A characterization result in minimum cost spanning tree games

Stergios Athanassoglou [*] Jay Sethuraman[†]

June 2008

Abstract

We consider minimum cost spanning tree games. Assuming a priority ordering of the agents, we show that Bird's rule is the only rule that simultaneously satisfies efficiency, core-selection, tree invariance, and merge-proofness.[1]

1 Introduction

Motivation. This work is motivated by economic settings in which a collection of agents wishes to collaborate and jointly invest in the formation of a common network. For example, imagine a group of cities that wish to connect to a newly-built water reservoir. Each individual city could choose to go it alone and build a direct link to the reservoir, but such a decision would likely be highly inefficient. Instead, it may be cheaper for some cities to connect directly to the reservoir, whereas others could connect indirectly via links to neighboring cities. Indeed, an efficient network will be a tree structure of minimum total cost. Once such a network has been identified, a scheme for assigning costs to individual cities in order to cover the total investment will have to be devised. However, for the agents to agree to a cost allocation, the proposed rule will need to satisfy certain criteria of efficiency and fairness.

The tools of cooperative game theory have been applied very successfully to such cost-sharing problems and the reader is referred to Moulin [13] and Young [16] for excellent surveys on the subject. Stability of the proposed allocation is greatly important: Clearly, a central planner wants to avoid situations in which a subset of the users finds it in its best interest to deviate from the grand coalition and form its own sub-network to the source. Thus, a credible allocation rule

---

[*] IEOR Department, Columbia University, New York, NY; email: sa2164@columbia.edu

[†] IEOR Department, Columbia University, New York, NY; email: jay@ieor.columbia.edu

[1] Our work is independent of Ozsoy [14] who proves a similar result.

will need to ensure that no such group of agents will be able to, collectively, gain from such a maneuver. In cooperative game theory this notion is captured by the concept of core inclusion, which we will address in the next section. In general though, if an allocation is in the core of the market, then it is resistant to these sorts of sub-coalitional deviations. At first glance, one might think this to be too strict a requirement to enforce. Indeed, the core of a market can be empty or, even if non-empty, extremely expensive to compute. Fortunately, the model we are examining has a structure that ensures both the existence and computational tractability of the core. Thus, insisting on a rule that always selects a core allocation becomes a compelling issue.

An additional challenge for the mechanism designer arises when agents merge into a single mega-agent. For instance, in the case of the water reservoir, individual users may have the legal right to form an alliance and be treated as a single entity in any allocation decision. This would involve the construction of a minimum-cost sub-network, which would connect to the rest of the network in a way that is consistent with the initial tree solution. For a number of important reasons we would like an allocation rule to protect against this sort of merging behavior. First, it results in a network that has typically higher total cost than the original and so is inefficient. Second, in cases where agents may be legally constrained from forming alliances, requiring the cost-sharing procedure to be merge-proof ensures that the final allocation attains some measure of fairness. This is because, under the employed allocation rule, no group of agents feels that it could form an alliance and be better off than it currently is.

Contribution. Assuming a priority ordering of the agents to resolve cases that yield multiple minimum cost spanning trees, we prove a characterization result regarding the celebrated Bird [2] allocation rule. We show that it is the only rule that satisfies efficiency, core selection, tree invariance, and merge-proofness, all of which are defined in the next section. We note that our work is independent of Ozsoy [14], who proves the same result with minor differences.

Organization of the Paper. The rest of the paper is organized as follows. We discuss m.c.s.t. models more formally in Section 2 and provide an overview of the related work in Section 3. Section 4 provides an inductive proof of the characterization result and ends with a brief discussion of its tightness.

2 Model Description

2.1 Preliminaries

Let $\mathcal{N}$ denote the space of agents and 0 the source. Given an agent set $N \subset \mathcal{N}$ consider the natural ordering $\sigma(\cdot)$ over N so that $\sigma(i) \in \{1, 2, ..., n\}$ whenever $|N| = n$[2]. Agent i has higher priority than agent j if and only if $\sigma(i) < \sigma(j)$. We define the undirected graph $G = (N \cup 0, E)$ where each node corresponds to an agent or the source. From now on we write $N^+ = N \cup 0$. Next we connect all nodes in G to one another with an undirected link so that G becomes a complete undirected graph. The cost of an edge (i, j) is denoted by c_{ij}, so that $C_N = [c_{ij}]_{i,j \in N^+}$. This represents the cost of building a link to connect agents i and j. If $i \neq j$ then $c_{ij} > 0$, otherwise $c_{ii} = 0$ for all $i \in N^+$.

2.2 Prim's algorithm

There are many algorithms to compute a minimum cost spanning tree (m.c.s.t.) in a network. A prominent, and intuitive, one is Prim's algorithm [15]. Starting from a source node it builds a m.c.s.t. by greedily adding vertices to the existing tree. Ties are broken according to the predetermined priority ordering σ, which is specified in the input. We describe the algorithm formally:

```
Prim's Algorithm

1. Initialize $S = \{0\}, S^C = N$, $T = \emptyset$, $\sigma(\cdot)$.

2. Find $J \subseteq S^C$ so that all $j \in J$ satisfy $j = argmin_{i \in S, j \in S^C} c_{ij}$.
   Pick $j \in J$ so that $\sigma(j)$ is minimized.
   If there exist more than one $i \in S$ such that $c_{ij} = argmin_{i \in S} c_{ij}$, pick
   $i$ so that $\sigma(i)$ is minimized.

3. Update $S \leftarrow S \cup j$, $S^C \leftarrow S^C \setminus j$, $T \leftarrow T \cup (i, j)$.

4. Repeat steps 2 and 3 until $S = N$. STOP.
```

For a more in-depth treatment of Prim's algorithm and other m.c.s.t. algorithms the reader may refer to Ahuja et al [1]. While the network is undirected, we take the m.c.s.t. to be rooted at the source node 0.

---

[2] $\sigma(0)$ is set to 0

2.3 Allocation Rules and Properties

A cost matrix C_N and a priority ordering over N[3] give rise to a unique m.c.s.t. T_{C_N}. This tree can be computed efficiently by running Prim's greedy algorithm. Once a m.c.s.t. has been identified, the question arises: How should the agents share the cost of building the entire m.c.s.t.? An allocation rule ψ is a mapping that assigns an individual cost to each agent so that the total cost of connecting everyone to the source is covered. Let $\mathcal{C}_N$ denote the space of all $|N| \times |N|$ cost matrices.

Definition 1 *An allocation rule is a family of functions* $\{\psi^N\}_{N \subset \mathcal{N}}$ *such that* $\psi^N : (\mathcal{C}_N) \rightarrow \Re^N_+$ *that satisfies:*

$$\sum_{i \in N} \psi^N_i(C) \geq \sum_{e \in T_C} c_e, \ \ \forall C \in \mathcal{C}_N$$

Since a cost matrix implicitly defines an agent set, and to keep notation less cumbersome, we from now on drop N as a superscript. An example of an allocation rule would be to assign every agent the same cost by splitting the total cost of the m.c.s.t. evenly among all agents, i.e. set $\psi_i = \sum_{e \in T_{C_N}} c_e / |N|$ for all $i \in N$. At the other extreme, a rule could simply pick an agent at random and assign the whole cost of the m.c.s.t. to that agent. We now discuss properties that allocation rules may satisfy.

Definition 2 *An allocation rule* ψ *is efficient if it satisfies*

$$\sum_{i \in N} \psi_i(C) = \sum_{e \in T_C} c_e, \ \ \forall C \in \mathcal{C}_N$$

Efficiency ensures that agents are collectively charged exactly the cost of the m.c.s.t. that connects them.

Definition 3 *An allocation rule* ψ *is tree invariant if it is insensitive to changes in the cost matrix, which do not alter the m.c.s.t.*

Tree invariance amounts to requiring that the cost allocation only depend on the structure of the m.c.s.t. and not on the cost matrix that induces it.

Consider a coalition $S \subseteq N$ of agents and the restriction C_{S^+} of the cost matrix to nodes in $S \cup 0 = S^+$. The priority ordering σ is similarly restricted to elements in S^+. Let $T_{C_{S^+}}$ refer to the m.c.s.t. corresponding to C_{S^+} and denote its cost by $C(T_{C_{S^+}})$. A prominent cooperative

---

[3] From now on we suppress explicit dependence on the priority ordering.

game-theoretic property is that of core selection. Intuitively, a rule satisfies core selection if it always assigns costs in such a way that no sub-coalition of agents will find it in its best interest to deviate from the grand coalition and form its own m.c.s.t. to the source. More formally:

Definition 4 *A rule ψ satisfies core selection if*

$$\sum_{i \in S} \psi_i(C) \leq C(T_{C_{S+}}), \ \ \forall C \in \mathcal{C}_N, S \subseteq N$$

If this inequality fails for some coalition S, then by deviating and connecting to the source independently it will achieve a strictly positive cost savings for the agents in S collectively. On the other hand, if the inequality is reversed, then in any cost reallocation within S some agent will be strictly worse off than before. Thus, if a rule satisfies core selection it provides a cost-sharing solution, which is stable to sub-coalition deviations.

2.4 Merge-proofness

Now suppose that a coalition of agents S wishes to pool its resources and act as a single entity. Such merging behavior was first considered by Claus and Kleitman [3]. It can be formally described by a graph transformation $G \leftarrow G_S^m$. First, we alter the initial node set by merging all nodes in S into a node i_S so that $N_S^m = N \setminus S \cup i_S$. We impose that $i_S = \text{argmin}_{i \in S} \sigma(i)$, so that the merged node inherits the label of the highest priority node in the set S, so that the priority ordering is updated accordingly. In this graph we connect all nodes to one another and update edge costs in the following way:

$$\begin{aligned}(c_S^m)_{ij} &\leftarrow c_{ij}, \ \ \forall i, j \neq i_S \\ (c_S^m)_{i_S j} &\leftarrow \min_{i \in S}\{c_{ij}\}, \ \ \forall j \notin S\end{aligned}$$

Consider the restriction C_S of the cost matrix to nodes in S. Let T_{C_S} refer to the m.c.s.t. corresponding to C_S and denote its cost by $C(T_{C_S})$[4]. We are now ready to state our definition of merge-proofness.

Definition 5 *An allocation rule ψ is merge-proof if it satisfies*

$$\sum_{i \in S} \psi_i(C) \leq \psi_{i_S}(C_S^m) + C(T_{C_S}), \ \ \forall C \in \mathcal{C}^N, S \subseteq N$$

---

[4] Note that $0 \notin S$, so we are dealing with the m.c.s.t. that connects all nodes in S to one another, but not to the source.

2.5 Special Allocation Rules

Consider a cost matrix C_N along with a priority ordering over the agents, and the m.c.s.t. T_{C_N} that they jointly induce. In T_{C_N}, each vertex v has a unique predecessor, call it $p(v)$. The Bird allocation B is defined to be:

$$B_v(C_N) = c_{p(v),v}, \ \ \forall v \in N$$

The logic behind it is that each agent is charged the marginal cost of efficiently connecting her to the network. Bird [2] showed that this rule satisfies core selection.

There are a number of other rules one can use to compute an allocation. Many of them involve cooperative game theoretic concepts, since the underlying cost sharing problem can be cast as a cooperative game. Each coalition of agents S can achieve a certain outcome by deviating from the grand coalition and connecting to the source by itself. So it would be natural to consider solution concepts such as the nucleolus and the Shapley value, and that has already been done in the literature (Granot and Huberman [8], Kar [10]). A notable result is due to Dutta and Kar [4], who characterize a special rule (referred to as the Dutta-Kar rule) with regard to efficiency, and two additional consistency and monotonicity properties.

3 Related Work

Minimum cost spanning tree problems have been extensively studied. First introduced by Claus and Kleitman [3], their structure lends itself to cooperative game theoretic analysis and indeed, such efforts have been vigorously pursued in the literature. Emphasis has been placed on both algorithmic as well as structural results, and we first briefly discuss the former. Megiddo [11, 12] proved that the Shapley value and nucleolus can be efficiently computed. Granot and Huberman [8] develop an efficient algorithm to compute core allocations, whereas Granot and Granot [6] develop a strongly polynomial algorithm for computing core allocations in minimum cost spanning forest problems. More recently, Iwata and Zuiki [9] showed that the nucleolus and egalitarian solutions can be derived in $O(n \log n)$ time.

On the structural side, Bird [2] and Granot and Huberman [7] showed that minimum cost spanning tree games have a non-empty core. Feltkamp et al [5] consider the irreducible core of these games and provide an algorithm that generates all of its elements. Their mechanism, which specializes to the Bird rule, is characterized by certain consistency properties. Kar [10] shows that the Shapley value of these games is characterized by efficiency, fairness, and regularity properties. In a related paper Dutta and Kar [4] introduce a new allocation rule and show that it is characterized by appealing monotonicity and consistency axioms. They also prove a similar result for the Bird rule.

4 Characterization Result

In this section we prove that Bird's rule is characterized by efficiency, tree invariance, core selection, and merge-proofness. Throughout, we assume a priority ordering over agents. We begin with two lemmas.

Lemma 1 *Consider a cost matrix* $C \in \mathcal{C}_N$ *and the m.c.s.t.* T_C *that it induces. Let* $v \in N$ *and consider its descendant node set* ${D_v}^5$ *within the tree* T_C*. Denote* $v \cup D_v = D_v^+$ *and the restriction of* C *on* D_{v^+} *by* $C_{D_{v^+}}$*, which induce a m.c.s.t.* $T_{C_{D_{v^+}}}$*. We have:*

$$\{(i,j) \in T_C : (i,j) \in D_{v^+} \times D_{v^+}\} = T_{C_{D_{v^+}}}, \ \ \forall v \in N$$

Proof. Suppose the result is not true for some $v \in N$. Alter the m.c.s.t. by deleting all edges in $\{(i,j) \in T_C : (i,j) \in D_{v^+} \times D_{v^+}\}$ and replacing them with $T_{C_{D_{v^+}}}$. This tree will have cost strictly less than the T_C, contradicting the fact that T_C is a m.c.s.t. for the original problem. ■

Lemma 2 *Consider a cost matrix* $C \in \mathcal{C}_N$ *and the m.c.s.t.* T_C *that it induces. Assume that a coalition of agents* $S \subseteq N$ *merges into a node* i_S*. Then,*

$$B_{i_S}(C^m_S) = B_{v_1}(C)$$

where $v_1 \in S$ *is the first node to have been included in* T_C *by Prim's algorithm.*

Proof. When S merges into i_S, we obtain a new graph and cost matrix C^m_S where

$$\begin{aligned}(c^m_S)_{ij} &\leftarrow c_{ij}, \ \ \forall i, j \neq i_S \\ (c^m_S)_{i_S j} &\leftarrow \min_{i \in S}\{c_{ij}\}, \ \ \forall j \notin S\end{aligned}$$

C^m_S gives rise to a new m.c.s.t. $T_{C^m_S}$. Let us run Prim's algorithm to compute $T_{C^m_S}$. Let t_1 be the first time-step that a node in S, by definition this node is v_1, was added to the old m.c.s.t T_C. Assume node v_1's immediate predecessor was node u, where by definition $u \notin S$. For $t < t_1$, the

---

[5] This set includes all of v's children, and their children, and so and so forth. Recall, the tree is rooted at the source node 0.

two trees will be identical. At time t, node i_S will be added to $T_{C_S^m}$ from the same predecessor node u. Therefore we will have:

$$B_{i_S}(C_S^m) = c_{uv_1} = B_{v_1}(C)$$

■

We are now ready to prove our main result.

Theorem 3 *A rule ψ satisfies efficiency, tree invariance, core selection, and merge-proofness if only if $\psi = B$.*

Proof.

[If] It is trivial to note that Bird's rule is tree invariant and efficient. It has also been shown that it produces an allocation that is in the core (see Bird [2] and Granot and Huberman [7]). All that remains to show is that it is merge-proof.

To this end, consider a cost matrix C_N and the m.c.s.t. T_{C_N} it gives rise to and let $S = \{v_1, v_2, ..., v_k\} \subseteq N$. Given Lemma 2 assume without loss of generality that, after these vertices merge, we have $B_{i_S}(C_S^m) = B_{v_1}(C)$.

Now further consider the m.c.s.t. T_{C_S}, rooted at v_1 and connecting vertices in S. Let $C(T_{C_S})$ denote the cost of the edges in T_{C_S}.

Assume for the sake of deriving a contradiction that merge-proofness is violated for this coalition S. So we have the following holding:

$$\begin{aligned} B_{i_S}(C_S^m) + C(T_{C_S}) &< \sum_{i \in S} B_i(C) \\ \Rightarrow B_{v_1}(C) + C(T_{C_S}) &< B_{v_1}(C) + \sum_{i \in S \setminus v_1} c_i \Rightarrow C(T_{C_S}) < \sum_{i \in S \setminus v_1} c_i \end{aligned}$$

where c_i is the cost of the edge going into node i in the minimum cost spanning tree T_C.

Now consider altering the original tree T_C in the following way:

- Delete all edges (i, j) where $j = v_2, v_3, ..., v_k$.
- Insert the tree T_{C_S} and root it at node v_1.

We keep all other edges unchanged. This new tree will have strictly less cost than T_C, thereby contradicting that T_C is a minimum cost spanning tree. Hence, we conclude that Bird's rule is merge-proof. ■

[**Only if**] The argument will proceed by induction on the number of agents, i.e. the number of vertices in the network. Assume that the allocation rule ψ satisfies efficiency, tree invariance, core selection and merge-proofness. We will show that $\psi = B$.

Base Case - The inverted Y tree. Since merging requires at least two nodes in addition the source, the base case consists of networks with 3 vertices (including the source node 0). Let us refer to them as vertices 0, 1, and 2 and assume that $\sigma(1) = 1$ and $\sigma(2) = 2$. It is clear that for cost matrices that induce a m.c.s.t. of height 1 the only efficient allocation which is in the core is the Bird allocation. Therefore, let us restrict our attention to cost matrices that induce a m.c.s.t that forms a line with two edges. Setting $c_{01} < c_{02}$ and $c_{12} < c_{02}$[6], yields such a m.c.s.t with edges (0,1) and (1,2) as depicted in Figure 1. Our goal is to show that the rule ψ must produce an allocation such that $\psi_1 = c_{01}$ and $\psi_2 = c_{12}$.

To this end let us add an additional dummy vertex, call it vertex d, to our original graph and assume that $c_{1d} < c_{12} \leq c_{2d}$, $c_{0d} > c_{1d}$ and $c_{01} < c_{0d} < c_{02}$. Furthermore we impose the priority ordering $\sigma(1) = 1, \sigma(2) = 3, \sigma(d) = 2$, so that node d has higher priority than 2. The m.c.s.t. becomes an inverted Y graph consisting of edges $(0,1), (1,2), (1,d)$ as seen in Figure 1. We refer to the merge-proof, efficient, tree invariant core allocation of this problem as allocation A.

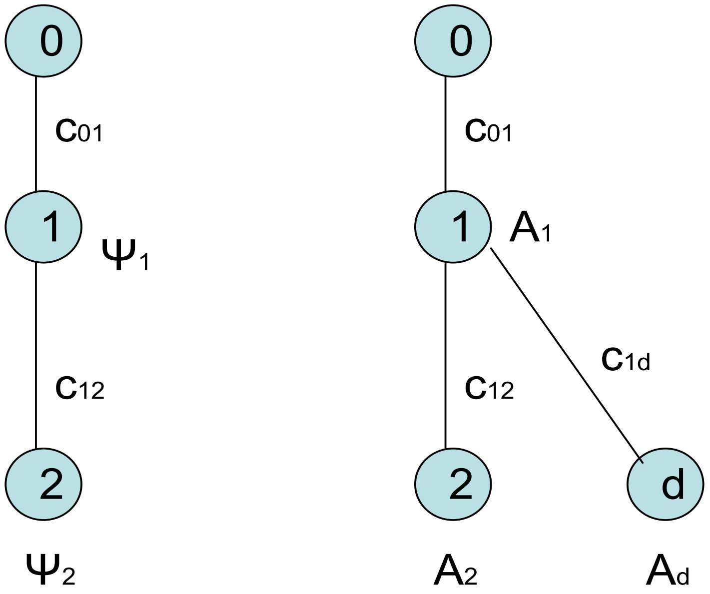

Figure 1: M.c.s.t. before and after insertion of dummy node d

---

[6] Since we are proving a characterization result, there is no harm in focusing on restricted domains.

We consider the three possible ways in which we can merge 2 vertices in this graph[7]. In particular, we may do the following:

(a) Merge nodes 1 and 2. Since $\sigma(1) < \sigma(2)$, we get a graph with vertex set $\{0, 1, d\}$ and edge costs $c^*_{01} = c_{01}$, $c^*_{1d} = c_{1d}$, $c^*_{0d} = c_{0d}$. These induce a m.c.s.t. that is a 2-edge line $0 - 1 - d$. Let us refer to the allocation that the rule ψ gives rise to in this problem as D.

(b) Merge nodes 1 and d. Since $\sigma(1) < \sigma(d)$ we get a graph with vertex set $\{0, 1, 2\}$ and edge costs $c^*_{01} = c_{01}$, $c^*_{02} = c_{02}$ and $c^*_{12} = c_{12}$. These induce a m.c.s.t. that is the 2-edge line $0 - 1 - 2$. Let us refer to the allocation that the rule ψ gives rise to in this tree as E. Note that this is the same problem that we started out with (same node set and cost matrix) so that $E_1 = \psi_1, E_2 = \psi_2$.

(c) Merge nodes 2 and d. Since $\sigma(d) < \sigma(2)$, we get a graph with vertex set $\{0, 1, d\}$ and edge costs $c^*_{01} = c_{01}$, $c^*_{1d} = c_{1d}$, and $c^*_{0d} = c_{0d}$. These induce a m.c.s.t. that is the 2-edge line $0 - 1 - d$. Let us refer to the allocation that the rule ψ gives rise to in this tree as F. Note that this is the same problem as (a) (node set and cost matrix) so that $D_1 = F_1$ and $D_d = F_d$.

The results of these merging manoeuvres are depicted in Figure 2. Initially, let us focus on tree (a). By mergeproofness we obtain:

$$D_1 + c_{12} \geq A_1 + A_2$$

By efficiency we have:

$$D_d = c_{01} + c_{1d} - D_1$$

Thus we get the following inequality:

$$\begin{aligned} D_d &= c_{01} + c_{1d} - D_1 \leq c_{01} + c_{12} + c_{1d} - A_1 - A_2 \\ &= A_1 + A_2 + A_d - A_1 - A_2 = A_d \Rightarrow \\ D_d &\leq A_d \end{aligned}$$

Recall that $D = F \Rightarrow D_d = F_d$. Applying mergeproofness on graph (c) we get:

$$F_d + c_{2d} \geq A_2 + A_d \Rightarrow A_2 \leq D_d - A_d + c_{2d} \leq c_{2d}$$

---

[7] For simplicity denote the merged edge costs with an asterisk

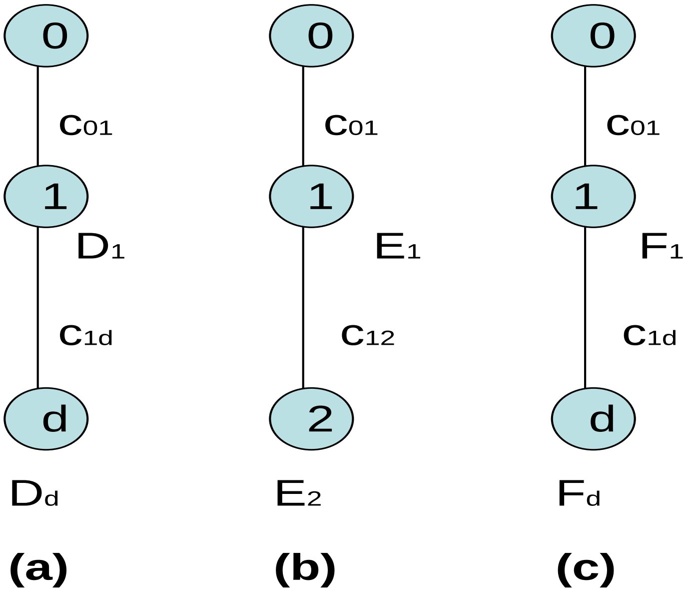

Figure 2: The three merged graphs

We note that changing the value of c_{2d} to $c_{12}+\delta$, for any $\delta \geq 0$, leaves the m.c.s.t. unchanged[8]. Thus, tree invariance implies that the allocation A should also remain unaffected. So, we may substitute c_{2d} with $c_{12}+\delta$ in the previous inequality to obtain

$$A_2 \leq c_{12}+\delta$$

By taking $\delta = 0$ his implies that $A_2 \leq c_{12}$.

It is fairly straightforward to establish the other side of the inequality. Since A is in the core, we have $A_1 + A_d \leq c_{01} + c_{1d}$. This, coupled with the fact that $A_1 + A_2 + A_d = c_{01} + c_{12} + c_{1d}$ implies that $A_2 \geq c_{12}$. Together with the previous inequality we obtain:

$$A_2 = c_{12} \Rightarrow A_1 + A_d = c_{01} + c_{1d}$$

By mergeproofness on problem (b) we have that:

$$\begin{aligned} E_1 + c_{1d} &\geq A_1 + A_d \Rightarrow \\ E_1 + c_{1d} &\geq c_{01} + c_{1d} \Rightarrow \\ E_1 &\geq c_{01} \end{aligned}$$

---

[8] For $\delta = 0$ the priority ordering is such that Prim's algorithm includes edge (1,2) in the tree and not (2,d). In any case, we could also impose $c_{2d} > c_{12}$, write $\delta > 0$ and take the limit as $\delta \downarrow 0$ to establish the same result.

Furthermore, inclusion in the core implies $E_1 \leq c_{01}$. Thus we get $E_1 = c_{01}$, which by efficiency also implies $E_2 = c_{12}$. As $E = \psi$ we have the following holding:

$$\psi_1 = c_{01}, \psi_2 = c_{12}$$

This concludes the proof of the base case. ■

The Main Induction Argument. The proof will proceed in two steps. Assume the result is true for $|N| < n$.

Step 1 - Implications of core selection and merge-proofness. First we begin with two helpful lemmas. We assume throughout a priority ordering σ that is updated in the appropriate way when nodes merge.

Lemma 4 *Consider a vertex v, its predecessor node $p(v)$, and its descendant node set D_v. Denote $v \cup D_v = D_v^+$. Core-selection and efficiency imply that*

$$\sum_{i \in D_v^+} \psi_i \geq c_{p(v),v} + C(T_{C_{D_{v^+}}}) = c_{p(v),v} + \sum_{(i,j) \in D_{v^+} \times D_{v^+} : (i,j) \in T_C} c_{ij}$$

Proof. Consider the coalition $N \setminus D_v^+$. Core selection implies $\sum_{i \in N \setminus D_v^+} \psi_i \leq C(T_{C_{[N \setminus D_v^+]+}})$, whereas efficiency dictates $\sum_{i \in N} \psi_i = C(T_C)$. These two facts yield

$$\sum_{i \in D_v^+} \psi_i \geq C(T_C) - C(T_{C_{[N \setminus D_v^+]+}}) = c_{p(v),v} + C(T_{C_{D_v^+}})$$

By Lemma 1 we observe that $C(T_{C_{D_{v^+}}}) = \sum_{(i,j) \in D_{v^+} \times D_{v^+} : (i,j) \in T_C} c_{ij}$ and the result is established.
■

Lemma 5 *Consider a non-leaf vertex v, its predecessor node $p(v)$, and its descendant node set D_v. Denote $v \cup D_v = D_v^+$. Merge-proofness implies that*

$$\sum_{i \in D_v^+} \psi_i \leq c_{p(v),v} + C(T_{C_{D_v^+}}) = c_{p(v),v} + \sum_{(i,j) \in D_{v^+} \times D_{v^+} : (i,j) \in T_C} c_{ij}$$

Proof. By merge-proofness we obtain:

$$\sum_{i \in D_v^+} \psi_i \leq \psi_{i_{D_v^+}}(C^m_{D_v^+}) + C(T_{C_{D_v^+}})$$

By the induction hypothesis $\psi_{i_{D_v^+}}(C^m_{D_v^+}) = B_{i_{D_v^+}}(C(T_{C^m_{D_v^+}}))$. Now by Lemma 2, $B_{i_{D_v^+}}(C(T_{C^m_{D_v^+}})) = B_v(C) = c_{p(v),v}$. By Lemma 1, $C(T_{C_{D_{v^+}}}) = \sum_{(i,j) \in D_{v^+} \times D_{v^+} : (i,j) \in T_C} c_{ij}$ and the result is established. ■

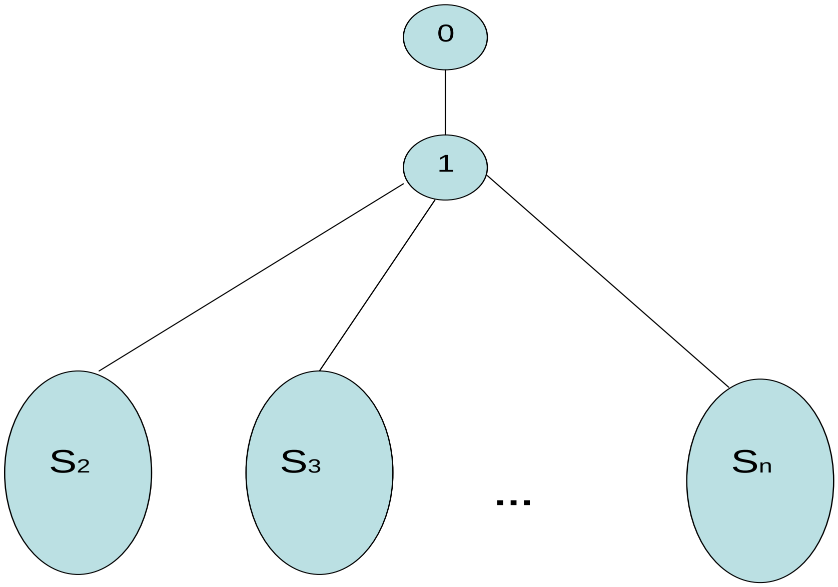

Figure 3: The m.c.s.t. and its subtrees

Step 2 - The main induction. We now proceed to the main inductive argument. For simplicity let us consider a tree where the root has only one child. The argument can be easily extended to the general case. This can be done since we can repeat the same argument for each subtree rooted at the root's children. This is because, by efficiency and inclusion in the core, the sum of the allocations in each subtree will have to equal the edge costs of that particular subtree

Thus, let us proceed with the canonical one-child example and refer to the single node at height 1 as node 1, and its descendants as nodes $2, 3, ..., n$. Each descendant node can be thought of as the root of a sub-tree with node sets (including its root) $S_2, S_3, ..., S_n$. For a subset of agents S, let $C(S)$ denote the cost of all incoming edges in the m.c.s.t. into nodes in S. We focus on node 1 and distinguish between 2 cases:

Case 1: Node 1 has no children which are leaves. By inclusion in the core we obtain $\psi_1 + \sum_{i \in S_2} \psi_i \leq c_{01} + C(S_2)$, where $C(S)$ is the sum of the costs of all edges going into nodes in S. This together with efficiency implies that $\sum_{i \in S_3,...,S_n} \psi_i \geq C(S_3, ..., S_n)$. By Lemma 5, we get: $\sum_{i \in S_j} \psi_i \leq C(S_j)$ for all $j = 3, ..., n$, implying $\sum_{i \in S_3,...,S_n} \psi_i \leq C(S_3, ..., S_n) \Rightarrow \sum_{i \in S_3,...,S_n} \psi_i = C(S_3, ..., S_n) \Rightarrow \psi_1 + \sum_{i \in S_2} \psi_i = c_{01} + C(S_2)$. Again applying Lemma 5, this time on S_2 we get $\sum_{i \in S_2} \psi_i \leq C(S_2) \Rightarrow \psi_1 \geq c_{01}$. But by inclusion in the core we have $\psi_1 \leq c_{01}$, hence we get $\psi_1 = c_{01}$.

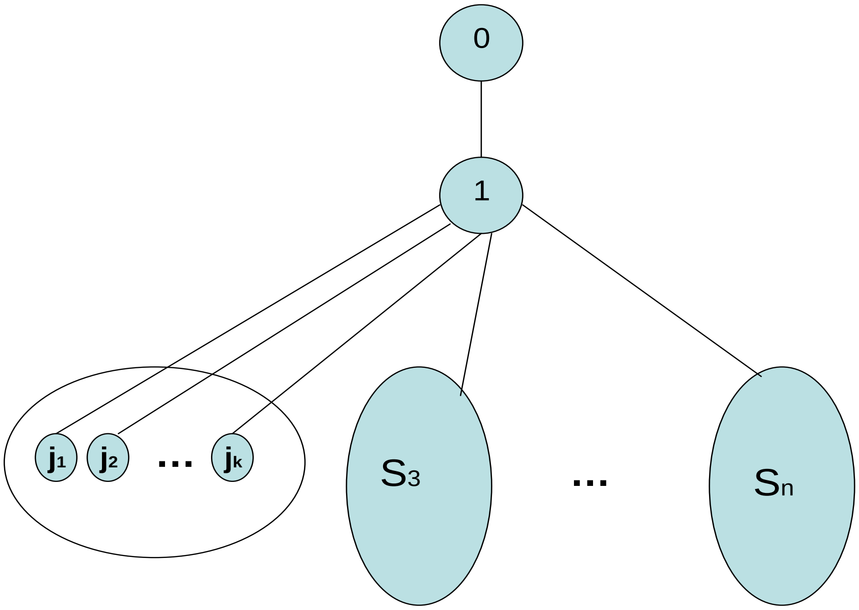

Figure 4: Node 1 has leaf children

Case 2: Node 1 has children which are leaves. In general, we can repeat the above procedure iteratively until we encounter the predecessors of leaf nodes. Then we can no longer perform the same argument since leaf nodes have no descendants that we can merge them with. Thus we obtain a subgraph which can be a 2-vertex line, or a 3 vertex Y-graph, or a multiple vertex Y-graph -according to the number of leaves whose predecessor is node 1. Let us label the nodes of this subgraph $j_1, j_2, ..., j_k$ and set $\tilde{S} = \{1, j_1, j_2, ..., j_k\}$. We note that node 1 may have other non-leaf children as well. However, we focus on $\tilde{S}$. Without loss of generality, let $c_{1,j_1} \leq c_{1,j_2} \leq ... \leq c_{1,j_k}$.

By inclusion in the core we obtain: $\psi_1 + \psi_{j_1} + ... + \psi_{j_k} \leq c_{01} + c_{1j_1} + ... + c_{1j_k}$. This implies that $\sum_{i \notin \tilde{S}} \psi_i \geq C(S_3, ..., S_n)$. But then we can separately merge the nodes in $S_3, S_4, ..., S_n$, apply Lemma 5 to each and, upon summing everything together, obtain the reverse inequality $\sum_{i \notin \tilde{S}} \psi_i \leq C(S_3, ..., S_n)$ so that $\sum_{i \notin \tilde{S}} \psi_i = C(S_3, ..., S_n)$. This implies that $\psi_1 + \psi_{j_1} + ... + \psi_{j_k} = c_{01} + c_{1j_1} + ... + c_{1,j_k}$.

For the moment, we assume that $|\tilde{S}| > 2$. Mergeproofness applied to $\{j_{k-1}, j_k\}$ implies that:

$$\psi_{j_{k-1}} + \psi_{j_k} \leq \psi_{i_S} + c_{j_{k-1},j_k}$$

By the induction hypothesis and Lemma 2 we obtain $\psi_{i_S} = c_{1,j_{k-1}}$. So we may write:

$$\psi_{j_k} + \psi_{j_{k-1}} \leq c_{1,j_{k-1}} + c_{j_{k-1},j_k} \Rightarrow \psi_{j_k} \leq c_{1,j_{k-1}} - \psi_{j_{k-1}} + c_{j_{k-1},j_k}$$

Applying Lemma 4 on node j_{k-1} we finally obtain:

$$\psi_{j_k} \leq c_{j_{k-1},j_k}$$

Now by tree invariance, we would obtain the same allocation for any value of c_{j_{k-1},j_k} that keeps the m.c.s.t. unchanged. This is true as long as $c_{j_{k-1},j_k} \geq c_{1,j_k}$. So we may write:

$$\psi_{j_k} \leq c_{1,j_k} + \delta, \ \ \forall \delta > 0$$

By taking the limit as $\delta \downarrow 0$ we obtain $\psi_{j_k} \leq c_{1,j_k}$. By Lemma 4, we have $\psi_{j_k} \geq c_{1,j_k}$ and so this yields $\psi_{j_k} = c_{1,j_k}$.

We use a recursive argument to establish $\psi_{j_i} = c_{1,j_i}$ for all $j_i \in \{j_1, ..., j_{k-1}\}$. First, Lemma 4 applied to nodes in $S_i = \{j_i, ..., j_k\}$ implies $\sum_{j \in S_i} \psi_j \geq \sum_{j \in S_i} c_{1,j}$. On the other hand, merge-proofness applied to S_i and tree invariance establish $\sum_{j \in S_i} \psi_j \leq \sum_{j \in S_i} c_{1,j}$. Thus we obtain $\sum_{j \in S_i} \psi_j = \sum_{j \in S_i} c_{1,j}$ for all $i = 2, ..., j_k$. Since we have argued that $\psi_{j_k} = c_{1,j_k}$ these equalities allow us to recurse backwards and conclude $\psi_{j_i} = c_{1,j_i}$ for all $j_i \in \{j_1, ..., j_{k-1}\}$. As $\psi_1 + \psi_{j_1} + ... + \psi_{j_k} = c_{01} + c_{1j_1} + ... + c_{1,j_k}$, we obtain $\psi_1 = c_{01}$.

Finally, if $|\tilde{S}| = 2$ we can appeal to the reasoning in the base case of the proof to obtain the equivalent result for two nodes. This concludes the proof. ■

Independence of the four properties. We now briefly exhibit the tightness of the characterization result by showing that none of the properties are implied by the presence of the other three. First, we note that the Dutta-Kar rule [4] satisfies efficiency (EF), core selection (CS), and tree invariance (TI) but fails merge-proofness (MP). A rule which satisfies CS, TI, and MP while violating EF would be the following modification of Bird's rule. Let $\psi^*(C, \sigma) = B$ for all (C, σ) that give rise to at least one binding MP and CS constraint. For all other (C, σ) pick the last agent that was added to the m.c.s.t. and set her allocation to $B_i + \delta$ where δ is small enough so that all the MP and CS inequality constraints are still satisfied. A rule which satisfies EF, TI, MP but fails CS would be to assign the cost of the whole m.c.s.t. to the agent with the highest priority. Finally we note that Ozsoy [14] provides an example of a rule that satisfies EF, CS, and MP while failing TI.

References

[1] Ahuja, Magnanti and Orlin ,“ Network Flows: Theory, Algorithms, and Applications”, Prentice Hall, 1993.

[2] C.J. Bird (1976), “On cost allocation for a spanning tree”, *Networks*, 3, 335–350.

[3] A. Claus and D. J. Kleitman (1973), “Cost allocation for a spanning tree”, *Networks*, 6, 335–350.

[4] B. Dutta and A. Kar (2004), “Cost monotonicity, consistency, and minimum cost spanning tree games”, *Games and Economic Behavior*, 48, 223–248.

[5] V. Feltkamp, S. Muto, and T. Tijs (1994), “On the irreducible core and the equal remaining obligations rule of minimum cost spanning extension games”, *mimeo*

[6] D. Granot and F. Granot (1993), “On computational complexity of a cost allocation approach to a fixed cost spanning forest problem”, *Mathematics of Operations Research*, 17, 765–780.

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

[8] D. Granot and G. Huberman (1984), “On the core and nuclelous of minimum cost spanning tree games”, *Mathematical Programming*, 29, 323–347.

[9] S. Iwata and N. Zuiki (2004), “A network flow approach to cost allocation for rooted trees”, *Networks*, 44, 297-301.

[10] A. Kar (2002), “Axiomatization of the Shapley value in minimum cost spanning tree games”, *Games and Economic Behavior*, 38, 265–277.

[11] N. Megiddo (1978), “Cost allocation for Steiner trees”, *Networks*, 8, 1–6.

[12] N. Megiddo (1978), “Computational complexity of the game theory approach to cost allocation for a tree”, *Mathematics of Operations Research*, 3, 189–196

[13] H. Moulin (1999), “Axiomatic cost and surplus sharing”, *In K.J. Arrow, A.K. Sen, K. Suzumura (Eds.), Handbook of Social Choice and Welfare*, North Holland

[14] H. Ozsoy (2006), “A characterization of Bird’s rule”, *mimeo.*

[15] R.C. Prim (1957), "Shortest connection network and some generalization", *Bell System Technical Journal*, 36, 1389–1401.

[16] H.P. Young (1994), "Cost Allocation", *In R. Aumann, Hart S. (Eds), Handbook of Game Theory with Economic Applications*, North Holland.