, however, the cut defined by (i,j) in G^{S_r} contains only a subset of this edges, namely $(U_i \cap S, (N_r \setminus U_i) \cap S_r)$. Therefore, $w_e \geq w_{t_e}^{G^{S_r}}$ and we get

$$\sum_{e \in \mathcal{T}_{|S_r}} w_e \geq \sum_{e \in \mathcal{T}_{|S_r}} w_{t_e}^{G^{S_r}} \geq \min_{\mathcal{T}^{S_r} \in G^{S_r}} \sum_{e \in \mathcal{T}^{S_r}} w_{t_e}^{G^{S_r}}$$

observing that a tree with the same edges as $\mathcal{T}_{|S_r}$ may not be the minimum cut tree for G^{S_r}. □

Theorem 3.3. *The minimum cut tree game is superadditive.*

Proof. Let $\mathcal{T}^{S \cup T \cup r}$ be a minimum cut tree of the union $S \cup T$, we construct trees $\mathcal{T}_{|S_r}$ and $\mathcal{T}_{|T_r}$ restricted to vertices in S_r and T_r as in Construction 3.1, in this case $N = S \cup T$. Doing this, the weight of each edge of $\mathcal{T}^{S \cup T \cup r}$ is assigned to exactly one of the restricted trees depending on its endvertex which is further away from r. Now,

$$v(S) + v(T) \leq \sum_{e \in \mathcal{T}_{|S_r}} w_e + \sum_{e \in \mathcal{T}_{|T_r}} w_e = \min_{\mathcal{T}^{S \cup T \cup r} \text{: tree in } G^{S \cup T \cup r}} \sum_{e \in \mathcal{T}^{S \cup T \cup r}} w_{t_e}^{G^{S \cup T \cup r}} = v(S \cup T).$$

The first inequality follows from Lemma 3.2. □

However, a superadditive game cannot be transformed to a minimum cut tree game in general, i.e., there may be no graph G implying a minimum cut tree game with the same characteristic function. Let a three-player game be defined by $w(S) = 1$ for $|S| = 1$, $w(S) = 4$ for $|S| = 2$ and $w(S) = 10$ for $|S| = 3$. It differs from the unitgame only by the value of the grand coalition and it is superadditive. A graph implying a minimum cut tree game with the same values has to have weight 1 for every edge adjacent to the root, as $v(S) = w_{ir}$ for $S = \{i\}$. Furthermore, for any other edge (i, j) with $i, j \neq r$ we get $v(\{i, j\}) = 2 + w_{ij} + \min\{1, w_{ij}\}$ and this is not equal to 4 for $w_{ij} \neq 1$. Therefore, the only graph which yields the required values for coalitions with less than 3 players is the complete unweighted graph with four vertices. But, as we know from the unitgame $v(N) = |N|^2 = 9 \neq 10 = w(N)$.

Theorem 3.4. *Minimum cut tree games with at most three players are supermodular.*

Proof. If $|N| = 3$ coalitions are either disjoint or one contains the other or they have one player in common. Supermodularity follows from superadditivity for the first two cases. Let $S \cap T = \{i\}$. We use the restriction from Construction 3.1 with a modification. When we consider vertex i as an S-vertex, we assign it to the next vertex of S on the path from i to r in the minimum cut tree for the grand coalition. Its weight is not $(U_i, N_r \setminus U_i)$ but $(U_i \cap S, (N_r \setminus U_i) \cap S_r)$ and still the argumentation of Lemma 3.2 holds. For $\mathcal{T}_{|T_r}$ we restrict analogously. The only edge in $(U_i, N_r \setminus U_i)$ whose weight we assign twice is (i, r). As $v(\{i\}) = w_{ir}$, we have $v(S) + v(T) \leq v(S \cup T) + v(S \cap T)$. □

Supermodularity for games with more than three players is not given in general as can be seen in Table 1, in this game coalitions $S = \{1, 2, 4\}$ and $T = \{2, 3, 4\}$ violate the condition for supermodularity, as $v(S) + v(T) = 17 + 17 > 6 + 27 = v(S \cap T) + v(S \cup T)$.

3.2 Edge Player Variant

An edge player variant of the minimum cut tree game is implied by an undirected graph $G = (V, M)$ with vertex set V and edge set $\{j_1, \ldots, j_m\}$. Given a coalition $S \subseteq M$, G^S denotes the subgraph of G containing all vertices but only edges of players in S and $w_e^S = w_e$ for $j \in S$ whereas $w_e^S = 0$ else. The corresponding characteristic function is

$$v(S) = \min_{\mathcal{T}:\,\text{tree in}\,G} \sum_{e \in S} w_e \cdot l_e^{\mathcal{T}} = \min_{\mathcal{T}:\,\text{tree in}\,G} \sum_{e \in M} w_e^S \cdot l_e^{\mathcal{T}}.$$

Theorem 3.5. *The edge player variant is superadditive.*

Proof. Let w_e^S, w_e^T, and $w_e^{S \cup T}$ be defined as above, then $w_e^{S \cup T} = w_e^S + w_e^T$ and at least one of the latter summands is equal to 0. We get

$$\begin{aligned}
v(S \cup T) &= \min_{\mathcal{T}:\,\text{tree in}\,G} \sum_{e \in S \cup T} w_e^{S \cup T} \cdot l_e^{\mathcal{T}} \\
&= \min_{\mathcal{T}:\,\text{tree in}\,G} \Big(\sum_{e \in S \cup T} w_e^{S} \cdot l_e^{\mathcal{T}} + \sum_{e \in S \cup T} w_e^{T} \cdot l_e^{\mathcal{T}} \Big) \\
&= \min_{\mathcal{T}:\,\text{tree in}\,G} \Big(\sum_{e \in S} w_e^{S} \cdot l_e^{\mathcal{T}} + \sum_{e \in T} w_e^{T} \cdot l_e^{\mathcal{T}} \Big) \\
&\geq \min_{\mathcal{T}:\,\text{tree in}\,G} \sum_{e \in S} w_e^{S} \cdot l_e^{\mathcal{T}} + \min_{\mathcal{T}':\,\text{tree in}\,G} \sum_{e \in T} w_e^{T} \cdot l_e^{\mathcal{T}'} \\
&= v(S) + v(T) \qquad \square
\end{aligned}$$

3.3 Core Allocations

Theorem 3.6. *Let G be a graph defining the minimum cut tree game v, let $\mathcal{T}^{N_r}$ be any minimum cut tree of G and assign to each vertex $i \in N$ the weight of the first edge of the path from i to the root in $\mathcal{T}^{N_r}$, i.e., $x_i = w_{t_{ij}}^{G^{N_r}}$ where $(i,j) \in P_{ir}^{\mathcal{T}^{N_r}}$. Then $x = (x_i)_{i \in N}$ is a core allocation for v.*

Proof. Obviously, the allocation is efficient, i.e., $\sum_{i \in N} x_i = v(N)$. It remains to show that the coalitions are satisfied, i.e., $\sum_{i \in S} x_i \geq v(S)$ for all $S \subseteq N$. Therefore, we construct the restricted tree $\mathcal{T}_{|S_r}$ according to Construction 3.1. The weight of $\mathcal{T}_{|S_r}$ is equal to $\sum_{i \in S} x_i$ and thus by Lemma 3.2 not smaller than $v(S)$. $\square$

The allocation described in the theorem above will be called *Gomory-Hu cut allocation* in the following. As the minimum cut tree is not unique in general, the allocation is not unique either.

Theorem 3.7. *For minimum cut tree games it holds:*

(i) The core is never empty and a core element can be found in $O(n^4)$.

(ii) The core is a singleton if and only if there are no edges with positive weight which are not adjacent to the root.

(iii) The convex hull of any set of Gomory-Hu cut allocations is a subset of the core.

(iv) They are totally balanced.

(v) They are a proper subset of maximum flow games.

Proof. (i) Every graph has a corresponding minimum cut tree which yields a core allocation by Theorem 3.6. The complexity results from the complexity of the minimum cut tree algorithm by Gomory and Hu [GH61].

(ii) If $w_{ij} = 0$ for all $i, j \neq r$ then $v(N) = \sum_{i \in N} w_{ir}$. A core allocation satisfies $x_i \geq v(\{i\}) = w_{ir}$ and we get $\sum_{i \in N} w_{ir} \leq \sum_{i \in N} x_i = x(N) = v(N) = \sum_{i \in N} w_{ir}$. Therefore, $x_i = w_{ir}$ is the only core allocation.

Let $w_{ij} > 0$ for some $i, j \neq r$, let $\mathcal{T}^{N_r}$ be a minimum cut tree for N_r and x be the corresponding Gomory-Hu cut allocation. There are three cases for the position of i, j and r: $j \in P_{ir}^{\mathcal{T}^{N_r}}$ (or vice versa) or $r \in P_{ij}^{\mathcal{T}^{N_r}}$ or none of the vertices is on a path between the other two. In the first case w_{ij} is a component of x_i and in the other cases of x_i and x_j. Define x' by $x'_i = x_i - w_{ij}$, $x'_j = x_j + w_{ij}$ and $x'_k = x_k$ else. We only have to show that $x'(S) \geq v(S)$ for $i \in S$ and $j \notin S$. Following Construction 3.1 the argument of Lemma 3.2 still holds if we do not assign weight w_{ij} to vertex $i \in S$ as the cut in G^{S_r} would not contain (i, j) anyway. Every convex combination of x and x' is an element of the core as well.

(iii) Given any minimum cut tree in N_r, the corresponding Gomory-Hu cut allocation is in the core by Theorem 3.6. As the core is convex, the result follows.

(iv) The minimum cut tree game is balanced as its core is never empty and every subgame of a minimum cut tree game is a minimum cut tree game and therefore, it is balanced itself.

(v) The class of totally balanced games is equivalent to the class of maximum flow games [KZ82] and there are totally balanced games which cannot be transformed to a minimum cut tree game, e.g., the superadditive game above. □

We introduce a core allocation for the edge player variant which also leads to another core allocation of the original game.

Theorem 3.8. *Let G be a graph defining the edge player variant of the minimum cut tree game v, let $\mathcal{T}$ be any minimum cut tree of G and assign to each edge $e \in M$ its weight multiplied with the number of cuts it is in, i.e., $x_e = w_e \cdot l_e^{\mathcal{T}}$. Then $x = (x_e)_{e \in M}$ is a core allocation for v.*

Proof. The allocation is efficient. The payoff to a coalition S is $x(S) = \sum_{e \in S} w_e \cdot l_e^{\mathcal{T}}$ which is not smaller than the value of a minimum cut tree, $v(S) = \min_{\mathcal{T}' : \text{tree in } G} \sum_{e \in S} w_e \cdot l_e^{\mathcal{T}'}$. $\square$

Theorem 3.9. *Let G be a graph defining the minimum cut tree game v, let $\mathcal{T}^N$ be any minimum cut tree of G and assign to each vertex $i \in N$ a share of the weights of its incident edges multiplied with the number of cuts they are in, i.e., $x_i = w_{ir} \cdot l_{ir}^{\mathcal{T}} + \sum_{j \neq i,r} \lambda_{ij} \cdot w_{ij} \cdot l_{ij}^{\mathcal{T}}$ with $\lambda_{ij} + \lambda_{ji} = 1$ and $0 \leq \lambda_{ij} \leq 1$. Then $x = (x_i)_{i \in N}$ is a core allocation for v which is denoted as* Gomory-Hu committee allocation.

Proof. Let w be the edge player variant of the minimum cut tree game and let y be the allocation in its core as given in Theorem 3.9. Now, let N be the set of vertex players who play *committee games* (cf. Curiel et al [CDT89]) on the edges of G. An edge joins a coalition if and only if its two endvertices join the coalition, this makes the endvertices so called *veto players*. If one endvertex of the edge is the root then the other endvertex is a *dictator*. Observe that we have the same power structure as in the vertex player game – for every coalition $S \subseteq N$ its value is the weight of the minimum cut tree in the graph containing all edges with both endvertices in S_r. The reward y_e of an edge in the edge player variant can be shared in an arbitrary proportion among the corresponding veto players or dictators. $\square$

The Gomory-Hu cut allocation is $x_i = |N|$ for all $i \in N$ for the unitgame. For the second game the Gomory-Hu cut allocation is $x = (7, 6, 7, 7)$, whereas Gomory-Hu committee allocations for $\lambda_{ij} = \frac{1}{2}$ and $\lambda_{4i} = 1$ for all $i, j \in N$ are $(4, 4, 4, 15)$ and $(7, 6, 6.5, 7.5)$, respectively.

In Theorem 3.7 we concluded that every minimum cut tree game is a maximum flow game and thus, a core allocation can be found in $O(n^3)$ once the game is transformed where n is the number of vertices in the maximum flow game. Kalai and Zemel [KZ82] transformed a totally balanced game – as the minimum cut tree game – to a maximum flow game as a minimum game of additive games where each additive game consists of two vertices and n parallel edges corresponding to the players. The minimum game of these additive games is their connection in series. In our case this transformation results in a huge number of vertices.

Theorem 3.10. *A supermodular minimum cut tree game of n players can be represented by the minimum game of at most $\binom{n}{n/2}$ additive games. This bound is sharp.*

Proof. Given a minimum cut tree game with n players, we pick $\binom{n}{n/2}$ marginal vectors and build additive games with each of them. We have to show that this number is sufficient and necessary to represent the original game. As for supermodular games all marginal vectors are in the core, they fulfill $\sum_{i \in S} m_i(v) = m(S) \geq v(S)$ for every additive game and also for the minimum game itself. It remains to show that $m(S) = v(S)$ for at least one additive game, for supermodular games this is the case for every permutation where the members of S come first. Hence, we need to have every coalition at least once at the beginning of a permutation. We can build chains of coalitions, i.e., a one-player-coalition is a subset of a two-player coalition and so on. By the theorem of Dilworth, $\binom{n}{n/2}$ chains are needed.

The sharpness of the bound can be seen in the unitgame where $v(S) = |S|^2$. For every player there has to be an edge with weight 1 in at least one additive game. In all other additive game her edge must not have weight less than 1. To induce the value for a two-player coalition, the players have to have weight 4 in at least one additive game. This can never be covered in one additive game for two two-player coalitions, otherwise their three- or four-player union would have weight less than 9, contradicting the requirement $v(S) = |S|^2$. Therefore we need at least as many additive games as there are two-player coalitions. The maximum number of coalitions with the same size is $\binom{n}{n/2}$ for coalitions with $\lfloor \frac{n}{2} \rfloor$ players. □

3.4 Special Cases

Lemma 3.11. *For special cases the weight of a minimum cut tree or an upper bound of it can be found as follows:*

(i) Let G be a graph with cut vertex v_c, i.e., the deletion of v_c increases the number of components of G. The graph can be decomposed into components not containing any cut vertex, observe that the cut vertices of the original graph appear in more than one component. Then a minimum cut tree can be found by composing the minimum cut trees of the components.

(ii) Let G be a tree graph, then a minimum cut tree is equal to the graph itself and its weight is the same as well. This holds for forest graphs as well if edges with weight 0 are added to connect the components.

(iii) *Let G be a cycle graph with m edges, then a minimum cut tree is equal to the maximum spanning tree and its weight is $\sum_{e\in G} w_e + (m-2)\min_{e\in G} w_e$.*

(iv) *Let $G = (N \cup r, E)$ be a graph with root r and let $w_{ij} \leq \min\{w_{ir}, w_{jr}\}$ hold for all $i, j \neq r$, then the minimum cut tree is a star tree with center r and its weight is $\sum_{i\in N} w_{ir} + 2\sum_{i<j\in N} w_{ij}$*

(v) *Let G be an unweighted graph with m edges and let δ_i be the degree of vertex i, the weight of a minimum cut tree is at most $2\cdot m - \Delta$ or equivalently $\sum_{i\in V}\delta_i - \Delta$ where Δ is the maximum degree of a vertex in G.*

Proof. (i) A minimum cut between two vertices does not contain edges of more than one component. If it contained edges of at least two components than a proper subset of these edges would disconnect the two vertices as well and the weight of this subset would be even smaller. Therefore, minimum cut trees of components are pairwise independent.

(ii) The tree is the result of the algorithm of Gomory and Hu if in every step the minimum cut between two vertices adjacent in the original graph is calculated. This minimum cut consists only of the edge in the original graph.

(iii) In a minimum cut tree there are $n-1$ cuts, every cut in a cycle graph contains at least two edges. Let $f = (i, j)$ be an edge with minimum weight and apply the algorithm of Gomory and Hu, in every iteration separate two vertices adjacent in the original graph, do not choose i and j in the same step. A minimum cut between these vertices contains the edge connecting them and a second edge, preferably an edge with minimum weight, namely, f. At the end, f contributes to $n-1$ minimum cuts and the other edges are contained in one cut.

(iv) Apply the algorithm of Gomory and Hu and choose to separate r and an arbitrary vertex k. We show that a minimum cut between them is the cut separating k from all other vertices. Assume the minimum cut separates $U \subseteq N$ from $\overline{U} := N_r \setminus U$, with $k \in U$ and $|U| > 1$. The weight of $(U, \overline{U})$ is

$$\sum_{i\in U,\, j\in\overline{U}} w_{ij} = w_{kr} + \sum_{i\in U\setminus k} w_{ir} + \sum_{j\in\overline{U}\setminus r} w_{kj} + \sum_{i\in U\setminus k,\, j\in\overline{U}\setminus r} w_{ij}.$$

The weight of the cut $(k, N_r \setminus k)$ is

$$\sum_{j\in N_r} w_{kj} = w_{kr} + \sum_{i\in U\setminus k} w_{ki} + \sum_{j\in\overline{U}\setminus r} w_{kj}$$

and is not greater than the weight of $(U,\overline{U})$ as $w_{ik} \leq w_{ir}$ for all $i \in N$.

(v) Let i be a vertex with maximum degree Δ, now consider the star tree with center i which is not necessarily a minimum cut tree. Every edge which is non-incident to i is contained in two cuts, the Δ edges incident to i are contained in one cut and the edges of the star tree which are not edges of G have weight 0 anyway. Therefore, the star tree induces total cut weight $2 \cdot (m - \Delta) + \Delta = 2 \cdot m - \Delta$. The remainder follows as $2 \cdot m = \sum_{i \in V} \delta_i$.

□

Theorem 3.12. *Let G be an unweighted graph, let S and T be two coalitions and let $G^{S_r \cap T_r}$ be the graph induced by their intersection. We define $s := |S_r|$, $t := |T_r|$, $c := |V(G^{S_r \cap T_r})|$, and $d := |E(G^{S_r \cap T_r})|$. The number of isolated vertices in $G^{S_r \cap T_r}$ is denoted by h and the number of disjoint paths by k. If the following conditions hold*

(i) S and T have at least three members each, neither of them is contained in the other

(ii) G^{S_r} and G^{T_r} are cycle graphs

(iii) $G^{S_r \cap T_r}$ consists only of disjoint paths or isolated vertices and $d \geq 1$ and $d + h \geq 2$

(iv) $G^{S_r \cap T_r} = G^{S_r} \cup G^{T_r}$ (i.e. there are no arcs having one vertex in $S \setminus T$ and the other in $T \setminus S$)

then the graph does not imply a supermodular game.

Proof. Observe that $c = h + k + d$, there are h vertices with degree 4, $2k$ vertices with degree 3 and $(s - c) + (t - c) + (d - k)$ vertices with degree 2 in $G^{S_r \cup T_r}$. Note that these numbers would change if (iv) did not hold. It follows from (ii) and Lemma 3.11 that $v(S) = 2(s - 1)$, $v(T) = 2(t - 1)$, and $v(S \cap T) = d$. From the same lemma we get an upper bound for $v(S \cup T)$ and therefore,

$$\begin{aligned} v(S \cap T) + v(S \cup T) &< d + 2 \cdot (s + t - 2c + d - k) + 3 \cdot 2k + 4 \cdot h - \Delta \\ &= 2s + 2t + 3d + 4h + 4k - 4c - \Delta \\ &= 2s + 2t - d - \Delta \end{aligned}$$

where $\Delta = 3$ if $h = 0$ and $\Delta = 4$ else. On the other hand $v(S) + v(T) = 2s + 2t - 4$. As $d + \Delta > 4$ by condition (iii), we get $v(S \cap T) + v(S \cup T) < v(S) + v(T)$. □

There is only one non-isomorphic graph for four players defining a non-supermodular game, namely, the unweighted version of the graph in Figure 1 and any of its vertices can be the root. If a graph contains this graph as an induced subgraph it does not imply a supermodular game. For five players there are 15 non-isomorphic non-supermodular graphs.

Theorem 3.13. *If G satisfies $w_{ij} \leq \min\{w_{ir}, w_{jr}\}$ for all $i, j \neq r$ then the game is supermodular and the Shapley value corresponds to the Gomory-Hu cut allocation.*

Proof. If $w_{ir} = 0$ for a vertex i it follows that $w_{ij} = 0$ for all $j \in N$, i.e., the vertex is isolated in the original graph. We assume, w.l.o.g., that $w_{ir} > 0$ for all $i \in N$. It follows from Lemma 3.11 that a minimum cut tree of a coalition S is a star tree with center r. The game is supermodular, because

$$\begin{aligned} v(S) + v(T) &= \sum_{i \in S} w_{ir} + 2 \sum_{i<j \in S} w_{ij} + \sum_{i \in T} w_{ir} + 2 \sum_{i<j \in T} w_{ij} \\ &= \sum_{i \in S \cup T} w_{ir} + 2 \sum_{i<j \in S \cup T} w_{ij} + \sum_{i \in S \cap T} w_{ir} + 2 \sum_{i<j \in S \cap T} w_{ij} \\ &= v(S \cup T) + v(S \cap T). \end{aligned}$$

If a player i enters a coalition S she adds $w_{ir} + 2\sum_{j \in S} w_{ij}$ to $v(S)$. In half of the permutations of N player i enters a coalition already including player j, in this case a part of her contribution is $2 \cdot w_{ij}$ which makes w_{ij} in average. Player i always adds w_{ir} to the value of a coalition. Therefore the Shapley value of player i is $w_{ir} + \sum_{j \in N} w_{ij}$ which equals the Gomory-Hu cut allocation. □

A *cactus graph* is a graph whose cycles are edge-disjoint. Special cases are tree graphs and cycle graphs.

Theorem 3.14. *For the class of cactus graphs it holds:*

(i) Cactus graphs imply supermodular games.

(ii) In tree graphs the Shapley value corresponds to the Gomory-Hu committee allocation with $\lambda_{ij} = \frac{1}{2}$ for all $i, j \in N$.

(iii) Let $C_1, \ldots, C_k$ be the cycles not containing r and $C_{k+1}, \ldots, C_l$ the cycles containing r. The edge with minimum weight in a cycle C_i is denoted by f_i. Then the Shapley

value is

$$\phi(i) = w_{ir} + \frac{1}{2}\sum_{j\neq r} w_{ij} + \sum_{C_h: i\in V(C_h), h\leq k} \frac{|V(C_h)|-2}{|V(C_h)|}\cdot w_{f_h} + \sum_{C_h: i\in V(C_h), h>k} \frac{|V(C_h)|-2}{|V(C_h)|-1}\cdot w_{f_h}.$$

Proof. (i) It follows from Lemma 3.11 that the value of a coalition S with cactus graph G^{S_r} is equal to $\sum_{i\in S} w_{ir} + \sum_{i,j\in S} w_{ij} + \sum_{C:\text{ cycle in } G^{S_r}} (n-2) \min_{e\in E(C)} w_e$. Observing that a cycle in the graph of one coalition is in the union of the coalitions and a cycle in the graph of two coalitions is also in their intersection, the result follows.

(ii) Following from Lemma 3.11, the value of a coalition S is equal to $\sum_{e\in G^{S_r}} w_e$ or equivalently $\sum_{i\in S} w_{ir} + \sum_{i<j\in S} w_{ij}$. If player i enters a coalition already including player j she adds $w_{ir} + \sum_{j\in S} w_{ij}$ to $v(S)$. With the same argument as in Theorem 3.13 the average contribution to the value of a coalition is $w_{ir} + \frac{1}{2}\sum_{j\in N} w_{ij}$. Hence, the Shapley value equals the Gomory-Hu committee allocation.

(iii) A player i always contributes w_{ir} to the value of a coalition, she adds w_{ij} if and only if player j is already in the coalition, i.e. in half of the cases. Moreover, whenever player i enters a coalition and therewith closes a cycle C in G^{S_r} she adds the weight of a minimum weight edge in $E(C)$ $|V(C)|-2$ times. This happens in $\frac{1}{|V(C)|}$ of the cases if r is not in C and in $\frac{1}{|V(C)|-1}$ of the cases else.

□

Planar graphs do not imply supermodular games in general as can be seen in the graph of Figure 1.

4 Conclusion

We introduced a cooperative game based on the minimum cut tree problem and showed how a core solution can be obtained. We started our investigations about the Shapley value on special graphs. Our future research concerns the Shapley value for general graphs as well as related cooperative cost games and competitive reward games.

Acknowledgements

The research of Anne Schwahn has been partially supported by the Center of Mathematical and Computational Modelling of the University of Kaiserslautern, $(CM)^2$.

References

[BHH01] P. Borm, H. Hamers, and R. Hendrickx. Operations research games: a survey. *Sociedad de Estadística e Investigación Operativa TOP*, 9(2):139–216, 2001.

[CDT89] I. Curiel, J. Derks, and S. Tijs. On balanced games and games with committee control. *OR Spectrum*, 11(2):83–88, 1989.

[Cur97] I. Curiel. *Cooperative game theory and applications: Cooperative games arising from combinatorial optimization problems*. Kluwer, Boston, Massachusetts, USA, 1997.

[DP94] X. Deng and C. H. Papadimitriou. On the complexity of cooperative solution concepts. *Mathematics of Operations Research*, 19(2):257–266, 1994.

[GH61] R. E. Gomory and T. C. Hu. Multi-terminal network flows. *Journal of SIAM*, 9(4):551–570, 1961.

[GR98] A. V. Goldberg and S. Rao. Beyond the flow decomposition barrier. *Journal of the ACM*, 45(5):783–797, 1998.

[KZ82] E. Kalai and E. Zemel. Totally balanced games and games of flow. *Mathematics of Operations Research*, 7(3):476–478, 1982.

[Sha53] L. S. Shapley. A value for n-person games. In *Contributions to the theory of games II*, pages 307–317. Princeton University Press, New Jersey, USA, 1953.