On Cooperative Connection Situations Where the Players Are Located at the Edges

Stefano Moretti[(✉)]

Université Paris-Dauphine, PSL Research University, CNRS, UMR [7243], 75016 Paris, France
stefano.moretti@dauphine.fr

Abstract. In classical cooperative connection situations, the agents are located at some nodes of a network and the cost of a coalition is based on the problem of finding a network of minimum cost connecting all the members of the coalition to a source.

In this paper we study a different connection situation with no source and where the agents are the edges, and yet the optimal network associated to each coalition (of edges) is not fixed and follows a cost-optimization procedure. The proposed model shares some similarities with classical minimum cost spanning tree games, but also substantial differences, specifically on the appropriate way to share the costs among the agents located at the edges. We show that the core of these particular cooperative games is always non-empty and some core allocations can be easily computed.

Keywords: Coalitional games · Connection situations
Cost allocation protocols · Core

1 Introduction

This paper deals with an alternative class of cooperative cost games defined on *minimum cost spanning tree* (*mcst*) situations. A (classical) mcst situation arises in the presence of a group of agents that are willing to be connected as cheap as possible to a source (e.g., a supplier of a service, if the agents are computers, or a water purifier, if the agents represent farms in a drainage system). Since links are costly, agents evaluate the opportunity of cooperating in order to reduce costs: if a group of agents decides to cooperate, a spanning network minimizing the total cost of connection of all the agents in the group with the source (i.e., a mcst) is constructed, and the total cost of the mcst must be shared among the agents of the group. The problem of finding an mcst can be easily solved

by means of alternative algorithms proposed in the literature (*e.g.*, the Kruskal algorithm [9] or the Prim algorithm [14]). However, finding an mcst does not guarantee that it is going to be really implemented: the agents must agree on the way the cost of the mcst must be shared, and then a cost allocation problem must be addressed. This cost allocation problem was introduced in [5] and has been studied with the aid of cooperative game theory since the basic paper [3]. After this seminal paper, many cost allocation methods have been proposed in the literature on *mcst games* (see, for instance, [2,4,6,7,10,18]).

More recently, alternative connection situations have been introduced where the focus of interest of rational agents are the edges of a network. For instance, in [8], the agents demand a connection between certain nodes of a network, using a single link or via longer paths, and it is assumed that the set of implemented edges is exogenously fixed and may be "redundant" (see also [11] for an alternative approach considering redundant links). A still different class of games has been studied in [1], where the players are the edges of a graph and a coalition of edges gets value one if it is a connected component in the graph, and zero otherwise. All the aforementioned models deal with coalitional games where the cost of a coalition is fixed, or its computation is based on a structural property of the graph. In this paper, we investigate a particular subclass of the family of games introduced in [12], where the complexity of solutions for cooperative games defined on matroids has been extensively investigated. In our framework, the players are the edges of a weighted undirected graph, and the cost associated to each coalition (of edges) is the one of an optimal network connecting the endpoints of the edges in the coalition. The model we study in this paper is quite natural in a context where different service providers wish to satisfy a demand of economic exchange between pairs of nodes of a network (*e.g.*, an airline network, a content delivery network on the web, a telecommunication network, etc.). For example, a very common strategic problem for airlines participating in pooled flights is to decide how to allocate joint revenues and costs. Consider, for instance, three airports 1, 2 and 3 which are connected to each other by three different flight operators, each providing an air transport service on a different single connection between two airports: an operator over the link $1-2$, another one over the link $1-3$ and a still different one over $2-3$ (see Fig. 1 for a graphical representation of this connection situation). Clearly, implementing each flight connection between two airports need not be the best strategy. In fact, the implementation of only two links would be sufficient to guarantee the connection among the three airports at a lower cost (provided that the capacity constraints imposed by the flight vectors satisfy the demand for the service). Consequently, the decision of the flight operators on whether to cooperate for the implementation of an optimal airline network, also depends on the allocation method used to share the monetary savings generated by this cooperation.

The structure of the paper is as follows. We start in the next section with some basic definitions on cooperative games and graphs. Then, in Sect. 3 we introduce the proposed model, namely, a *Link Connection (LC) situation*, and the associated (coalitional) LC game, and we study their properties. In Sect. 4 we

study a procedure to decompose an LC game as a positive linear combination of "simple" LC games, which are defined on weighted networks with weights equal to 0 or 1. Section 5 deals with the problem of finding allocations in the core of an LC game. Section 6 concludes with some research directions.

2   Preliminaries and Notations

A *coalitional cost game* (or, shortly, a *cost game*) is a pair (N, c), where $N = \{1, \ldots, n\}$ denotes the set of players and $c : 2^N \to \mathbb{R}$ is the *characteristic function*, (by convention, $c(\emptyset) = 0$). A group of players $S \subseteq N$ is called *coalition* and $c(S)$ is the *cost* incurred by coalition S. If the set N of players is fixed, we identify a cost game (N, c) with its characteristic function c and we denote as $\mathcal{CG}^N$ the class of all cost games with N as the set of players. For a coalition $S \subseteq N$, we shall denote by s or $|S|$ its cardinality.

A cost game (N, c) is said to be *subadditive* if it holds that $c(S \cup T) \leq c(S) + c(T)$ for all $S, T \subseteq N$ such that $S \cap T = \emptyset$. Moreover, a game (N, c) is said to be *concave* or *submodular* if it holds that $c(S \cup T) + c(S \cap T) \leq c(S) + c(T)$ for all $S, T \subseteq N$. Equivalently, a game (N, c) is said to be concave if it holds that $m_i^c(S) \geq m_i^c(T)$ for all $i \in N$ and all $S \subseteq T \subseteq N \setminus \{i\}$, and where $m_i^c(S) = c(S \cup \{i\}) - c(S)$ is the *marginal contribution* of player i to $S \cup \{i\}$. Given a cost game c, an *allocation* is a vector $x \in \mathbb{R}^N$ such that the *efficiency* condition $\sum_{i \in N} x_i = c(N)$ is satisfied.

An important subset of allocations is the *core*, which represents a classical "solution set" for TU-games. The core of game (N, c) is defined as the set of allocation vectors for which no coalition has an incentive to leave the grand coalition N, precisely,

$$Core(c) = \{x \in \mathbb{R}^N : \sum_{i \in N} x_i = c(N), \sum_{i \in S} x_i \leq c(S) \quad \forall S \subset N\}.$$

A (one-point) *solution* for cost games in $\mathcal{CG}^N$ is a map $\psi : \mathcal{CG}^N \to \mathbb{R}^N$ assigning to each cost game c in $\mathcal{CG}^N$ an $|N|$-vector of real numbers. The Shapley value [15] ϕ is a special solution assigning to each cost game (N, c) an $|N|$-vector computed according to the following formula:

$$\phi_i(c) = \sum_{S \in 2^{N \setminus \{i\}}} p_s m_i^c(S) \tag{1}$$

for each $i \in N$ and with $p_s = \frac{1}{n\binom{n-1}{s}}$ for each $s = 0, \ldots, |N| - 1$.

We provide now some notations about graphs. An undirected *graph* or *network* is a pair $\langle V, E \rangle$, where V is a finite set of vertices or nodes and E is a set of edges e of the form $\{i, j\}$ with $i, j \in V$, $i \neq j$. Given a graph $\langle V, E \rangle$, let $V(E) = \bigcup_{\{i,j\} \subseteq E} \{i, j\}$ be the set of vertices (of the edges) in E. A *path* between i and j in a graph $\langle V, E \rangle$ is a sequence of nodes $(i_0, i_1, \ldots, i_k)$, where $i = i_0$ and $j = i_k$, $k \geq 1$, such that $\{i_t, i_{t+1}\} \in E$ for each $t \in \{0, \ldots, k-1\}$ and such

that all these edges are distinct. Two nodes $i, j \in V$ are said to be *connected* in $\langle V, E \rangle$ if $i = j$ or there exists a path between i and j in $\langle V, E \rangle$. A *component* of $\langle V, E \rangle$ is a maximal subset of V with the property that any two nodes in this subset are connected. The set $\mathcal{P}_E$ of all components in $\langle V, E \rangle$ is a partition of V. A graph $\langle V, E \rangle$ is *connected* if for each $i, j \in V$ with $i \neq j$ there exists a path between i and j in $\langle V, E \rangle$. A *cycle* in $\langle V, E \rangle$ is a path from i to i for some $i \in V$. A path $(i_0, i_1, \ldots, i_k)$ is *without cycles* if there do not exist $a, b \in \{0, 1, \ldots, k\}$, $a \neq b$, such that $i_a = i_b$. A graph where all paths are without cycles is called *forest*, and a forest that is also connected is called *tree*.

3   Link Connection Games

A *link connection* (*LC*) *situation* is defined as a triple $\mathcal{L} = \langle V, E, w \rangle$, where $\langle V, E \rangle$ is an undirected graph and $w : E \to [0, \infty)$ is a *weight function*, that is a map assigning to each edge $\{i, j\} \in E$ a non-negative number $w(\{i, j\})$ (in order to simplify our notation, an edge $\{i, j\}$ will be also denoted as ij, whenever no confusion can arise). Each edge $\{i, j\} \in E$ identifies an economic entity (*e.g.*, a service provider) aimed to satisfy a demand of connection between nodes i and j for the fruition of a service (*e.g.*, a communication channel in a telecommunication network, an on-line service on the web, a flight in an airlines network, etc.). A service connection between i and j can be implemented directly at a cost $w(\{i, j\})$, or indirectly, via a path between i and j in $\langle V, E \rangle$ using edges whose connection is already activated. Differently stated, once a connection between two nodes $\{i, j\} \in E$ is activated (at a cost $w(\{i, j\})$), the same connection can be exploited to implement the delivery of other services with no extra-costs. Each service provider $\{i, j\} \in E$ may decide whether to directly satisfy the request between i and j (at the cost $w(\{i, j\})$) or, in alternative, to cooperate with other service providers in order to exploit the connection already implemented.

In the following, the cost of a *network* $\langle V, L \rangle$ in an LC situation $\mathcal{L} = \langle V, E, w \rangle$ and with $L \subseteq E$ is denoted by $w(L) = \sum_{e \in L} w(e)$. Given an LC situation $\mathcal{L} = \langle V, E, w \rangle$, it is possible to determine at least one *minimum cost spanning forest* (*mcsf*) $\langle V, \Gamma \rangle$ for $\mathcal{L}$, *i.e.* a network without cycles of minimum cost with $\Gamma \subseteq E$ and such that i and j are connected in $\langle V, E \rangle$ if and only if they are connected in $\langle V, \Gamma \rangle$, for each $i, j \in V$. So, the set of components $\mathcal{P}_\Gamma$ in $\langle V, \Gamma \rangle$ coincides with the set of components $\mathcal{P}_E$ in $\langle V, E \rangle$. If $\langle V, E \rangle$ is a connected graph, then a mcsf $\langle V, \Gamma \rangle$ for $\mathcal{L}$ is a tree and it is called *minimum cost spanning tree* (*mcst*) for $\mathcal{L}$. In the following we will also use the notation $\mathcal{L}_{|S} = \langle V(S), S, w_{|S} \rangle$ to denote the (sub-) LC situation such that $w_{|S} : S \to \mathbb{R}$ with $w_{|S}(e) = w(e)$ for each $e \in S$ (here $V(S) := \bigcup_{e \in S} e$ is the set of vertices of the edges belonging to S).

Definition 1. Given an LC situation $\mathcal{L} = \langle V, E, w \rangle$, the corresponding *LC game* is defined as the cost game (E, c), where E is the set of players (service providers, located at the edges of the network) and the cost $c(S)$ of each coalition $S \in 2^N \setminus \{\emptyset\}$, is as follows:

$$c(S) = \min\{w(\Gamma) | \langle V, \Gamma \rangle \text{ is a spanning forest for } \langle V(S), S, w_{|S} \rangle\}.$$

Remark 1. In Definition [1], and in the remaining of this paper, we are motivated to study a cooperative situation where the cost of coalition $S \subseteq N$ does not depend on the actions adopted by service providers in $N \setminus S$. Therefore we make the assumption that the service providers of a coalition S can only implement the services over the edges in S, and are not allowed to use connections in the complementary coalition.

Example 1. Consider the LC situation depicted in Fig. [1]. The corresponding LC game $(E = \{12, 13, 23\}, c)$ is such that $c(\{12\}) = 4$, $c(\{13\}) = 2$, $c(\{23\}) = 3$, $c(\{12, 23\}) = 7$, $c(\{12, 13\}) = 6$, $c(\{13, 23\}) = 5$ and $c(\{12, 13, 23\}) = 5$. Notice that the core of the game (E, c) is $Core(c) = \{x \in \mathbb{R}^E : \sum_{i \in E} x_i = 5, \ 4 \geq x_{12} \geq 0, \ 2 \geq x_{13} \geq -2, \ 3 \geq x_{23} \geq -1\}$.
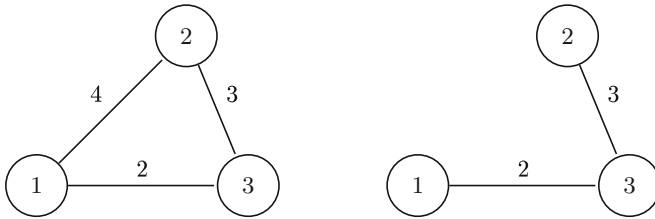

Fig. 1. An LC situation $\mathcal{L} = \langle V, E, w \rangle$, with $V = \{1, 2, 3\}$, $E = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$, $w(1, 2) = 4$, $w(1, 3) = 2$, $w(2, 3) = 3$ (left side) and the corresponding mcst (right side).

Proposition 1. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation. The corresponding LC game (E, c) is subadditive.*

Proof. The proof is straightforward and therefore is omitted.   □

It is well known that concave games have a non-empty core, which also contains the Shapley value [16]. The following example shows that, in general, LC games are not concave, so we cannot use the concavity argument to guarantee that the core of LC games is non-empty.

Example 2 (LC games are not necessarily concave). Consider the LC situation $\mathcal{L} = \langle V, E, w \rangle$ depicted in Fig. [2], with $E = \{12, 13, 23, 24, 34\}$. Clearly, the cost of many coalitions of edges is simply the sum of the costs of the individual edges (*e.g.*, $c(13, 24) = 3$). For other coalitions, the construction of spanning forests determine some extra monetary savings (*e.g.*, the spanning tree $\Gamma = \{12, 13, 24\}$ is the optimal configuration which guarantees the connection of the adjacent nodes of all possible links in the graph at a total cost of 4). Notice that the corresponding LC game is not concave. Consider the coalitions $S = \{23, 34\}$ and $T = \{12, 13, 23, 34\}$. Then, $c(S \cup 24) = 6$, $c(T \cup 24) = 4$, and $c(S) = 12$, $c(T) = 6$. So, $m_{24}^c(S) = -6$ and $m_{24}^c(T) = -2$.
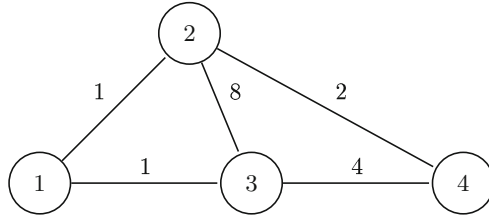
Fig. 2. An LC situation whose corresponding LC game is not concave.

Table 1. The LC game corresponding to the LC situation of Fig. 2. All omitted coalitions have an additive cost, that is $c(S) = \sum_{e \in S} w(e)$.

S	12,13,23	23,24,34	12,13,24,24	12,13,23,34	12,13,23,24	23,24,34,13	23,24,34,12	E
$c(S)$	2	6	4	6	4	7	7	4

Notice also that, according to relation (1), the Shapley value of game (E, c) is $(\phi_{12}(c), \phi_{13}(c), \phi_{24}(c), \phi_{34}(c), \phi_{23}(c)) = (-\frac{16}{15}, -\frac{16}{15}, -\frac{1}{15}, \frac{29}{15}, \frac{64}{15})$, which is not an element of $Core(c)$, since $\phi_{24}(c) + \phi_{34}(c) + \phi_{23}(c) > 6 = c(24, 34, 23)$.

Consider an LC situation $\mathcal{L} = \langle V, E, w \rangle$. Nodes $i, j \in V$ are called $\mathcal{L}$-*connected* if $i = j$ or if there exists a path $(i_0, \ldots, i_k)$ from i to j in $\langle V, E \rangle$, with $w(\{i_s, i_{s+1}\}) = 0$ for every $s \in \{0, \ldots, k-1\}$. A $\mathcal{L}$-*component* of $\mathcal{L}$ is a maximal subset of V with the property that any two nodes in this subset are $\mathcal{L}$-connected. We denote by $\mathcal{C}(\mathcal{L})$ the set of all the $\mathcal{L}$-components. Given a component T in $\langle V, E \rangle$, let $\mathcal{C}_T(\mathcal{L}) = \{C \subseteq T : C$ is a $\mathcal{L}$-component$\}$ be the set of all $\mathcal{L}$-components contained in T (notice that $\mathcal{C}_T(\mathcal{L})$ forms a partition of T and that $\mathcal{C}_E(\mathcal{L}) = \mathcal{C}(\mathcal{L})$). Similarly, for each non-empty coalition $S \subseteq E$, $\mathcal{C}_T(\mathcal{L}_{|S}) = \{C \subseteq T : C$ is a $\mathcal{L}_{|S}$-component$\}$ denotes the set of all $\mathcal{L}_{|S}$-components (*i.e.*, in the restriction $\mathcal{L}_{|S} = \langle V(S), S, w_{|S} \rangle$) contained in T.

An LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ such that $w'(e) \in \{0, 1\}$ for each $e \in E$ is said to be *simple*. Following the decomposition in [4] for classical connection situations, the next lemma shows that an LC situation can be decomposed as a sum of simple LC situations. We first need some further notations. Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation. We define the set Σ_E of *linear orders* on E as the set of all bijections $\sigma : \{1, \ldots, |E|\} \to E$. For each $\sigma \in \Sigma_E$ define the simple LC situation $\mathcal{L}^{\sigma,k} = \langle V, E, e^{\sigma,k} \rangle$, for each $k \in \{1, 2, \ldots, |E|\}$, where the vector $e^{\sigma,k} \in \{0, 1\}^E$, is such that $e^{\sigma,1}(\sigma(j)) = 1$ for all $j \in \{1, 2, \ldots, |E|\}$, and for each $k \in \{2, \ldots, |E|\}$

$$e^{\sigma,k}(\sigma(1)) = e^{\sigma,k}(\sigma(2)) = \ldots = e^{\sigma,k}(\sigma(k-1)) = 0$$
$$\text{and} \tag{2}$$
$$e^{\sigma,k}(\sigma(k)) = e^{\sigma,k}(\sigma(k+1)) = \ldots = e^{\sigma,k}(\sigma(|E|)) = 1.$$

Lemma 1. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation. Let $\sigma \in \Sigma_E$ be such that $w(\sigma(1)) \leq w(\sigma(2)) \leq \ldots \leq w(\sigma(|E|))$. Then we have that*

$$w = w(\sigma(1))e^{\sigma,1} + \sum_{k=2}^{|E|} \left(w(\sigma(k)) - w(\sigma(k-1)) \right) e^{\sigma,k}. \tag{3}$$

Proof. The proof is very similar to the decomposition procedure introduced in [4]. □

Example 3 (follows Example 2). Consider the LC situation of Example 2 and the ordering $\sigma = (\{1,2\}, \{1,3\}, \{2,4\}, \{3,4\}, \{2,3\})$. Notice that $w(\sigma(1)) \leq \ldots \leq w(\sigma(5))$. According to Lemma 1 we have that

$$w = e^{\sigma,1} + 0e^{\sigma,2} + e^{\sigma,3} + 2e^{\sigma,4} + 4e^{\sigma,5}$$

where the weight vectors $e^{\sigma,1}, \ldots, e^{\sigma,5}$ are such that $e^{\sigma,1} = (1,1,1,1,1)$, $e^{\sigma,2} = (0,1,1,1,1)$, $e^{\sigma,3} = (0,0,1,1,1)$, $e^{\sigma,4} = (0,0,0,1,1)$ and $e^{\sigma,5} = (0,0,0,0,1)$.

4   A Decomposition Theorem

It is easy to check that for a simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ and a component T in $\langle V, E \rangle$, the total cost of a tree spanning all nodes in T at the minimum cost is equal to the the number of elements in $\mathcal{C}_T(\mathcal{L}')$ minus one, which is precisely the minimum number of edges of cost 1 that are needed to connect all $\mathcal{L}'$-components. So, for a simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ it holds that the corresponding LC game (E, c') can be rewritten as

$$c'(S) = \sum_{T \text{ is a component in } \langle V(S), S \rangle} (|\mathcal{C}_T(\mathcal{L}'_{|S})| - 1) \tag{4}$$

for each $S \in 2^E \setminus \{\emptyset\}$. In other terms, the cost of a coalition S is given by the sum, over all the components T in the sub-graph $\langle V(S), S \rangle$, of the minimum number of links of cost 1 needed to connect all the $\mathcal{L}'_{|S}$-components contained in T.

Example 4 (follows Example 2). Consider the simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ with $\langle V, E \rangle$ of Example 2 and w' such that $w'(2,3) = w'(2,4) = w'(3,4) = 1$ and $w'(1,2) = w'(1,3) = 0$, as depicted in Fig. 3. We have $\mathcal{C}(\mathcal{L}') = \{\{1,2,3\}, \{4\}\}$ and, by relation (4), $c(E) = |\mathcal{C}(\mathcal{L}')| - 1 = 1$.

Now, let $S = \{13, 24\}$. The LC situation $\mathcal{L}'_{|S} = \langle V(S), S, w'_{|S} \rangle$ is such that there are two components in $\langle V(S), S \rangle$, precisely, $\{1,3\}$ and $\{2,4\}$. Component $\{1,3\}$ contains only one $\mathcal{L}'_{|S}$-component, *i.e.*, $\mathcal{C}_{\{1,3\}}(\mathcal{L}'_{|S}) = \{\{1,3\}\}$, whereas component $\{2,4\}$ contains two $\mathcal{L}'_{|S}$-components, *i.e.*, $\mathcal{C}_{\{2,4\}}(\mathcal{L}'_{|S}) = \{\{2\}, \{4\}\}$. So, according to relation (4), $c'(S) = |\mathcal{C}_{\{1,3\}}(\mathcal{L}'_{|S})| - 1 + |\mathcal{C}_{\{2,4\}}(\mathcal{L}'_{|S})| - 1 = 1$. Differently, if $S = \{13, 12, 23, 24\}$, then $\langle V(S), S \rangle$ is connected, and, again, we have $\mathcal{C}_{\{1,2,3,4\}}(\mathcal{L}'_{|S}) = \{\{1,2,3\}, \{4\}\}$ and $c(S) = 1$.
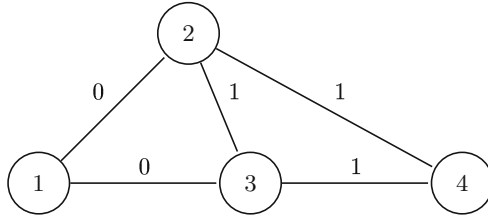
Fig. 3. A simple LC situation.

Following the approach introduced in [13] to decompose mcst games, we can now prove the following lemma.

Lemma 2. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation with at least one edge $e \in E$ such that $w(e) > 0$, and let $\alpha = \min\{w(e) : w(e) > 0\}$ be its minimum weight. Let $\mathcal{L}' = \langle V, E, w' \rangle$ be the simple LC situation defined by $w'(e) = 1$ if $w(e) > 0$ and $w'(e) = 0$, otherwise, for each $e \in E$, and let $\mathcal{L}'' = \langle V, E, w'' \rangle$ be the LC situation with $w'' = w - \alpha w'$. Finally, let c, c' and c'' be the LC games corresponding to $\mathcal{L}$, $\mathcal{L}'$ and $\mathcal{L}''$ respectively. Then, $c = \alpha c' + c''$.*

Proof. Clearly, by definition we have $w = \alpha w' + w''$. Let $S \in 2^E \setminus \{\emptyset\}$ and let $\langle V(S), \Gamma' \rangle$ be a mcsf for $\langle V(S), S, w'_{|S} \rangle$. Write $\Gamma' = L^0 \cup L^1$ where $L^0 := \{l \in \Gamma' : w'(l) = 0\}$ and $L^1 := \{l \in \Gamma' : w'(l) = 1\}$. The cost of Γ' is:

$$c'(S) = w'(\Gamma') = |L^1| = \sum_{T \text{ is a component in } \langle V(S), S \rangle} (|\mathcal{C}_T(\mathcal{L}'_{|S})| - 1)$$
$$= \sum_{T \text{ is a component in } \langle V(\Gamma'), \Gamma' \rangle} (|\mathcal{C}_T(\mathcal{L}'_{|\Gamma'})| - 1), \tag{5}$$

where the first equality follows from relation (4) and the second one from the fact that Γ' is a spanning forest in $\langle V(S), S \rangle$, which means that $\mathcal{P}_{\Gamma'} \equiv \mathcal{P}_S$.

We first show that there exists a mcsf Γ'' for $\langle V(S), S, w'' \rangle$ with $L^0 \subseteq \Gamma''$. Take an arbitrary mcsf Γ for S in $\langle V(S), S, w'' \rangle$. If $L^0 \not\subseteq \Gamma$ choose an $l \in L^0 \setminus \Gamma$. Since $\Gamma \cup \{l\}$ contains a cycle R, whereas Γ', and hence L^0, do not contain cycles, we can find an edge $l' \in R$ with $l' \notin L^0$. Define $\tilde{\Gamma} := (\Gamma \cup \{l\}) \setminus \{l'\}$. Since $w''(l) = 0$ and $w''(l') \geq 0$ we find that also $\tilde{\Gamma}$ is a mcsf for $\langle V(S), S, w'' \rangle$. Moreover $|\tilde{\Gamma} \cap L^0| = |\Gamma \cap L^0| + 1$. Repeating this argument results in the tree Γ'' with $L^0 \subseteq \Gamma''$.

Note that the set $\mathcal{P}_{\Gamma'}$ of all components in $\langle V(\Gamma'), \Gamma' \rangle$ coincides with the one $\mathcal{P}_{\Gamma''}$ in $\langle V(\Gamma''), \Gamma'' \rangle$. Moreover, since $L^0 \subseteq \Gamma''$, then for each component $T \in \mathcal{P}_{\Gamma'}$ (or, equivalently, in $\mathcal{P}_{\Gamma''}$), the number of $\mathcal{L}'_{|\Gamma''}$-components contained in T must be at most the corresponding number of $\mathcal{L}'_{|\Gamma'}$-components. Consequently, we have that

$$w'(\Gamma'') = \sum_{T \text{ is a component in } \langle V(\Gamma''), \Gamma'' \rangle} (|\mathcal{C}_T(\mathcal{L}'_{|\Gamma''})| - 1)$$
$$\leq \sum_{T \text{ is a component in } \langle V(\Gamma'), \Gamma' \rangle} (|\mathcal{C}_T(\mathcal{L}'_{|\Gamma'})| - 1) = w'(\Gamma'). \tag{6}$$

Therefore, Γ'' is also a mcsf for $\langle V(S), S, w' \rangle$. Having $w = \alpha w' + w''$ and the fact that Γ'' is a mcsf for S in both $\langle V(S), S, w' \rangle$ and $\langle V(S), S, w'' \rangle$ we

may conclude that Γ'' is also a mcsf for S in $\langle V(S), S, w \rangle$. So, $c(S) = w(\Gamma'') = \alpha w'(\Gamma'') + w''(\Gamma'') = \alpha c'(S) + c''(S)$.   □

The following decomposition theorem shows that every link game can be written as a non-negative combination of LC games corresponding to simple LC situations.

Theorem 1. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation and let (E, c) be its corresponding LC game. Let $\sigma \in \Sigma_E$ be such that $w(\sigma(1)) \leq w(\sigma(2)) \leq \ldots \leq w(\sigma(|E|))$. Define the LC game $(E, c^{\sigma,k})$ corresponding to the simple LC situation $\mathcal{L}^{\sigma,k} = \langle V, E, e^{\sigma,k} \rangle$, for each $k \in \{1, \ldots, |E|\}$. Then,*

$$c = w(\sigma(1))c^{\sigma,1} + \sum_{k=2}^{|E|} \left(w(\sigma(k)) - w(\sigma(k-1)) \right) c^{\sigma,k}. \tag{7}$$

Proof. The proof follows directly by Lemma 1 and the recursive application of Lemma 2, using $e^{\sigma,j}$ in the role of w', $w(\sigma(j)) - w(\sigma(j-1))$ in the role of α, and $\sum_{k=j+1}^{|E|} \left(w(\sigma(k)) - w(\sigma(k-1)) \right) e^{\sigma,k}$ in the role of w'' at each recursive call $j \in \{1, \ldots, |E| - 1\}$ (and setting, by convention, $w(\sigma(0)) = 0$).   □

Example 5 (follows Examples 2 and 3). Consider again the LC situation of Examples 2 and 3, with $\sigma = (\{1, 2\}, \{1, 3\}, \{2, 4\}, \{3, 4\}, \{2, 3\})$. According to Theorem 1 we have $c = c^{\sigma,1} + 0c^{\sigma,2} + c^{\sigma,3} + 2c^{\sigma,4} + 4c^{\sigma,5}$, where the LC games $c^{\sigma,1}, \ldots, c^{\sigma,5}$ corresponding to the simple LC situations $e^{\sigma,1}, \ldots, e^{\sigma,5}$ are those shown in Table 2. One can easily verify that the last row of Table 2 coincides with the LC game c, as computed in Example 2 (see Table 1).

**Table 2.** Decomposition of the LC game corresponding to the LC situation of Fig. 2.

S	$\{12, 13, 23\}$	$\{23, 24, 34\}$	$\{12, 13, 23, 34\}$	$\{12, 13, 23, 24\}$	$\{23, 24, 34, 13\}$	$\{23, 24, 34, 12\}$	E
$c^{\sigma,1}(S)$	2	2	3	3	3	3	3
$c^{\sigma,2}(S)$	2	2	3	3	3	3	3
$c^{\sigma,3}(S)$	0	2	1	1	2	2	1
$c^{\sigma,4}(S)$	0	1	1	0	1	1	0
$c^{\sigma,5}(S)$	0	0	0	0	0	0	0
Sum	2	6	6	4	7	7	4

5   The Core of an LC Game

In this section we prove that LC games have a non-empty core and that core allocations can be efficiently computed, even if, as we have shown in the previous section, LC games are not necessarily concave. One could argue that the savings due to cooperation in an LC situation originate from the possibility to break cycles without destroying the connectivity of the network. On the other hand, it is not immediately clear how those savings should be shared among the

links involved in the cycle in order to obtain a core allocation. Next example shows that trivial allocation protocols, to be more specific, the equal sharing rule applied to the edges involved in the cycles, in general does not provide a core allocation.

Example 6. Consider the simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ depicted in Fig. 4, with the set E composed by the 15 edges depicted in Fig. 4 and where the cost $w'(e)$ of each edge $e \in E$ is equal to 1. In order to obtain a mcsf on $\mathcal{L}'$ it suffices to eliminate four edges such that no cycles appear and the network remains connected (*e.g.*, deleting edges $\{1, 2\}, \{4, 5\}, \{7, 8\}$ and $\{10, 11\}$), therefore leading to an optimal network of cost 11. On the other hand, if we split equally the total cost 11 (or the total saving 4) among the edges of the network we obtain that each link in E should pay $\frac{11}{15}$ which is not in the core of the corresponding LC game, since $c(12, 13, 23) = 2 < 3\frac{11}{15} = x_{12} + x_{13} + x_{23}$.
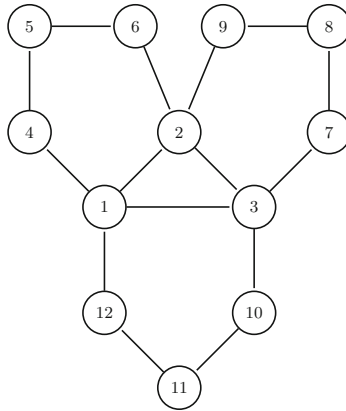
Fig. 4. A simple LC situation where the equal sharing allocation does not belong to the core (the cost of each edge is 1).

Even if the cycles play a central role in the determination of the savings (as illustrated in the previous example), defining an allocation rule based on the analysis of the cycles of a graph could be computationally very hard (one edge may belong to several cycles). In the following, our objective is to prove that LC games have a non-empty core and core allocations can be easily computed without looking at the cycles of a graph.

Let $\mathcal{L}' = \langle V, E, w' \rangle$ be a simple LC situation (with $w'(e) \in \{0, 1\}$). For each $i \in V$, let $C_i(\mathcal{L}')$ be the $\mathcal{L}'$-component to which i belongs. We denote by $B_{ij}^{\Gamma} = \{\{k, l\} \in E : k \in C_i(\mathcal{L}') \text{ and } l \in C_j(\mathcal{L}')\}$ the *bridge* set of all edges connecting the two $\mathcal{L}'$-components $C_i(\mathcal{L}')$ and $C_j(\mathcal{L}')$ (clearly, $\{i, j\} \in B_{ij}^{\Gamma}$ and $w(k, l) = 1$ for each $\{k, l\} \in B_{ij}^{\Gamma}$). Moreover, let $\bar{E}^{\Gamma} = E \setminus \bigcup_{e \in \Gamma : w'(e) = 1} B_{ij}^{\Gamma}$ be the set of edges in E that do not belong to any set B_{ij}^{Γ} with $\{i, j\} \in \Gamma$ and $w'(i, j) = 1$.

Remark 2. Let $\langle V, \Gamma \rangle$ be a mcsf for the simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$. Note that for $\{i, j\}, \{k, l\} \in \Gamma$ with $\{i, j\} \neq \{k, l\}$ and $w'(i, j) = w'(k, l) = 1$, we have $B_{ij}^{\Gamma} \cap B_{kl}^{\Gamma} = \emptyset$, since each edge of cost 1 in Γ connects two disjoint $\mathcal{L}'$-components.

Example 7 (follows Example 4). Consider again the simple LC situation $\mathcal{L}' = \langle V, E, w' \rangle$ of Example 4. Let the tree $\Gamma = \{\{1, 2\}, \{1, 3\}, \{2, 4\}\}$ be a mcst in $\mathcal{L}'$. Then, we have that $B_{24}^{\Gamma} = \{\{2, 4\}, \{3, 4\}\}$ and $\bar{E}^{\Gamma} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}\}$.

Now, we can introduce a family of cost sharing vectors for simple LC games.

Definition 2. *Let $\mathcal{L}' = \langle V, E, w' \rangle$ be a simple LC situation and let $\langle V, \Gamma \rangle$ be a mcsf for $\mathcal{L}' = \langle V, E, w' \rangle$. We denote by $\mathcal{X}(\mathcal{L}', \Gamma)$ the set of (positive) vectors $x \in \mathbb{R}_+^E$ satisfying the following two conditions:*

(i) $\sum_{e \in B_{ij}^{\Gamma}} x_e = 1$ *for all* $\{i, j\} \in \Gamma$ *such that* $w'(i, j) = 1$;
(ii) $x_e = 0$ *for all* $e \in \bar{E}$,

where $\langle V, \Gamma \rangle$ is a mcsf for the simple LC situation $\mathcal{L}'$.

In other words, $\mathcal{X}(\mathcal{L}', \Gamma)$ is the set of positive allocation vectors such that the service providers located over the edges in $\bar{E}^{\Gamma}$ pay nothing and those over the edges in B_{ij}^{Γ}, for each $\{i, j\} \in \Gamma$ with $w'(i, j) = 1$, share the cost to connect $C_i(\mathcal{L}')$ and $C_j(\mathcal{L}')$.

Lemma 3. *Let $\mathcal{L}' = \langle V, E, w' \rangle$ be a simple LC situation (with $w'(e) \in \{0, 1\}$) and let $\langle V, \Gamma \rangle$ be a mcsf for $\mathcal{L}'$. Then $\mathcal{X}(\mathcal{L}', \Gamma) \neq \emptyset$ and $\sum_{e \in E} x_e = w'(\Gamma)$.*

Proof. To prove that $\mathcal{X}(\mathcal{L}', \Gamma) \neq \emptyset$, simply take the vector $x \in \mathbb{R}_+^E$ such that

$$x_e = \begin{cases} \frac{1}{|B_{ij}^{\Gamma}|} & \text{if } \exists \{i, j\} \in \Gamma \text{ with } w'(i, j) = 1 \text{ and } e \in B_{ij}^{\Gamma}, \\ 0 & \text{otherwise,} \end{cases} \tag{8}$$

for each $e \in E$. It is immediate to check that the vector x defined according to relation (8) satisfies conditions (i) and (ii) in Definition 2. To see that every vector $x \in \mathcal{X}(\mathcal{L}', \Gamma)$ is efficient, simply notice that

$$\sum_{e \in E} x_e = \sum_{\{i,j\} \in \Gamma : w'(i,j)=1} \sum_{e \in B_{ij}^{\Gamma}} x_e = \sum_{\{i,j\} \in \Gamma : w'(i,j)=1} 1 = w'(\Gamma),$$

where the first equality follows from condition (ii) in Definition 2, and the second equality from condition (i). $\square$

Example 8 (follows Examples 4 and 7). The allocation vectors in $\mathcal{X}(\mathcal{L}', \Gamma)$, whatever mcsf Γ for $\mathcal{L}'$ is constructed, is such that the edges $\{2, 4\}$ and $\{3, 4\}$ share the cost of connecting the $\mathcal{L}'$-components $\{1, 2, 3\}$ and $\{4\}$. We have that

$$\mathcal{X}(\mathcal{L}', \Gamma) = \{x \in \mathbb{R}_+^E : x_{12} = x_{13} = x_{23} = 0 \text{ and } x_{24} + x_{34} = 1\}.$$

Next lemma, that holds for general LC situations, is useful to prove the non-emptiness of the core of simple LC games, as shown by Proposition 2.

Lemma 4. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation, let $\langle V, \Gamma \rangle$ be a mcsf for $\mathcal{L}$ and let c be the corresponding LC game. For each $S \subseteq E$. Then,*

$$c(S) \geq w(\Gamma \cap S), \tag{9}$$

Proof. Recall that $c(S) = w(\Gamma_S) = \sum_{e \in \Gamma_S} w(e)$, where Γ_S is a mcsf for the restriction $\mathcal{L}_S = \langle V(S), S, w_{|S} \rangle$, and $w(\Gamma \cap S) = \sum_{e \in \Gamma \cap S} w(e)$.

First note that each component T in $\langle V(\Gamma \cap S), \Gamma \cap S \rangle$ is also a connected set of nodes in the network $\langle V(S), \Gamma_S \rangle$. So, if two nodes $i, j \in V$ are connected in the network $\langle V, \Gamma \rangle$ they must be connected also in the network $\langle V, (\Gamma \setminus S) \cup \Gamma_S \rangle$ (i.e., $\langle V, (\Gamma \setminus S) \cup \Gamma_S \rangle$ is a spanning network for $\langle V, E \rangle$, meaning that $\mathcal{P}_{(\Gamma \setminus S) \cup \Gamma_S} \equiv \mathcal{P}_E$), and this directly implies relation (9). Suppose, on the contrary, that $w(\Gamma \cap S) > c(S) = w(\Gamma_S)$. By simple considerations on the sets of edges Γ, S and Γ_S we obtain that

$$w(\Gamma) = w((\Gamma \cap S) \cup (\Gamma \setminus S)) = w(\Gamma \cap S) + w(\Gamma \setminus S) > w(\Gamma_S) + w(\Gamma \setminus S) \geq w(\Gamma_S \cup (\Gamma \setminus S)),$$

which yields a contradiction with the fact that Γ is a mcsf for $\langle V, E \rangle$ (notice that the second equality follows from the fact that $(\Gamma \cap S) \cap (\Gamma \setminus S) = \emptyset$ and the last inequality from the fact that $\Gamma_S \cap (\Gamma \setminus S)$ is not necessarily empty). □

Proposition 2. *Let $\mathcal{L}' = \langle V, E, w' \rangle$ be a simple LC situation (with $w'(e) \in \{0, 1\}$) and let (E, c') be the corresponding LC game. Then, $\mathcal{X}(\mathcal{L}', \Gamma) \subseteq Core(c') \neq \emptyset$.*

Proof. By Lemma 3 we know that $\mathcal{X}(\mathcal{L}', \Gamma) \neq \emptyset$ and that each element $x \in \mathcal{X}(\mathcal{L}', \Gamma)$ is an efficient allocation. Now, in order to prove that $x \in \mathcal{X}(\mathcal{L}', \Gamma)$ is in $Core(c')$ we need to prove that $\sum_{e \in S} x_e \leq c'(S)$ for all $S \subseteq E$. Notice that

$$\sum_{e \in S} x_e = \sum_{\{i,j\} \in S \cap \Gamma: w'(i,j)=1} \sum_{e \in S \cap B_{ij}^{\Gamma}} x_e \leq \sum_{e \in S \cap \Gamma: w'(i,j)=1} 1 = \sum_{e \in S \cap \Gamma} w'(e) \leq c'(S), \tag{10}$$

for each $S \subseteq E$, where the first equality follows from condition (ii) in Definition 2, the first inequality from condition (i) in Definition 2, the second equality from the fact that w' is a simple LC situation and the second inequality from Lemma 4. □

We can finally prove that LC games have a non-empty core.

Theorem 2. *Let $\mathcal{L} = \langle V, E, w \rangle$ be an LC situation and let (E, c) be the corresponding LC game. Then, $Core(c) \neq \emptyset$.*

Proof. Let $\sigma \in \Sigma_E$ be such that $w(\sigma(1)) \leq w(\sigma(2)) \leq \ldots \leq w(\sigma(|E|))$. For each $k \in \{1, \ldots, |E|\}$, take $x^k \in \mathcal{X}(\mathcal{L}^{\sigma,k}, \Gamma^k)$, where $\langle V, \Gamma^k \rangle$ is a mcsf for $\mathcal{L}^{\sigma,k}$ and

$c^{\sigma,k}$ is the LC game corresponding to the simple LC situation $e^{\sigma,k}$. Define the vector $x \in \mathbb{R}^E$ such that

$$x = w(\sigma(1))x^k + \sum_{k=2}^{|E|} \big(w(\sigma(k)) - w(\sigma(k-1))\big)x^k.$$

For each $S \subseteq E$ with $S \neq \emptyset$ we have that

$$\sum_{e \in S} x_e \leq w(\sigma(1))c^{\sigma,1}(S) + \sum_{k=2}^{|E|} \big(w(\sigma(k)) - w(\sigma(k-1))\big)\ c^{\sigma,k}(S) = c(S), \quad (11)$$

where the inequality follows from the fact that, by Proposition 2, $x^k \in Core(c^{\sigma,k})$ and the fact that $w(\sigma(1)) \geq 0$ and $w(\sigma(k)) - w(\sigma(k-1)) \geq 0$ for each $k \in \{1, \ldots, |E|\}$, and the second equality follows directly from Theorem 1; similarly, for the efficiency condition of core allocations in $\mathcal{X}(\mathcal{L}^{\sigma,k}, \Gamma^k)$ we have

$$\sum_{e \in N} x_e = w(\sigma(1))c^{\sigma,1}(E) + \sum_{k=2}^{|E|} \big(w(\sigma(k)) - w(\sigma(k-1))\big)\ c^{\sigma,k}(E) = c(E) = w(\Gamma).$$

Then it has been established that $x \in Core(c)$. $\square$

Example 9 (follows Examples 2, 3 and 5). Let $\Gamma^k = \Gamma = \{\{1,2\}, \{1,3\}, \{2,4\}\}$ for each $k \in \{1, \ldots, 5\}$ (notice that this is a mcsf obtained using the Kruskal algorithm [9] on the ordering of the edges σ). It is easy to check that $\mathcal{X}(\mathcal{L}^{\sigma,1}, \Gamma) = \{(x_{12}, x_{13}, x_{24}, x_{34}, x_{23}) = (1,1,1,0,0)\}$, $\mathcal{X}(\mathcal{L}^{\sigma,2}, \Gamma) = \{x \in \mathbb{R}_+^E : x_{13} + x_{23} = x_{24} = 1 \text{ and } x_{12} = x_{34} = 0\}$, $\mathcal{X}(\mathcal{L}^{\sigma,3}, \Gamma) = \{x \in \mathbb{R}_+^E : x_{12} = x_{13} = x_{23} = 0 \text{ and } x_{24} + x_{34} = 1\}$ and $\mathcal{X}(\mathcal{L}^{\sigma,4}, \Gamma) = \mathcal{X}(\mathcal{L}^{\sigma,5}, \Gamma) = \{(x_{12}, x_{13}, x_{24}, x_{34}, x_{23}) = (0,0,0,0,0)\}$. Consider for instance the core allocations $x^k \in \mathcal{X}(\mathcal{L}^{\sigma,k}, \Gamma)$ computed according to relation (8) as follows:

	$\{1,2\}$	$\{1,3\}$	$\{2,4\}$	$\{3,4\}$	$\{2,3\}$
x^1	1	1	1	0	0
x^2	0	$\frac{1}{2}$	1	0	$\frac{1}{2}$
x^3	0	0	$\frac{1}{2}$	$\frac{1}{2}$	0
x^4	0	0	0	0	0
x^5	0	0	0	0	0
$x = x^1 + 0x^2 + x^3 + 2x^4 + 4x^5$	1	1	$\frac{3}{2}$	$\frac{1}{2}$	0

One can easily verify that $x \in Core(c)$, as immediately suggested by Theorem 2.

6   Concluding Remarks

In this paper we studied a class of cooperative games where the players are the edges of a weighted graph and the goal of a coalition of edges is to connect the adjacent nodes at a minimum cost. We also provided a procedure based on a decomposition theorem to easily generate allocation vectors in the core of an LC game. An interesting research direction is related to the property-driven analysis of particular one-point solutions for LC situations, *i.e.* maps that associate to

each LC situation a particular allocation vector, independently from the selected mcsf and, possibly, in the core of the corresponding LC game. Alternative cost allocation protocols keeping into account the role of edges in maintaining the connectivity of the network should be further investigated.

As shortly suggested in Example 9, the procedure to find core allocations used in the proof of Theorem 2 is strongly related to the Kruskal algorithm for finding a spanning network of minimum cost on weighted graphs [9]. In general, it is possible to define a procedure aimed at computing vectors in $\mathcal{X}(\mathcal{L}^{\sigma,k}, \Gamma^k)$ at each k-th step of the Kruskal algorithm, and obtain, after precisely n-steps (where n is the number of nodes, if the graph is connected) both an optimal network and an allocation in the core of the LC game. On the other hand, the procedure used in Theorem 2 selects the elements of a particular subset of the core, and the issue of how to efficiently generate all the allocations in the core of an LC game (or other specific subsets of stable allocations) is still an open problem. Notice that the non-emptiness of the core for LC games can be also proved using the results in [12] for games on matroids (LC games being a special case of games on matroids), and an interesting related question is whether the procedure used in Theorem 2 can be generalized to the more general framework of matroids.

Another open question concerns the existence of solutions for LC games that are *cost monotonic* (*i.e.*, such that if some connection costs go down, then no edges will pay more) and, in addition, that can be extended to a *population monotonic allocation scheme* (pmas) [17] (roughly speaking, an allocation method is pmas extendible if it assigns an allocation vector to every coalition in a monotonic way and such that the cost allocated to some edge does not increase if the coalition of edges to which it belongs becomes larger). It would be interesting to analyse whether the core allocations computed according to the procedure used in Theorem 2 satisfy these properties.

Finally, as an alternative framework, one could imagine a version of a (monotonic) LC game where each service provider has the power, alone or in cooperation, to control the implementation of the services over the entire network, and not only those using the connections within a given coalition, like in the current version of the model.

References

1. Aziz, H., Lachish, O., Paterson, M., Savani, R.: Wiretapping a hidden network. In: Leonardi, S. (ed.) WINE 2009. LNCS, vol. 5929, pp. 438–446. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-10841-9_40
2. Bergañtinos, G., Lorenzo, L., Lorenzo-Freire, S.: A generalization of obligation rules for minimum cost spanning tree problems. Eur. J. Oper. Res. **211**, 122–129 (2011)
3. Bird, C.: On cost allocation for a spanning tree: a game theoretic approach. Networks **6**, 335–350 (1976)
4. Branzei, R., Moretti, S., Norde, H., Tijs, S.: The P-value for cost sharing in minimum cost spanning tree situations. Theory Decis. **56**, 47–61 (2004)

5. Claus, A., Kleitman, D.J.: Cost allocation for a spanning tree. Networks **3**, 289–304 (1973)
6. Feltkamp, V.: Cooperation in controlled network structures, PhD Dissertation, Tilburg University, The Netherlands (1995)
7. Granot, D., Huberman, G.: On minimum cost spanning tree games. Math. Prog. **21**, 1–18 (1981)
8. Hougaard, J.L., Moulin, H.: Sharing the cost of redundant items. Games Econ. Behav. **87**, 339–352 (2014)
9. Kruskal, J.B.: On the shortest spanning subtree of a graph and the traveling salesman problem. Proc. Amer. Math. Soc. **7**, 48–50 (1956)
10. Moretti, S., Tijs, S., Branzei, R., Norde, H.: Cost allocation protocols for supply contract design in network situations. Math. Meth. Oper. Res. **69**(1), 181–202 (2009)
11. Moulin, H., Laigret, F.: Equal-need sharing of a network under connectivity constraints. Games Econ. Behav. **72**(1), 314–320 (2011)
12. Nagamochi, H., Zeng, D.Z., Kabutoya, N., Ibaraki, T.: Complexity of the minimum base game on matroids. Math. Oper. Res. **22**(1), 146–164 (1997)
13. Norde, H., Moretti, S., Tijs, S.: Minimum cost spanning tree games and population monotonic allocation schemes. Eur. J. Oper. Res. **154**(1), 84–97 (2004)
14. Prim, R.C.: Shortest connection networks and some generalizations. Bell Syst. Tech. J. **36**, 1389–1401 (1957)
15. Shapley, L.S.: A Value for n-Person Games. In: Kuhn, H.W., Tucker, A.W. (ed.) Contributions to the Theory of Games II, Ann. Math. Studies 28, 307–317, Princeton University Press (1953)
16. Shapley, L.S.: Cores and convex games. Int. J. Game Theory **1**, 1–26 (1971)
17. Sprumont, Y.: Population monotonic allocation schemes for cooperative games with transferable utility. Games Econ. Behav. **2**, 378–394 (1990)
18. Tijs, S., Branzei, R., Moretti, S., Norde, H.: Obligation rules for minimum cost spanning tree situations and their monotonicity properties. Eur. J. Oper. Res. **175**, 121–134 (2006)