



Cost-allocation problems for fuzzy agents in a fixed-tree network

Julio R. Fernández¹ · Inés Gallego² · Andrés Jiménez-Losada¹  · Manuel Ordóñez¹

Accepted: 5 October 2021
© The Author(s) 2022

Abstract

Cost-allocation problems in a fixed network are concerned with distributing the costs for use by a group of clients who cooperate in order to reduce such costs. We work only with tree networks and we assume that a minimum cost spanning tree network has already been constructed and now we are interested in the maintenance costs. The classic problem supposes that each agent stays for the entire time in the same node of the network. This paper introduces cost-allocation problems in a fixed-tree network with a set of agents whose activity over the nodes is fuzzy. Agent's needs to pay for each period of time may differ. Moreover, the agents do not always remain in the same node for each period. We propose the extension of a very well-known solution for these problems: Bird's rule.

Keywords Cost allocation · Bird's rule · Fuzzy sets · Networks · Choquet integral · Cooperative games

1 Introduction

The cost-allocation problem in a network considers a group of agents where each agent needs to be connected to a source either directly or via other agents. A feasible network connects all of these agents to the source and the costs for using all the links in the network are known. Cooperation of the agents is assumed in order to reduce the cost. In graph theory, classic algorithms that search for minimum spanning trees enable all the cheapest trees to be found for the agents. Claus and Kleitman (1973) introduce a

✉ Andrés Jiménez-Losada
ajlosada@us.es

¹ Dpto. Matemática Aplicada II, Escuela Técnica Superior de Ingeniería, Universidad de Sevilla, Sevilla, Spain

² Dpto. Didáctica de las Matemáticas, Facultad de Ciencias de la Educación, Universidad de Sevilla, Sevilla, Spain

second problem once a minimum spanning tree is chosen: how to allocate the minimum cost among the agents. This problem is known as the minimum cost spanning tree problem. Bird (1976) defined a rule to allocate the cost for this problem. This rule has been analyzed in several studies in the literature, including axiomatizations and generalizations (Feltkamp et al. 2000; Dutta and Kar 2004 ; Hougaard et al. 2013). Numerous rules have been defined for the minimum cost spanning tree problem: Granot and Huberman (1984), Feltkamp et al. (1994), Kar (2002), Dutta and Kar 2004, Bergantiños and Vidal-Puga (2007), Bogomolnaia et al. (2010), Juarez and Kumar (2013) or Norde (2019). A related problem introduced in Meggido (1978) describes the allocation of maintenance and usage costs of a fixed-tree network. This problem is called fixed tree cost-allocation problem and it has also been studied in Granot et al. (1996), and Koster et al. (2002) in the context of game theory. Bird's rule (Bird 1976; Ertan et al. 2020) is based on a particular solution for each fixed minimum cost spanning tree of the network and then, it can also be used also for this problem. Several interesting families of rules that contain Bird's rule have been defined. Chun and Lee (2012) introduced sequential contributions rules as a family of rules following a certain sequential way to determine the payoffs of the agents. In this context, Bird's rule is named the sequential full contributions rule. Gellekom (2000) and Bergantiños and Gómez-Rúa (2010) introduced populations on the nodes in both problems, by assuming that there is a set of agents in each vertex (this population can also be empty).

This paper deals with cost-allocation problems in a fixed-tree. We propose a new generalization of the problem in the following sense. In the classic model, it is supposed that agents stay in the same node for the whole period of use and that these agents are using the network full time. However, an agent may not use the network all the time or to use it a third of the time in one node and the rest in another node. For instance, the clients in an intranet using mobile cloud computing can use applications in different points of the network. The population set in each vertex is now a fuzzy set. There exist other kind of solutions to this problem, such as in Dai et al. (2016).

In Sect. 2 several necessary preliminaries regarding fuzzy sets and the Choquet integral are given. Section 3 is dedicated to introducing cost-allocation problems in a fixed-tree with populations and a Bird's rule for these situations is defined. The main problem is presented in Sect. 4. The position of the agents on the nodes is now variable. This kind of situation is known as a cost-allocation problem in a fixed-tree with fuzzy agents. Our objective in the next sections is to define a Bird's rule as an extension of the classic rule (Sect. 5) and get an axiomatization of this rule (Sect. 6). Finally we have a section dedicated to discussion and conclusions.

2 Preliminaries

Let N be a finite set. A *fuzzy set* (Zadeh 1965) s in N is determined by a membership function $\tau_s : N \rightarrow [0, 1]$ where $\tau_s(i)$ is named the membership level of $i \in N$ in s . The family of fuzzy sets in N is denoted as $[0, 1]^N$. Each (crisp) subset $S \subseteq N$ is also a fuzzy set with membership function e^S , $e^S(i) = 1$ if $i \in S$, and is $e^S(i) = 0$ otherwise. Specifically, we denote $e^\emptyset = 0$. The *support* of $s \in [0, 1]^N$ is $\text{supp}(s) =$

$\{i \in N : \tau_s(i) \neq 0\}$ and the *image* of τ is the set

$$\text{im}(s) = \{\lambda \in (0, 1] : \exists i \in N \text{ with } \tau_s(i) = \lambda\}.$$

In this paper, the operators \wedge, \vee are used as the minimum and the maximum, respectively. For all $s_1, s_2 \in [0, 1]^N$, the standard operations between fuzzy sets become:

$$\begin{aligned} s_1 \cap s_2 \in [0, 1]^N & \text{ with } \tau_{s_1 \cap s_2} = \tau_{s_1} \wedge \tau_{s_2}, \\ s_1 \cup s_2 \in [0, 1]^N & \text{ with } \tau_{s_1 \cup s_2}(i) = \tau_{s_1} \vee \tau_{s_2}, \\ s_1 + s_2 \in [0, 1]^N & \text{ with } \tau_{s_1 + s_2} = \tau_{s_1} + \tau_{s_2} \\ & \text{(since this sum is less than or equal 1),} \\ \lambda s_1 \in [0, 1]^N & \text{ with } \tau_{\lambda s_1} = \lambda \tau_{s_1} \text{ (being } \lambda \in [0, 1]). \end{aligned}$$

If $s \in [0, 1]^N$, then $\vee s = \vee\{\tau_s(i) : i \in N\}$. If $\lambda \in (0, 1]$ and $s \in [0, 1]^N$, then the λ -cut is the crisp set

$$[s]_\lambda = \{i \in N : \tau_s(i) \geq \lambda\}.$$

Choquet (1953) introduced *capacity* as a monotonic set function $v : 2^N \rightarrow \mathbb{R}$ which verifies $v(\emptyset) = 0$. The Choquet integral with regard to a capacity is used as a way of measuring the expected utility of a fuzzy event. This integral was extended in de Waegenaere and Wakker (2001) to include capacities without monotonicity (non-monotonic capacities). Let v be a non-monotonic capacity, the (signed) *Choquet integral* of $s \in [0, 1]^N$ with regard to v is defined as

$$\int s \, dv = \sum_{k=1}^m (\lambda_k - \lambda_{k-1}) v([s]_k),$$

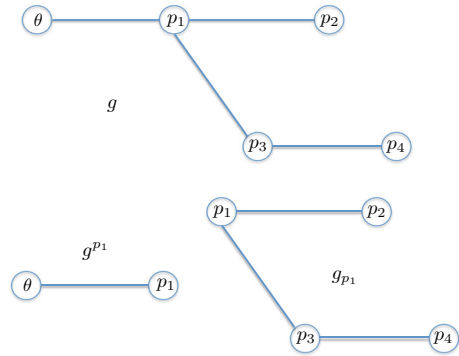
where $\text{im}(s) = \{\lambda_1 < \dots < \lambda_m\}$, $\lambda_0 = 0$, and $[s]_k = [s]_{\lambda_k}$ for each $k = 1, \dots, m$. The properties of the Choquet integral that will be used include the following.

- (1) $\int S \, dv = v(S)$, for all $S \subseteq N$.
- (2) $\int \lambda s \, dv = \lambda \int s \, dv$, for all $\lambda \in [0, 1]$.
- (3) $\int s \, d(v + v') = \int s \, dv + \int s \, dv'$.
- (4) $\int s \, dv = K(\vee s)$ if $v([s]_\lambda) = K$ for all $\lambda \in \text{im}(s)$.

3 Cost-allocation problems in a fixed-tree

A fixed-tree network that connects a set of nodes with a source is assumed. The cost-allocation problem for this network determines how to allocate the costs for the usage

Fig. 1 A tree and its partition by a node



or for maintenance by a group of clients. These problems commonly identify nodes and agents, by establishing one agent in each node. Bearing our objective in mind, however, these problems need to be applied in a wider sense. According to Gellekom (2000), each node contains a determined population of agents, and some of these populations can be empty.

We describe first the network. Given a finite set V , we denote $LV = \{\{p, q\} : p, q \in V, p \neq q\}$ as the set of unordered pairs of different elements in V . A graph is a pair (V, L) where V is a finite set of nodes and $L \subseteq LV$ is a set of links. A sequence of k different nodes (p_1, \dots, p_k) is a path in the graph if $\{p_l, p_{l+1}\} \in L$ for $l = 1, \dots, k - 1$. The graph is connected if for all $p, q \in V$ with $p \neq q$ there is a path (p_1, \dots, p_k) in which $p_1 = p$ and $p_k = q$. We say that (V, L) is cycle-free if, for every two different connected nodes $p, q \in N$, there is only one path connecting p and q . A rooted tree (hereinafter, tree) is a connected cycle-free graph with a particular node θ , known as the source. We represent a tree as the graph $g = (V \cup \{\theta\}, L)$. The selection of a source determines a partial ordering over V , where $p <_g q$ if p is in the unique path from θ to q .

With a fixed $p \in V$, we denote $V^p = \{q \in V : q \leq_g p\}$ and $V_p = \{q \in V : q \geq_g p\}$. The height of a node $p \in V$ in the tree is $h_g(p) = |V^p| - 1$. Each link $\{p, q\} \in L$ is written as pq where p is the lowest node (initial node), and q the highest node (the end node). Furthermore, link pq is usually identified with its end node as $pq = l_q$ since there is only one link with this condition. If $p \in V \setminus \{\theta\}$, then g^p is the chain up to p , namely the unique path from θ to p (also a tree), with set the intermediate nodes V^p and the corresponding links. Also g_p is the subtree from p , namely that uses all the nodes in V_p , their corresponding links except l_p , and now takes p as its source (see Fig. 1). Given a node $p \in V$, there is only one node $q \in V \cup \{\theta\}$ satisfying $qp \in L$, since g is acyclic.

A cost-allocation problem in a fixed-tree is $\gamma = (N, g, c, d)$, where N is a finite set of n agents using a network represented by a tree g , and where $c : V \rightarrow \mathbb{R}_+$ is a cost function yielding the cost for usage for a certain period of the corresponding link l_p for each node p (different to the source). If $|V| = r$ then $d = \{S_1, \dots, S_r\}$ is a partition of N that determines the position of the agents in the network. Partition d can be represented by a matrix $d = (d_{pi})_{r \times n}$ where $d_{pi} = 1$ if $i \in S_p$ and $d_{pi} = 0$ otherwise, and where $\sum_{p \in V} d_{pi} = 1$ for all $i \in N$. This kind of matrix is named

partition matrix of N over V . For each $p \in V$, set S_p is the group of those agents using p during the period. Set S_p can be empty. If we have only one agent placed in a vertex p , then this agent must pay the cost for using the corresponding links from the source to p , that is, the cost is $\sum_{q \in V^p} c(q)$. Given $\gamma = (N, g, c, d)$ a cost-allocation problem in a fixed-tree, we denote by $V(\gamma) = \{p \in V : S_p \neq \emptyset\}$ and therefore the cost to allocate is

$$c(\gamma) = \sum_{p \in \bigcup_{q \in V(\gamma)} V^q} c(p).$$

A solution for a cost-allocation problem γ in a fixed network is a vector containing the payment that each agent must make to cover the cost of the problem, namely the vector is an allocation of $c(\gamma)$ among the agents. There are numerous allocation rules giving a solution for each cost-allocation problem in a fixed tree. Bird (1976) introduced a rule in the context of the minimum cost spanning tree problems. We define Bird's rule for our case in an easy way. If $\gamma = (N, g, c, d)$ is a cost-allocation problem in a fixed tree then for each $p \in V$ we denote V_p^{min} as the minimal nodes in $V_p \cap V(\gamma)$ with regard to the ordering $<_\gamma$, and $N_p^{min} = \bigcup_{q \in V_p^{min}} N_q$. Observe that if $p \in V(\gamma)$ then $V_p^{min} = \{p\}$ and $N_p^{min} = N_p$, otherwise $V_p(\gamma)$ is the set of those nodes closest to p moving away from p , and $N_p(\gamma)$ is the set of their populations. The *sequential full contributions rule* (Bird's rule) defines for γ the payment

$$b_i(\gamma) = \sum_{\{p \in V : i \in N_p^{min}\}} \frac{c(p)}{|N_p^{min}|},$$

for all $i \in N$. This rule supposes that the cost of a link is paid by the agents in the first nodes from p moving away from the source (full contributions). Hence, although other agents may need this link in more remote nodes (because p lies on in the only path from θ) they do not have to pay for using l_p .

Example 1 Let $\gamma = (N, g, c, d)$ with $N = \{1, 2, 3, 4, 5\}$ and

$$g = (\{p_1, p_2, p_3, p_4\} \cup \{\theta\}, \{\theta p_1, p_1 p_2, p_1 p_3, p_3 p_4\}) \text{ (see Fig. 2).}$$

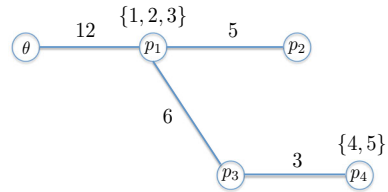
The distribution of the agents in the nodes is given by the partition matrix d ,

$$d = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix}.$$

The cost function is: $c(p_1) = 12, c(p_2) = 5, c(p_3) = 6, c(p_4) = 3$.

The cost of the problem is $c(\gamma) = 21$. The sequential full contributions rule for this problem is $b(\gamma) = (4, 4, 4, 4.5, 4.5)$.

Fig. 2 A cost-allocation problem in a fixed-tree



Bird (1976) introduced a cooperative game to represent cost allocation problems in networks. In our case, fixed $\gamma = (N, g, c, d)$, the γ -game is defined as (N, v^γ) where for each coalition $S \subseteq N$,

$$v^\gamma(S) = c(\gamma_S),$$

with $\gamma_S = (S, g, c, d^S)$ and $(d^S_{pi})_{|V| \times |S|}$ the submatrix of d taking only the columns in S .

4 Cost-allocation tree problems with fuzzy agents

Our proposal is now different in the sense that the set of agents construct a minimal tree network but they ignore how they can use it in each period. We are interested in allocating the maintenance or usage costs for each period between the agents. We work with flexible agents in the sense that in the same period they can stay at several different nodes and they may not be interested in using the network all the time. The full-time usage of the network in the period is therefore considered as 1. Each billing period provides information about the usage of each node by the agents. It is supposed that the order of use of the nodes is of no importance. The population of a vertex is hence a fuzzy set of N where the membership level of an agent is defined by his time in the vertex.

Definition 1 A cost-allocation tree problem with fuzzy agents is $\Gamma = (N, g, c, \tau)$ where:

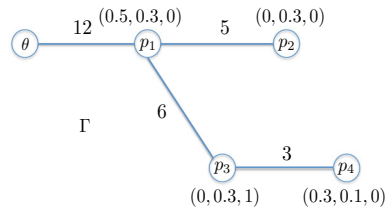
- 1) N is a finite set of agents with cardinality n ,
- 2) $g = (V \cup \{\theta\}, L)$ is a tree representing a network with $|V| = r$,
- 3) $c : V \rightarrow \mathbb{R}_+$ determines the cost of full-time use of the corresponding link of each node during a period (this cost is independent of the number of users), and
- 4) $\tau = \{s_1, \dots, s_r\}$ is a family of fuzzy sets of N where $\tau_p = \tau_{s_p}$ is the membership function of each vertex p . For each $i \in N$,

$$\sum_{p \in V} \tau_p(i) \leq 1.$$

The family of cost-allocation tree problems with fuzzy agents is denoted as \mathcal{G} .

The fuzzy population s_p in a vertex p represents the membership level of the agents in vertex p , that is, for each $i \in N$, number $\tau_p(i)$, is the possibility of finding i in p .

Fig. 3 A cost-allocation problem in a fixed tree with fuzzy agents



From now on we identify each fuzzy set s with its membership function τ_s in order to simplify the notation of the paper. Therefore, we use τ as s if $\tau = \tau_s$.

Example 2 Let us assume three companies $N = \{1, 2, 3\}$ distributing a certain good obtained from a single point of production θ through a distribution and storage network with four nodes (out of this network each distributor then uses a particular network to reach its customers). Network $g = (V \cup \{\theta\}, L)$ is defined by $V = \{p_1, p_2, p_3, p_4\}$ and $L = \{\theta p_1, 12, 13, 34\}$. We are interested in how to distribute the maintenance expenses incurred in the common network during a month (those incurred in the individual networks would logically be paid by each of the distributors). The usage costs (maintenance in this case) of the links for a month (full time) are

$$c(p_1) = 12, \quad c(p_2) = 5, \quad c(p_3) = 6 \text{ and } c(p_4) = 3.$$

where $c(p_2)$ for instance represents the cost for using the corresponding link l_{p_2} . The fuzzy populations of the vertices with the needs of the agents on the network for this month are given by τ (1 is full time, i.e., all month),

$$\begin{aligned} \tau_{p_1} &= (0.5, 0.3, 0), & \tau_{p_2} &= (0, 0.3, 0), \\ \tau_{p_3} &= (0, 0.3, 1), & \tau_{p_4} &= (0.3, 0.1, 0). \end{aligned}$$

So, Distributor 3 uses only one of the nodes to supply its customers while the other two use several nodes. With this data, we obtain a cost-allocation tree problem with fuzzy agents, $\Gamma = (N, g, c, \tau)$.

Agents strive to organize visits to the nodes in order to minimize the usage costs.

Definition 2 Given a cost-allocation tree problem $\Gamma = (N, g, c, \tau) \in \mathcal{G}$, the cost of the problem is the minimum cost for the use of the network according to τ . This cost is denoted by $c(\Gamma)$.

Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$. We introduce tree g by the adjacency matrix of g , $e^g = (e_{pq}^g)_{(r+1) \times r}$ where $e_{pq}^g = 1$ if $pq \in L$, and $e_{pq}^g = 0$ otherwise. The first row is for θ , observe that there are not links with end node θ and therefore no new column needs to be added. The fuzzy population τ can also be introduced by a matrix $\tau_{r \times n}$ with $\tau_{pi} = \tau_p(i)$. Observe that agents can organize their needs for a period in any way. Each option determines a partition by levels of matrix τ . A *partition by levels* of τ is a finite set of pairs $(\lambda_k, d_k)_{k=1}^z$ where $\lambda_k \in (0, 1]$, and d_k is a partition matrix of

a subset N^k of N over V , that satisfies

$$\sum_{k=1}^z \lambda_k d_k = \tau.$$

Numbers λ_k represent those moments when certain agents decide to change their positions in the tree. Each d_k is identified with a cost-allocation tree problem $\gamma_k = (N^k, g, c, d_k)$ valid for a time of λ_k . The cost of the problem Γ is actually the minimum cost among all of the feasible partitions by levels,

$$c(\Gamma) = \min \left\{ \sum_{k=1}^z \lambda_k c(\gamma_k) : \sum_{k=1}^z \lambda_k d_k = \tau \right\}.$$

We will prove in Proposition 1 that algorithm given below obtains the worth of $c(\Gamma)$ as a dynamical optimization problem.

Algorithm TIME

Input: (N, g, c, τ) Output: $c(\Gamma)$

$c(\Gamma) = 0$

While $V \neq \emptyset$,

 Take $p \in V$ such that $\sum_{q \in V} e_{pq}^g = 0$

$t_p = \vee_{i \in N} \tau_{pi}$

$c(\Gamma) = c(\Gamma) + t_p c(p)$

 Let $q \in V$ with $e_{qp}^g = 1$, do

$\tau_{qi} = \tau_{qi} + \tau_{pi}$ for all $i \in N$

$V = V \setminus \{p\}$

Example 3 We apply *TIME* to the cost-allocation tree problem with fuzzy agents in Example 2. N, V, c were defined in the above example. The adjacency matrix e^g and function τ are introduced as

$$e^g = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}, \quad \tau = \begin{bmatrix} 0.5 & 0.3 & 0 \\ 0 & 0.3 & 0 \\ 0 & 0.3 & 1 \\ 0.3 & 0.1 & 0 \end{bmatrix}.$$

Nodes p that satisfy $\sum_{q \in V} e_{pq}^g = 0$ are the leaves of the tree, in our case p is p_2 or p_4 . Suppose we choose $p = p_4$, then $t_{p_4} = 0.3$ and $v(\Gamma) = 0.3 \cdot 3 = 0.9$. The only node with $e_{qp_4}^g = 1$ is $q = p_3$, and therefore row p_3 in matrix τ is modified as

$$\tau_{p_3} = (0, 0.3, 1) + (0.3, 0.1, 0) = (0.3, 0.4, 1).$$

Table 1 *TIME* algorithm

Step	p	t_p	$c(\Gamma)$	τ	V
1	p_4	0.3	0.9	$\tau_{p_3} = (0.3, 0.4, 1)$	$\{p_1, p_2, p_3\}$
2	p_2	0.3	2.4	$\tau_{p_1} = (0.5, 0.6, 0)$	$\{p_1, p_3\}$
3	p_3	1	8.4	$\tau_{p_1} = (0.8, 1, 1)$	$\{p_1\}$
4	p_1	1	20.4		\emptyset

We also modify set V as $V = \{p_1, p_2, p_3\}$ before the next step is begun. The following table shows the different steps of the algorithm, whereby in each row we can observe: the chosen node, the cost of the problem and how τ and V change. We obtain $c(\Gamma) = 20.4$ as result.

During the algorithm process, matrix τ changes. The last matrix τ with all these changes provides information about the use that each agent of each node to connect with the source.

Definition 3 Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$. The usage matrix of the problem is $(\tau_{pi}^{use})_{r \times n}$ where

$$\tau_{pi}^{use} = \sum_{q \in V_p} \tau_{qi}$$

The element τ_{pi}^{use} shows the time that agent i needs node p in order to connect to the source.

In Example 2 we first construct the usage matrix

$$\tau^{use} = \begin{bmatrix} 0.8 & 1 & 1 \\ 0 & 0.3 & 0 \\ 0.3 & 0.4 & 1 \\ 0.3 & 0.1 & 0 \end{bmatrix},$$

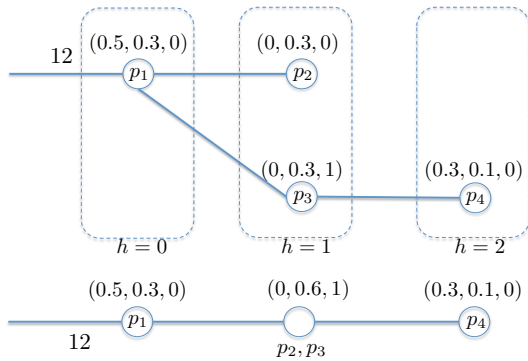
which is the last one in the algorithm (see Table 1). So, agent 2 needs link l_{p_3} at level 0.4.

Remark 1 Theoretically, the result of *TIME* is the same if we first add all the rows of the nodes in g_p , to each row p in τ namely if we first construct τ^{use} it is not to change τ in each step. In order to look for these subtrees, a similar process is needed, and hence this option is not better numerically.

Proposition 1 Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$. Algorithm *TIME* obtains the minimum cost for usage of the network according to τ .

Proof Clearly if there is only one node in V then *TIME* is efficient. Suppose it is true that the algorithm is efficient with $|V| = m - 1$ and we prove the result for $|V| = m$. Let $p \in V$ be a leaf (a node with maximal height in the tree). The link corresponding

Fig. 4 The chain of moves from a node



to this leaf must be used for at least the maximum time in τ_p . This means that all the agents in the support of τ_p must use the node while those with maximum time use it. Therefore, $c(p) \max\{\tau_{p_i} : i \in N\}$ is the minimal cost for using p . However, each time that an agent uses p , this agent must also use q with $l_p = qp$, and then this time is added to that node. We can delete node p . Using the induction hypothesis we get the result. \square

5 The fuzzy sequential full contributions rule

We follow Bird (1976) to define an allocation rule for cost-allocation problems with fuzzy agents in a fixed-tree with the same philosophy: if agents and nodes are identified in a tree network then each agent pays only for his corresponding link (the corresponding link of his node). In the model considered in Sect. 3, the sequential full contributions rule translates this payer role for each node to the minimal non-empty nodes moving away from that node. Observe that, given a fixed node, the agents in nodes that remain at a great distance from the minimal nodes “cop out” of paying for its use. When we have fuzzy agents in each node, agents have different needs. Let $\Gamma = (N, g, c, \tau)$ be a cost-allocation problem with fuzzy agents in a fixed-tree. Suppose $p \in V$ with row $\tau_p \neq 0$. Following Bird’s idea, while agents use the corresponding link l_p they have to pay and therefore we look for the minimum level in s_p and all the agents in the support of the fuzzy set must pay for this time. Then we find the next moment in s_p and agents in the support of the cut for this level pay for the differential time. Hence we are describing a Choquet integral (see Sect. 2). The question is what happens with the rest of agents in the support of τ_p while this node is being paid. They can take this time to stay in other nodes in V_p . We propose carrying out these moves by the following method: agents run away from the node covering their times in each height. Returning to Example 2 and take node $p = p_1$. We consider graph g_{p_1} and join nodes with the same height as in Fig. 4. This graph is denoted by $g_{p_1}^*$ (we call it *chain of moves*).

We transform τ only to allocate the cost of link l_{p_1} into the matrix

$$\tau^{p_1} = \begin{bmatrix} 0.5 & 0.3 & 0 \\ 0 & 0.6 & 1 \\ 0.3 & 0.1 & 0 \end{bmatrix}$$

representing the projection of τ to the chain of p_1 . For 0.3, node p_1 is inhabited by agents $\{1, 2\}$; following Bird, 0.15 of $c(p_1) = 12$ must therefore be paid by each of these players. Subsequently, only agent 1 stays at vertex p_1 . In this time, agent 3 uses any node in the following height (node 2 or 3) and he does not have to pay for node p_1 . At level 0.3, agent 2 changes his position and is now in any node in the second height, as is agent 3. So, the only inhabitant in node p_1 is agent 1, who must pay for the link for $0.5 - 0.3 = 0.2$ time. Meanwhile agents 2 and 3 use nodes in the second height and the next height if necessary. Hence, agent 1 pays for 0.35 time and agent 2 for 0.15. In this first step, we change matrix τ^{p_1}

$$\tau^{p_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.4 & 0.5 \\ 0.3 & 0.1 & 0 \end{bmatrix}.$$

Now V_p^{min} (this set was defined in the crisp case, it is analogous with fuzzy agents) are in the second height and the corresponding agents must pay for vertices $\{p_2, p_3\}$. Hence, they can be studied as only one vertex (see Fig. 4) and any option can therefore be used to delete time in the same height. The process is repeated. Coalition $\{2, 3\}$ pays until 0.4 and then 2 moves to the third height. Agent 1 is now in the last height and spends his needs of l_{p_1} in that height. Agent 1 then pays no more. After 0.4 the only inhabitant in the second height is agent 3, who pays for the use of l_{p_1} with coefficient 0.1. Agent 2 now uses the last height and then no longer has to pay for l_{p_1} . We obtain

$$\tau^{p_1} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}.$$

To the above payments, 0.2 is added to agent 2 and 0.3 to agent 3. Agents have to pay for this node $c(p_1) \cdot \max\{\tau_{li}^{use}\} = 12$ and this payment is allocated as $12 \cdot (0.35, 0.35, 0.3) = (4.2, 4.2, 3.6)$. The process is repeated with each node. The idea can be formalized.

First, we formalize the concept of chain of moves in a node.

Definition 4 Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$. Let $p \in V$ and $h_p = \vee\{h_{g_p}(q) : q \in V_p\}$. The projection by heights in p is a new tree g_p^* (a chain) formed from g_p with only one node q_h for each $h = 0, \dots, h_p$ and links $q_{h-1}q_h$ with $h = 1, \dots, h_p$. The chain of moves from p is the matrix τ^p defined as $\tau^p = (\tau_{q_h i}^p)_{(h_p+1) \times n}$ with

$$\tau_{q_h}^p = \sum_{\{q \in V_p : h_{g_p}(q) = h\}} \tau_q.$$

Second we introduce the capacity to allocate the payment equitably.

Definition 5 Let N be a finite set. For each agent $i \in N$, the equality capacity is $v_i : 2^N \rightarrow \mathbb{R}$ with

$$v_i(S) = \begin{cases} \frac{1}{|S|}, & \text{if } i \in S \\ 0, & \text{otherwise.} \end{cases}$$

Finally, an algorithm is introduced to decrease time in a tree from a node. We consider $t = (t_i)_{i \in N} \in [0, 1]^N$ as a vector with the time that can be reduced for each agent.

Algorithm WALK

Input: (N, g, τ, t, p) Output: τ

```

S = {i ∈ N : τpi ≠ ∨τp}
τpi = τpi ∧ (∨τp - ti) ∀i ∈ N
h = hg(p)
While S ≠ ∅ do
  h = h + 1
  V' = {q ∈ Vp : hg(q) = h}
  While V' ≠ ∅ do
    Take q ∈ V'
    For all i ∈ S do
      τqi = (τqi - ti) ∨ 0
      if ti > τqi do
        ti = ti - τqi
      else S = S \ {i}
    V' = V' \ {q}

```

A Bird's rule for fuzzy agents is designed using capacities v_i for the agents, the algorithm *WALK*, and the chains of moves.

Definition 6 Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$ be a cost-allocation tree problem with fuzzy agents. The fuzzy sequential full contributions rule of Γ is defined as

$$\beta(\Gamma) = \sum_{p \in V} c(p)\beta^p,$$

where $\beta^p = (\beta_i^p)_{i \in N}$ is the result of the following algorithm

Algorithm BIRD

Input: g_p^*, τ^p Output: β^p

$\beta^p = 0$

From $h = 0$ to $h = h_p$

For each $i \in N$ do

$$\beta_i^p = \beta_i^p + \int \tau_{q_h}^p dv_i$$

$$t_i = \vee \tau_{q_h}^p - \tau_{q_h}^p, t = (t_i)_{i \in N}$$

WALK(N, g_p^*, τ^p, t, q_h)

if $\sum_{i \in N} \beta_i^p = \vee \tau_p^{use}$ then STOP

Table 2 BIRD algorithm over node p_1

node q	q_0	q_1
$\int \tau_q^{p1} dv_i$	(0.35, 0.15, 0)	(0, 0.2, 0.3)
β^{p1}	(0.35, 0.15, 0)	(0.35, 0.35, 0.3)
t	(0, 0.2, 0.5)	(0.5, 0.1, 0)
τ^{p1}	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.4 & 0.5 \\ 0.3 & 0.1 & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$

Example 4 We revisit our example in Fig. 3. Remember that TIME determined that agents will pay $c(\Gamma) = 20.4$ at the end of the first month. We apply BIRD to each node. For p_1 , we previously obtained τ^{p1} with $h_{p_1} = 2$ and $\vee \tau_{p_1}^{use} = 1$.

$$\tau^{p1} = \begin{bmatrix} 0.5 & 0.3 & 0 \\ 0 & 0.6 & 1 \\ 0.3 & 0.1 & 0 \end{bmatrix}.$$

Table 2 shows how the algorithm works over node p_1 . Node q_3 is not used in the chain of moves of this node because when we arrive at node q_1 we obtain $\sum_{i \in N} \beta_i^{p1} = 1$.

For nodes p_2 and p_4 (the leaves), the algorithm uses only one node and it only consists of the application of the Choquet integral, $\beta^{p2} = (0, 0.3, 0)$, $\beta^{p4} = (0.25, 0.05, 0)$. Finally, we apply algorithm BIRD over node p_3 : in this case $h_{p_3} = 2$, $\vee \tau_{p_3}^{use} = 1$ and

$$\tau^{p3} = \begin{bmatrix} 0 & 0.3 & 1 \\ 0.3 & 0.1 & 0 \end{bmatrix}.$$

For the first node of the chain of moves we get $\beta^{p3} = (0, 0.15, 0.85)$ and then we have finished. The fuzzy sequential full contributions rule is

$$\begin{aligned} \beta(\Gamma) &= 12(0.35, 0.35, 0.3) + 5(0, 0.3, 0) + 6(0, 0.15, 0.85) \\ &\quad + 3(0.25, 0.05, 0) = (4.95, 6.75, 8.7). \end{aligned}$$

So, returning to the example of the distribution companies, the above vector indicates how much each of them must pay for the maintenance expenses incurred during the month.

6 Axiomatization of the rule

In this section, we introduce several properties of the fuzzy sequential full contributions rule. These properties will be used as axioms to determine the rule.

Let f be a rule for cost-allocation tree problems with fuzzy agents, namely a function that obtains a payoff vector $f(\Gamma) \in \mathbb{R}^N$ for each $\Gamma = (N, g, c, \tau)$. Obviously, we are searching for an allocation of the cost of the problem.

Efficiency. Rule f satisfies efficiency if, for all $\Gamma \in \mathcal{G}$, it holds

$$\sum_{i \in N} f_i(\Gamma) = c(\Gamma).$$

A node $p \in V$ is called *non-free* in Γ if $\sum_{q \in V^p} c(q) > 0$, namely not all the links in the path from the root to p have no cost. An agent $i \in N$ is a *null agent* in Γ if $\tau_{pi} = 0$ for all non-free node p . Null agents pay nothing.

Null agent. If $i \in N$ is a null agent in $\Gamma = (N, g, c, \tau)$ then $f_i(\Gamma) = 0$.

If two agents have the same needs in all of the non-free nodes of the tree, then we can suppose that they should pay the same quantity.

Equal treatment. If $i, j \in N$ satisfy $\tau_{pi} = \tau_{pj}$ for every non-free node $p \in V$ in a problem $\Gamma = (N, g, c, \tau)$ then $f_i(\Gamma) = f_j(\Gamma)$.

The following property implies good mathematical condition. Additivity implies a differentiated treatment for each problem even if the agents, the tree, and their needs are the same.

Additivity. For all $\Gamma, \Gamma' \in \mathcal{G}$ with $\Gamma = (N, g, c, \tau)$ and $\Gamma' = (N, g, c', \tau)$

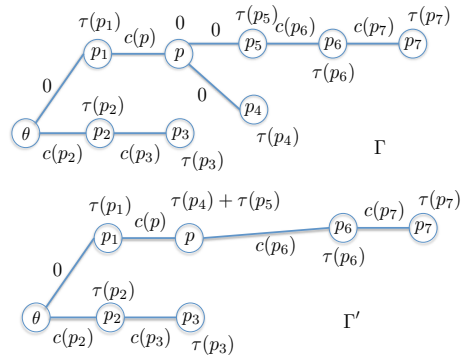
$$f(\Gamma) + f(\Gamma') = f(\Gamma + \Gamma')$$

where $\Gamma + \Gamma' = (N, g, c + c', \tau)$.

The contraction property in cost problems in trees describes when it is possible to contract a link by adding information from the last node to the first node. Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$. Suppose that $p \in V$ with $c(p) \neq 0$ and $\tau_p = 0$ satisfies $c(q) = 0$ if $q <_g p$ or $h_{g_p}(q) = 1$: that is, it is a minimal non-null node (with non-empty support in τ) in the tree and its nearest nodes are null. In this situation the elimination of the nodes is possible in the first height from p by contraction; we call a node under these conditions as *node in a contraction situation*. If p is a node in a contraction situation, then we denote $\Gamma' = (N, g', c', \tau')$ such that

- * $V' = V \setminus \{q \in V : h_{g_p}(q) = 1\}$
- * $L' = L \setminus \{pq : h_{g_p}(q) = 1\} \cup \{pq : h_{g_p}(q) = 1\}$
- * $c'(q) = c(q)$ for all $q \in V'$

Fig. 5 Height contraction



* $\tau'_q = \tau_q$ for all $q \in V' \setminus \{p\}$, and

$$\tau'_p = \sum_{\{q \in V : h_{g_p}(q)=1\}} \tau_q.$$

This notation may be cumbersome but the concept can be viewed in a simple way with a diagram, see Fig. 5.

Height contraction. Following the above notation, for all $\Gamma = (N, g, c, \tau) \in \mathcal{G}$ and $p \in V$ in a contraction situation, it holds $f(\Gamma) = f(\Gamma')$.

The last axiom is based on a known property of Bird’s rule: consistency in the first node (Chun and Lee 2012). However, in our case, the first node is considered to be the first step of payoff. We assume a problem $\Gamma = (N, g, c, \tau)$ with only one node p satisfying $c(p) \neq 0$. Let $\text{im}(\tau_p) = \{\lambda_1 < \dots < \lambda_m\}$ and $M(p) = \{i \in N : \tau_{pi} = \lambda_m\}$. We now introduce the following two new problems: $\Gamma^{\text{node } p} = (N, g, c, \tau^{\text{node } p})$ and $\Gamma^{\text{step } 1} = (N, g, c, \tau^{\text{step } 1})$, where

- * $\tau^{\text{node } p}_{pi} = \lambda_m - \lambda_{m-1}$ if $i \in M(p)$ and $\tau^{\text{node } p}_{qi} = 0$ otherwise.
- * $\tau^{\text{step } 1} = \text{WALK}(N, g, \tau, t, p)$ with $t_i = \lambda_m - \lambda_{m-1}$ for all $i \in N$.

Observe the new problems after the first step in the next example. Figure 6 has a unique node p_1 with non-null cost.

Consistency dictates that, in the example, 0.2 of $c(p_1)$ must be paid only by agent 1.

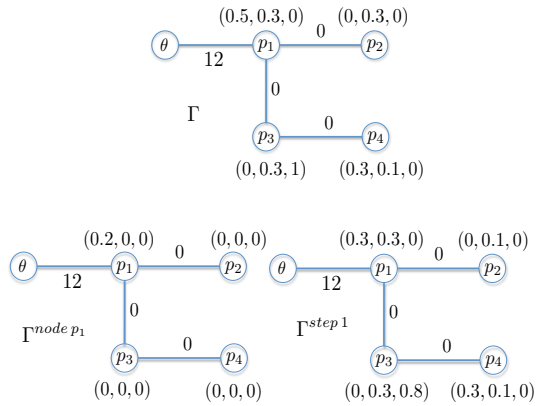
Consistency in the first step Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$ with only one non-null cost $c(p)$. Following the above notation,

$$f(\Gamma) = f(\Gamma^{\text{node } p}) + f(\Gamma^{\text{step } 1}).$$

We now prove that our rule satisfies all the properties.

Theorem 1 *The fuzzy sequential full contribution rule satisfies efficiency, null agent property, equal treatment property, additivity, height contraction and consistency in the first step.*

Fig. 6 Two new problems in consistency



Proof Let $\Gamma = (N, g, c, \tau) \in \mathcal{G}$ be a cost-allocation tree problem with fuzzy agents. **EFFICIENCY.** We know from *TIME* that

$$c(\Gamma) = \sum_{p \in V} \vee \tau_p^{use} c(p).$$

It is sufficient to test that $\sum_{i \in N} \beta_i^p = \vee \tau_p^{use}$. From Definition 3, we obtain

$$\vee \tau_p^{use} = \vee \left(\sum_{h=0}^{h_p} \tau_{q_h}^p \right)$$

for matrix τ^p at the beginning of algorithm *BIRD*. We denote $\tau^p(h)$ as the matrix τ^p at the end of step h . Following *BIRD*, we have

$$\vee \tau_p^{use} = \sum_{h=0}^{h_p} \vee \tau_{q_h}^p(h).$$

Observe that $\sum_{i \in N} v_i(S) = 1$ for every coalition $S \subseteq N$ from Definition 5. Hence, for each h we have

$$\sum_{i \in N} \int \tau_{q_h}^p(h) dv_i = \int \tau_{q_h}^p(h) d \sum_{i \in N} v_i = \vee \tau_{q_h}^p(h).$$

Hence

$$\begin{aligned} \sum_{i \in N} \beta_i^p &= \sum_{i \in N} \sum_{h=1}^{h_p} \int \tau_{q_h}^p(h) dv_i = \sum_{h=1}^{h_p} \sum_{i \in N} \int \tau_{q_h}^p(h) dv_i \\ &= \sum_{h=1}^{h_p} \vee \tau_{q_h}^p(h) = \vee \tau_p^{use}. \end{aligned}$$

NULL AGENT. If $i \in N$ is a null agent, then $\beta_i^p = 0$ for all non-free nodes.

EQUAL TREATMENT. Obviously if $\tau_{pi} = \tau_{pj}$ for every non-free node $p \in V$ then $\beta_i^p = \beta_j^p$ and then $\beta_i(\Gamma) = \beta_j(\Gamma)$.

ADDITIVITY. The calculation of β_i^p does not depend on the cost function c and hence β is a linear function over c .

HEIGHT CONTRACTION. Let $p \in V$ be a node in a contraction situation. It is not necessary to apply *BIRD* to nodes without cost (null cost), thus nodes in the family $\{q \in V : h_{g_p}(q) = 1\}$ appear only in the calculation of β^p . If we consider Γ and Γ' then vectors β^q are the same in both problems if $q \neq p$. For our node p , the chain of moves in Γ' contains one height less, q_1 in the chain of moves in Γ . Algorithm *BIRD* works in same way after the first step in Γ' and the second step in Γ . We test that the first two steps in *BIRD* for Γ obtain the same vector β^p as does the first step for Γ' . We apply algorithm *BIRD* over Γ . In the chain of moves of p , as $\tau_p = 0$, then the first step in *BIRD* obtains $\beta^p = 0$. In the second step, all the nodes in the above family are taken as only one q_1 with

$$\tau_{q_1}^p = \sum_{\{q \in V : h_{g_p}(q)=1\}} \tau_q.$$

So, after two steps the result for β^p in *BIRD* is

$$\beta_i^p = \int \tau_{q_1}^p dv_i.$$

Suppose now Γ' . The first step of *BIRD* obtains

$$\beta_i^p = \int \tau_p' dv_i = \int \sum_{\{q \in V : h_{g_p}(q)=1\}} \tau_q dv_i.$$

CONSISTENCY IN THE FIRST STEP. We assume that we have only one node p with $c(p) \neq 0$. Only one vector β^p is non-null. We denote the corresponding β^p in the algorithm *BIRD* for Γ , $\Gamma^{node\ p}$, $\Gamma^{step\ 1}$ as β^p , $\beta^p(node)$, $\beta^p(step)$. Let $im(\tau_p) = \{\lambda_1 < \dots < \lambda_m\}$ (and also consider $\lambda_0 = 0$). In the first step of *BIRD* we get

$$\beta_i^p = \int \tau_p dv_i = \begin{cases} (\lambda_m - \lambda_{m-1}) \frac{1}{|M(p)|} \\ + \sum_{k=1}^{m-1} (\lambda_k - \lambda_{k-1}) v_i([\tau_p]_k), & \text{if } i \in M(p) \\ \sum_{k=1}^{m-1} (\lambda_k - \lambda_{k-1}) v_i([\tau_p]_k), & \text{otherwise.} \end{cases}$$

Observe that the first step of *BIRD* for $\Gamma^{node\ p}$ gives

$$\begin{aligned} \beta_i^p(\text{node}) &= \int \tau_p^{node\ p} dv_i \\ &= \begin{cases} (\lambda_m - \lambda_{m-1}) \frac{1}{|M(p)|}, & \text{if } i \in M(p) \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

Following *WALK*, we obtain

$$\tau_{pi}^{step\ 1} = \tau_{pi} \wedge \lambda_{m-1} = \begin{cases} \lambda_{m-1}, & \text{if } i \in M(p) \\ \tau_{pi}, & \text{otherwise.} \end{cases}$$

Therefore, the first step of *BIRD* in $\Gamma^{step\ 1}$ obtains, for all agent i ,

$$\beta_i^p(\text{step}) = \int \tau_p^{step\ 1} dv_i = \sum_{k=1}^{m-1} (\lambda_k - \lambda_{k-1}) v_i([\tau_p]_k).$$

Hence we have

$$\beta^p = \beta^p(\text{node}) + \beta^p(\text{step}).$$

Moreover, the action of *WALK* for the other nodes in the chain of moves of p in Γ corresponds to the composition of *WALK* over τ with $t_i = \lambda_m - \lambda_{m-1}$ and *WALK* over $\tau^{step\ 1}$. Observe that *WALK* does not affect $\tau^{node\ p}$. □

Now we prove that there is only one rule that satisfies these properties.

Theorem 2 *The only rule for cost-allocation tree problems with fuzzy agents that satisfies efficiency, null agent property, equal treatment property, additivity, height contraction and consistency in the first step is the fuzzy sequential full contributions rule.*

Proof We take f as a rule for cost allocation tree problems with fuzzy agents satisfying all the axioms. Using additivity we can reduce the problem to $\Gamma = (N, g, c, \tau)$ with only one node $p \in V$ with $c(p) \neq 0$. In this case the set of non-free nodes of the problem is V_p .

Suppose first that p is a leaf in the tree, namely $V_p = \{p\}$. In fact, p is the only non-free node. If $\tau_p = 0$, then we apply the null agent property, obtaining only one payoff for each agent, $f(\Gamma) = 0$. Consider then $\tau_p \neq 0$. Let $im(\tau_p) = \{\lambda_1 < \dots < \lambda_m\}$ and $\lambda_0 = 0$. Using consistency in the first step we get two new problems from Γ . One problem is $\Gamma^{node\ p}$. We have two kinds of agents: those $i \in N$ with $\tau_{pi}^{node\ p} = \lambda_m - \lambda_{m-1}$, for which we denote the set as N^+ , and those $i \in N$ with $\tau_{pi}^{node\ p} = 0$. From the null agent property, the second family satisfy $f_i(\Gamma^{node\ p}) = 0$. Agents $i, j \in N^+$

in the first family verify $\tau_{pi}^{node p} = \tau_{pj}^{node p}$, hence $f_i(\Gamma^{node p}) = f_j(\Gamma^{node p}) = K$. From the efficiency we obtain

$$\sum_{i \in N} f_i(\Gamma^{node p}) = \sum_{i \in N^+} f_i(\Gamma^{node p}) = |N^+|K = c(\Gamma^{node p}).$$

There is therefore a unique payoff for each agent in $\Gamma^{node p}$. The other problem is $\Gamma^{step 1}$. Observe that now $im(\tau_p^{step 1}) = \{\lambda_1 < \dots < \lambda_{m-1}\}$. The application of consistency can be sequentially repeated, obtaining always one problem with a unique solution and another one with one element fewer in the image of row p . When we reach $\tau^{step 1}$ with $|im(\tau_p^{step 1})| = 1$ we get the uniqueness by again using the null agent property, the equal treatment property and efficiency.

Finally, suppose that p is not a leaf, i.e. $|V_p| > 1$. We apply consistency as before but with an additional application until $\tau_p^{step 1} = 0$. The solution again is unique for all $\tau^{node p}$ got in the process. We now apply height contraction because p is in state of contraction, and the payoffs are the same if we take τ' (see definition of contraction). In τ' we have deleted one height from p . If the idea is repeated several times, then we obtain a problem where p is a leaf and therefore the uniqueness follows as in the above situation. □

7 Discussion and conclusions

In this paper we have presented a new problem related to the maintenance of a network used by several agents. Once a minimum cost tree network is constructed the problem to be solved is how to distribute the costs due to the use of the network among the users. The classical form problem has been posed with a fixed population of agents at each node of the tree, Granot et al. (1996) assumed that there was a single player at each vertex while ? introduced the idea that at a vertex there may be a certain population. In reality, the incorporation of more than one agent in a node does not mean an important variation in the model, but this second version also allowed the appearance of vertices without population, which modified the formulation of the distribution rules and their axiomatizations. Our model generalizes both (an empty p node means in our case to take the fuzzy set with membership function $\tau_p = 0$) but also allows to analyze situations where agents have interests distributed over the set of nodes during the maintenance billing time. The construction of Bird’s rule for this case is a sample of how to approach the elaboration of partitioning solutions for these situations and the complications they entail from the algorithmic and axiomatic point of view.

The situation we have posed is restricted to considering known fixed costs per full-time billing period and assuming that the costs of any percentage of different usage have a proportional cost. Therefore such cost can be generalized either by an increasing cost function over the interval $[0, 1]$ or even with a fuzzy-valued cost function.

As mentioned above, the article focuses on maintenance costs once the tree is already built. An interesting problem that opens up in the future is to pose the same situation of agents with distributed interests in the vertices for the previous elaboration

of the network, i.e. a new version of the problem of sharing the costs of elaborating a minimum-cost spanning tree. The problem of finding a minimum cost spanning tree has been studied with fuzzy costs in Janiak and Kasperski (2008); Zhou et al. (2016) or Dehpande and Chaudhari (2020) while the distribution of the costs of such a tree by cooperative game theory has been addressed (with interval payments) by Moretti et al. (2011). The situation that may arise in view of the article is obviously not the same. In fact, it would not change the first step, i.e. finding a minimum cost tree to build the network. But the formulation of the cost sharing rule should be modified, since in this case the object represented by a fuzzy set is not the cost, but the situation of the agents.

The model can also be exported to problems not necessarily related to networks, such as the minimum cost consensus model (Gong et al. 2021). This article gives as an example the problem of finding a fair charge between coal companies and the State for the cost of ecological damage. Consideration could be given to taking into account as real agents the investors behind the companies, among whom there may be some with stakes in several of them.

Acknowledgements This research has been partially supported by the Spanish Ministry of Economy and Competitiveness project MTM2017-83455-P, and by the FQM237 grant of the Andalusian Government.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Bergantiños, G., & Gómez-Rúa, M. (2010). Minimum cost spanning tree problems with groups. *Economic Theory*, 43, 227–262.
- Bergantiños, G., & Vidal-Puga, J. (2007). A fair rule in minimum cost spanning tree problems. *Journal of Economic Theory*, 137(1), 326–352.
- Bird, C. G. (1976). On cost allocation for a spanning tree: A game theoretic approach. *Networks*, 6(4), 335–350.
- Bogomolnaia, A., Holzman, R., & Moulin, H. (2010). Sharing the cost of a capacity network. *Mathematics of Operations Research*, 35(1), 173–192.
- Choquet, G. (1953). Theory of capacities. *Annals de l'Institut Fourier*, 5, 131–295.
- Chun, Y., & Lee, J. (2012). Sequential contribution rules for minimum cost spanning tree problems. *Mathematical Social Sciences*, 64(2), 136–143.
- Claus, A., & Kleitman, D. J. (1973). Cost allocation for a spanning tree. *Networks*, 3(4), 289–304.
- Dai, Q., Li, Y., & Liang, L. (2016). Allocating fixed costs with considering the return to scale: A DEA approach. *Journal of Systems Science and Complexity*, 29(5), 1320–1341.
- de Waegenare, A., & Wakker, P. P. (2001). Nonmonotonic Choquet integrals. *Journal of Mathematical Economics*, 32(1), 45–60.
- Dehpande, A. A., & Chaudhari, O. K. (2020). Fuzzy approach to compare a minimal spanning tree problem by using various algorithms. *Advances in Fuzzy Mathematics*, 15(1), 47–58.

- Dutta, B., & Kar, A. (2000). Cost monotonicity, consistency and minimum cost spanning tree games. *Games and Economic Behavior*, 48(2), 223–248.
- Ertan, Z., Zhibin, T., Yunfei, B., & Zhigang, C. (2020). Minimum-cost spanning tree games: Bird rule revisited. *Scientia Sinica Mathematica*, 50(9), 1405.
- Feltkamp, V., Tijs, S. & Muto, S. (1994). Minimum cost spanning extension problems: the proportional rule and the decentralized rule. *Discussion paper 96*. Tilburg University. Center for Economic Research.
- Feltkamp, V., Tijs, S., & Muto, S. (2000). Bird's tree allocations revisited, in game practice: Contributions from applied game theory. *Theory and Decision Library*, 23, 75–89.
- Gellekom, A. van. (2000). *Sharing the cost in a network*. Ph.D. Thesis, Tilburg University.
- Gong, Z., Xu, X., Guo, W., Herrera-Viedma, E., & Cabrerizo, F. J. (2021). Minimum cost consensus modelling under various linear uncertain-constrained scenarios. *Information Fusion*, 66, 1–17.
- Granat, D., & Huberman, G. (1984). On the core and nucleolus of minimum cost spanning tree games. *Mathematical Programming*, 29(3), 323–347.
- Granat, D., Maschler, M., Owen, G., & Zhu, W. R. (1996). The kernel/nucleolus of a standard tree game. *International Journal of Game Theory*, 25(2), 219–244.
- Hougaard, J. L., Tvede, M., & Osterdal, L. P. (2013). *Cost sharing in chains and other fixed trees*. Discussion Papers of Business and Economics: University of Southern Denmark.
- Janiak, A., & Kasperski, A. (2008). The minimum spanning tree problem with fuzzy costs. *Fuzzy Optimization and Decision Making*, 7(2), 105–118.
- Juarez, R., & Kumar, R. (2013). Implementing efficient graphs in connection networks. *Economic Theory*, 54(2), 359–403.
- Kar, A. (2002). Axiomatization of the Shapley value on minimum cost spanning tree games. *Games and Economic Behavior*, 38(2), 265–277.
- Koster, M., Molina, E., Sprumont, Y., & Tijs, S. (2002). Sharing the cost of a network: Core and core allocations. *International Journal of Game Theory*, 30(4), 567–599.
- Meggido, N. (1978). Computational complexity of the game theory approach to cost allocation for a tree. *Mathematics of Operations Research*, 3, 189–196.
- Moretti, S., Gök, S. Z. A., Branzei, R., & Tijs, S. (2011). Connection situations under uncertainty and cost monotonic solutions. *Computers & Operations Research*, 38(11), 1638–1645.
- Norde, H. (2019). The degree and cost adjust folk solution for minimum cost spanning tree games. *Games and Economic Behavior*, 113, 734–742.
- Zadeh, L. A. (1965). Fuzzy sets. *Information and Control*, 8(3), 338–353.
- Zhou, J., Chen, L., Wang, K., & Yang, F. (2016). Fuzzy α -minimum spanning tree problem: Definition and solutions. *International Journal of General Systems*, 45(3), 311–355.