Minimum Cost Arborescences*

Bhaskar Dutta[†] Debasis Mishra[‡]

May 28, 2011

Abstract

In this paper, we analyze the cost allocation problem when a group of agents or nodes have to be connected to a *source*, and where the cost matrix describing the cost of connecting each pair of agents is not necessarily symmetric, thus extending the well-studied problem of minimum cost spanning tree games, where the costs are assumed to be symmetric. The focus is on rules which satisfy axioms representing incentive and fairness properties. We show that while some results are similar, there are also significant differences between the frameworks corresponding to symmetric and asymmetric cost matrices.

JEL Classification Numbers: D85, C70

Keywords: directed networks, cost allocation, core stability, continuity, cost monotonicity.

*We are most grateful to two referees and an associate editor whose extensive comments have significantly improved the paper, and to Ibrahim Baris Esmerok for pointing out an error in an earlier version of the paper. We thank Daniel Granot, Anirban Kar, and Herve Moulin for comments.

†Corresponding author. Department of Economics, University of Warwick, Coventry CV4 7AL, England. Email: `B.Dutta@warwick.ac.uk`

‡Indian Statistical Institute, 7 S.J.S. Sansanwal Marg, New Delhi 110016, India. Email: `dmishra@isid.ac.in`

1 Introduction

In a variety of contexts, a group of users may be jointly responsible for sharing the total cost of a joint "project". Often, there is no appropriate "market mechanism" which can allocate the total cost to the individual agents. This has given rise to a large literature which describes axiomatic methods in distributional problems involving the sharing of costs and benefits, the axioms typically representing notions of fairness. In the vast bulk of this literature, the agents have no particular "positional" structure. However, there is a large number of practical problems in which it makes sense to identify the agents with nodes in a graph. Consider, for instance, the following examples.

(i) Multicast routing creates a directed network connecting the source to all receivers; when a packet reaches a branch point in the tree, the router duplicates the packet and then sends a copy over each downstream link. Bandwidth used by a multicast transmission is not directly attributable to any one receiver, and so there has to be a cost-sharing mechanism to allocate costs to different receivers.[1]

(ii) Several villages are part of an irrigation system which draws water from a dam, and have to share the cost of the distribution network. The problem is to compute the minimum cost network connecting all the villages either directly or indirectly to the *source*, i.e. the dam (which is a computational problem), and to distribute the cost of this network amongst the villages.

(iii) In a *capacity synthesis* problem, the agents may share a network for bilateral exchange of information, or for transportation of goods between nodes. Traffic between any two agents i and j requires a certain capacity t_{ij} (width of road, bandwidth). The cost allocation problem is to share the minimum cost of a network in which each pair i and j is connected by a path in which each edge has a capacity of at least t_{ij}. [2]

The combinatorial structure of these problems raises a different set of issues (for instance computational complexity) and proof techniques from those which arise when a network structure is absent. Several recent papers have focused on cost allocation rules appropriate for *minimum cost spanning networks*.[3] In these networks, the agents are each identified with distinct nodes, and there is an additional node (the "source"). Each agent has to be connected either directly or indirectly to the source through some path. A *symmetric* cost

[1] See, for instance, Herzog et al. (1997).

[2] See Bogomolnaia et al. (2010), who show that under some assumptions, the capacity synthesis problem is similar, though not identical, to the minimum cost spanning tree problem.

[3] See, for instance, Bird (1976), Bergantinos and Vidal-Puga (2007a), Bergantinos and Kar (2010), Bogomolnaia and Moulin (2010), Branzei et al. (2004), Feltkamp et al. (1994), Kar (2002), Branzei et al. (2005).

matrix specifies the cost of connecting each pair of nodes. Obviously, the cheapest graph connecting all nodes to the source must be a tree rooted at the node. The cost allocation problem is to assign the total cost of the minimum cost spanning tree to the agents.

The assumption that the cost matrix is symmetric implies that the spanning network can be represented as an undirected graph. However, in several situations the cost of connecting agent i to agent j may not be the same as the cost of connecting agent j to agent i. The most obvious examples of this arises in contexts where the geographical position of the nodes affect the cost of connection. For instance, the villages in the second example may be situated at different *altitudes*. In the capacity synthesis problem, the nodes may be towns located along a river, so that transportation costs depend on whether the towns are upstream or downstream. In this paper, we extend the previous analysis by permitting the cost matrix to be asymmetric. The spanning tree will then be a *directed* graph and is called an *arborescence*. The minimum cost arborescence can be computed by means of an algorithm due to Chu and Liu (1965) and Edmonds (1967). This algorithm is significantly different from the algorithms (Prim's and Kruskal's algorithms) used to compute a minimum cost spanning tree in the symmetric case. However, from a computational perspective, the algorithm for finding a minimum cost arborescence is still a *polynomial time* algorithm.

Our interest is in the cost sharing problem. Following the literature on cost allocation for minimum cost spanning tree problems, we too focus on an axiomatic approach, the axioms representing a combination of incentive and fairness properties. The first property is the well-known *Stand-Alone Core property*. This requires that no group of individuals be assigned costs which add up to more than the total cost that the group would incur if it built its own subnetwork to connect all members of the group to the source. We provide a constructive proof that the *core* is non-empty by showing that the directed version of the *Bird Rule*,[4] due not surprisingly to the seminal paper of Bird (1976), yields an allocation which belongs to the core of the cost game. Of course, Bird himself had proved the same result when the cost matrix is symmetric. We then prove a result which shows that the set of cost allocation rules that are core selections and which satisfy an invariance condition (requiring that the allocation be invariant to costs of edges not figuring in any minimum cost arborescence) assign each individual a cost which is at least the minimum cost assigned by the set of Bird Rules. This also means that there can be only one such rule when the cost matrix is such that it gives a unique minimum cost arborescence - namely the Bird Rule itself. Of course, this result has no parallel in the minimum cost tree problems, and emphasizes the difference in the two frameworks.

We then go on to impose two other "minimal" or "basic" requirements - *Continuity*, which requires that the cost shares depend continuously on the cost matrix, and a *Monotonicity*

[4]The Bird Rule is defined with respect to a specific tree, and stipulates that each agent pays the cost of connecting to her predecessor.

requirement which requires that each individual is "primarily" responsible for the cost of his or her incoming edges. We interpret this to mean the following - if the only difference between two cost matrices is that the cost of an incoming edge of agent i goes up, then the cost share of i should (weakly) go up by at least as much as that of any other agent. This property is slightly stronger than the monotonicity condition which was initially defined by Dutta and Kar (2004), and subsequently used in a number of papers on symmetric cost matrices.[5] We construct a rule which satisfies these three basic properties, using Bird's concept of *irreducible* cost matrices. In particular, we show that the Shapley value of the cost game corresponding to the irreducible cost matrix constructed by us satisfies these three properties.

Readers familiar with the literature on the original minimum cost spanning tree problem will immediately recognize that this is exactly the procedure adopted by Bergantinos and Vidal-Puga (2007a) to construct the "folk solution" for minimum cost spanning tree problems.[6] Indeed, the solution constructed by us actually coincides with the folk solution on the class of symmetric cost matrices. However, despite the coincidence on the restricted class of matrices, it is important to realize that the folk solution belongs to a very different class of rules from the one that we construct in this paper. In particular, Bird's irreducible cost matrix is *uniquely* defined for any symmetric cost matrix.[7] The construction of this irreducible cost matrix only uses information about the costs of edges figuring in some minimum cost spanning tree - costs of edges not figuring in a minimum cost spanning tree are irrelevant. This obviously means that the folk solution too does not utilize all the information contained in the original cost matrix. Indeed, this forms the basis of the critique of "reductionist" solutions (solutions which only utilize information about the costs of edges figuring in some minimum cost spanning tree) by Bogomolnaia and Moulin (2010).

In contrast, we show that in our framework, the irreducible cost matrix constructed by us (and hence our solution) requires more information than is contained in the minimum cost arborescence(s). This is one important sense in which our solution is qualitatively different from the folk solution. We go on to highlight another important difference. We show that our solution satisfies the directed version of *Ranking*, a property due to Bogomolnaia and Moulin (2010). Our version of Ranking is the following. If the costs of all incoming edges of i are higher than the costs of corresponding edges of j, and the corresponding outgoing edges of i and j are the same, then i should pay strictly more than j. Bogomolnaia and Moulin (2010)

[5]See, for instance, Bergantinos and Vidal-Puga (2007a), Bergantinos and Vidal-Puga (2007b).

[6]This is a term coined by Bogomolnaia and Moulin (2010) because this allocation rule has been independently proposed and analyzed in a number of papers. See, for instance, Bergantinos and Vidal-Puga (2007a), Bergantinos and Vidal-Puga (2007b), Bogomolnaia and Moulin (2010), Branzei et al. (2004), Feltkamp et al. (1994), Norde et al. (2001) Branzei et al. (2005).

[7]We show by means of an example that there can be an infinity of irreducible cost matrices corresponding to a symmetric cost matrix in our framework.

point out that all reductionist solutions in the symmetric case - and hence the folk solution - must violate Ranking.

We also provide characterization results for our cost allocation rule on restricted classes of cost matrices. The extension of these characterizations to the entire domain of cost matrices is an open question which we hope to resolve in subsequent work.

Our results demonstrate that there are significant differences between the frameworks corresponding to symmetric and asymmetric cost matrices, and emphasizes the need for more systematic analysis of the cost allocation problem for minimum cost arborescences.

2 Framework

Let $N = \{1, 2, \ldots, n\}$ be a set of n agents. We are interested in *directed graphs* or *digraphs* where the nodes are elements of the set $N^+ \equiv N \cup \{0\}$, where 0 is a distinguished node which we will refer to as the *source*. We assume that the set of edges of such digraphs come from the set $\{ij : i \in N^+, j \in N, j \neq i\}$, where ij is the directed edge from i to j. Notice that we ignore edges of the form ii, as well as of edges from any $i \in N$ to the source. We will also have to consider digraphs on some subsets of N^+. So, for any set $S \subsetneq N$, let S^+ denote the set $S \cup \{0\}$. Then, a digraph on S^+ consists of a set of *directed edges* out of the set $\{ij : i \in S^+, j \in S, i \neq j\}$.

A typical graph[8] over S^+ will be represented by g_S whose edges are out of the set $\{ij : i \in S^+, j \in S\}$. When there is no ambiguity about the set S (usually when we refer to a graph on N^+), we will simply write g, g' etc instead of g_S, g'_S.

A *cost matrix* $C = (c_{ij})$ for N^+ represents the cost of various edges which can be constructed from nodes in N^+. That is, c_{ij} is the cost of the edge ij. We assume that each $c_{ij} \geq 0$ for all ij. Note that the cost of an edge ij need not be the same as that of the edge ji - the direction of the edge does matter. In fact, this distinguishes our approach from the literature on minimum cost spanning tree problems. Given our assumptions, each cost matrix is nonnegative, and of order $n + 1$. The set of all cost matrices for N is denoted by $\mathcal{C}_N$. For any cost matrix C, denote the cost of a graph g as $c(g)$. That is,

$$c(g) = \sum_{ij \in g} c_{ij}$$

Similarly, $c'(g)$ will denote the cost of the graph g when the cost matrix is C'.

A *path* in g is a sequence of distinct nodes $(i^1, \ldots, i^K)$ such that $i^j i^{j+1}$ is an edge in g for all $1 \leq j \leq K - 1$. If $(i^1, \ldots, i^K)$ is a path, then we say that it is a path from i^1 to i^K using edges $i^1 i^2, i^2 i^3, \ldots, i^{K-1} i^K$. A *cycle* in g is a sequence of nodes $(i^1, \ldots, i^K, i^{K+1})$ such that $(i^1, \ldots, i^K)$ is a path in g, $i^K i^{K+1}$ is an edge in g, and $i^1 = i^{K+1}$.

[8] Henceforth, we will use the term "graph" to denote digraphs. Similarly, we will use the term "edge" to denote a directed edge.

A node i is *connected* to node j if there is a path from node j to node i. Our interest is in graphs in which every agent in N is connected to the source 0.

DEFINITION **1** *A graph g is an arborescence rooted at 0 for N if and only if g contains no cycle and every node $i \in N$ has only one incoming edge.*

Let A_N be the set of all arborescences for N. A *minimum cost arborescence* (MCA) corresponding to cost matrix C is an arborescence g such that $c(g) \leq c(g')$ for all $g' \in A_N$. We describe later on a recursive algorithm whose output will be an MCA for any given cost matrix.

Let $M(C)$ denote the set of minimum cost arborescences corresponding to the cost matrix C for the set N, and $T(C)$ the total cost associated with any element $g \in M(C)$. While our main interest is in minimum cost arborescences for N, we will also need to define the minimum cost of connecting *subsets* of N to the source 0. The set of arborescences for any subset S of N will be denoted A_S and the set of minimum cost arborescences will be represented by $M(C, S)$.

Clearly, a minimum cost arborescence is analogous to a minimum cost spanning tree (MCST) for undirected graphs. Alternatively, an MCA may be viewed as a generalization of an MCST when the cost matrix is not symmetric.

2.1 The Cost Allocation problem

The total cost of an MCA corresponding to any cost matrix C is typically less than the cost of directly connecting each agent to the source. So, the group as a whole gains from cooperation. This raises the issue of how to distribute the cost savings amongst the agents or, what is the same thing, how to allocate the total cost to the different agents.

DEFINITION **2** *A cost allocation rule is a function $\mu : \mathcal{C}_N \to \Re^N$ satisfying $\sum_{i \in N} \mu_i(C) = T(C)$ for all $C \in \mathcal{C}_N$.*

So, for each cost matrix, a cost allocation rule specifies how the total cost of connecting all agents to the source should be distributed. Notice that our definition incorporates the notion that the rule should be *efficient* - the costs distributed should be exactly equal to the total cost.

In this paper, we follow an axiomatic approach in defining "fair" or "reasonable" cost allocation rules. The axioms that we will use in this paper reflect concerns for "stability", fairness and computational simplicity.

The notion of stability reflects the view that any specification of costs must be acceptable to all groups of agents. That is, no coalition of agents should have a justification for feeling that they have been overcharged. This leads to the notion of the *core* of a specific cost allocation game.

Consider any cost matrix C on N^+. Let $g \in M(C)$. The set of all agents incur a total cost of $c(g)$ to connect each node to the source. Consider any subset S of N, and assume that if S "threatens" to build its own MCA, then it can only use nodes in S itself. Then, S incurs a corresponding cost of $c(g_S)$ where $g_S \in M(C, S)$. It is natural to assume that agents in any subset S will refuse to cooperate if an MCA for N is built and they are assigned a total cost which exceeds $c(g_S)$ - they can then issue the credible threat of building their own MCA.

So, each cost matrix C yields a cost game (N, c) where

$$\text{for each } S \subseteq N, c(S) = c(g_S) \text{ where } g_S \in M(C, S).$$

The *core* of a cost game (N, c) is the set of all allocations x such that

$$\text{for all } S \subseteq N, \sum_{i \in S} x_i \leq c(S), \sum_{i \in N} x_i = c(N)$$

We will use $Co(N, C)$ to denote the core of the cost game corresponding to C.

Definition **3** *A cost allocation rule μ is a Core Selection (CS) if for all C, $\mu(C) \in Co(N, C)$.*

A rule which is a core selection satisfies the intuitive notion of stability since no group of agents can be better off by rejecting the prescribed allocation of costs.

The next couple of axioms are essentially properties which help to minimize the computational complexity involved in deriving a cost allocation. The first property requires the cost allocation to depend only on the costs of edges involved in the MCAs. That is, if two cost matrices have the same set of MCA s, and the costs of edges involved in these arborescences do not change, then the allocation prescribed by the rule should be the same.

For any N, say that two cost matrices C, C' are *arborescence equivalent* if (i) $M(C) = M(C')$, and (ii) if ij is an edge in some MCA, then $c_{ij} = c'_{ij}$.

Definition **4** *A cost allocation rule μ satisfies Independence of Irrelevant Costs (IIC) if for all C, C', $\mu(C) = \mu(C')$ whenever C and C' are arborescence-equivalent.*

The next axiom is a stronger independence axiom which requires that the cost allocation of a node must only depend on the incoming edge costs of that node. This was introduced into the literature on MCST games by Bergantinos and Vidal-Puga (2007a).

Definition **5** *A cost allocation rule μ satisfies Independence of Other Costs (IOC) if for all $i \in N$ and for any pair of cost matrices $C, C' \in \mathcal{C}_N$ with $c_{ji} = c'_{ji}$ for all $j \in N^+ \setminus \{i\}$, we have $\mu_i(C) = \mu_i(C')$.*

A form of fairness requires that if all incoming edge costs to some node i go up uniformly, while the edge costs of other nodes remain the same, then i should be fully responsible for the increase in its incoming costs.

Definition 6 *A cost allocation rule μ satisfies Invariance(INV) if for all cost matrices C and C' such that for some $j \in N$ and for all $i \in N^+ \setminus \{j\}$, $c'_{ij} = c_{ij} + \epsilon$, and $c'_{pq} = c_{pq}$ for all $q \in N \setminus \{j\}$ and for all $p \in N^+ \setminus \{p\}$, we have $\mu_j(C') = \mu_j(C) + \epsilon$, and $\mu_q(C') = \mu_q(C)$ for all $q \in N \setminus \{j\}$.*

Remark 1 *Note that IOC implies INV.*

In the present context, a fundamental principle of fairness requires that each agent's share of the total cost should be monotonically related to the vector of costs of its own *incoming* edges. So, if the cost of say edge ij goes up, and all other edges cost the same, then j's share of the total cost should not go down. This requirement is formalized below.

Definition 7 *A cost allocation rule satisfies Direct Cost Monotonicity (DCM) if for all C, C' and for all $i \in N^+, j \in N$, if $c_{ij} < c'_{ij}$, and for all other edges $kl \neq ij$, $c_{kl} = c'_{kl}$, then $\mu_j(C') \geq \mu_j(C)$.*

This is the counterpart of the assumption of Cost Monotonicity introduced by Dutta and Kar (2004). Clearly, DCM is also compelling from the point of view of incentive compatibility. If DCM is not satisfied, then an agent has an incentive to inflate costs (assuming an agent is responsible for its incoming edge costs).

Notice that DCM permits the following phenomenon. Suppose the cost of some edge ij goes up while other edge costs remain the same. Then, the allocation rule charges individual j an additional amount of ϵ, but charges some other individual k an additional amount exceeding ϵ. This is clearly against the spirit of the principle that each individual node is primarily responsible for its vector of incoming costs. The next axiom rules out this possibility.

Definition 8 *A cost allocation rule satisfies Direct Strong Cost Monotonicity (DSCM) if for all C, C' and for all $i \in N^+, j \in N$, if $c_{ij} < c'_{ij}$ and for all other edges $kl \neq ij$, $c_{kl} = c'_{kl}$ we have $\mu_j(C') - \mu_j(C) \geq \mu_k(C') - \mu_k(C)$ for all k.*

Remark 2 *Note that IOC implies DSCM, which in turn implies DCM. IOC is an extremely stringent requirement, and we do not know of any reasonable cost allocation rule which satisfies this condition on the full domain of cost matrices.*

Our next axiom is a symmetry condition which requires that if two nodes i and j have identical vectors of costs of incoming and outgoing edges, then their cost shares should not differ.

Definition 9 *A cost allocation rule μ satisfies Symmetry (S) if for all $i, j \in N$ and for all cost matrices C with $c_{ki} = c_{kj}$ for all $k \in N^+ \setminus \{i, j\}$, $c_{ik} = c_{jk}$ for all $k \in N \setminus \{i, j\}$, and $c_{ij} = c_{ji}$, we have $\mu_i(C) = \mu_j(C)$.*

A stronger version of symmetry stipulates that two agents pay the same cost if they have the same incoming edge cost, even if they do not have the same outgoing edge costs.

Definition 10 *A cost allocation rule μ satisfies Strong Symmetry (SS) if for all $i, j \in N$ and for all cost matrices C with $c_{ki} = c_{kj}$ for all $k \in N^+ \setminus \{i, j\}$ and $c_{ij} = c_{ji}$, we have $\mu_i(C) = \mu_j(C)$.*

The *Ranking* axiom is adapted from Bogomolnaia and Moulin (2010). Ranking compares cost shares across individual nodes and insists that if costs of all incoming edges of i are uniformly higher than the corresponding costs for j, while the costs of outgoing edges are the same, then i should pay strictly more than j. Notice that it is similar in spirit to the monotonicity axioms since it too implies that nodes are "primarily" responsible for their incoming costs.

Definition 11 *A cost allocation rule μ satisfies Ranking (R) if for all $i, j \in N$ and for all cost matrices C with $c_{ik} = c_{jk}$ for all $k \in N \setminus \{i, j\}$, and $c_{ki} > c_{kj}$ for all $k \in N^+ \setminus \{i, j\}$, and $c_{ji} > c_{ij}$, we have $\mu_i(C) > \mu_j(C)$.*

The next couple of axioms are straightforward.

Definition 12 *A cost allocation rule μ satisfies Non-negativity (NN) if $\mu_i(C) \geq 0$ for all $i \in N$ and for all $C \in \mathcal{C}_N$.*

Definition 13 *A cost allocation rule μ satisfies Continuity (CON) if for all $\epsilon > 0$, there is $\delta > 0$ such that for all C, C' if $|c_{ij} - c'_{ij}| \leq \delta$ for all $ij \in \{pq | p \in N^+, q \in N, p \neq q\}$, then $|\mu_i(C) - \mu_i(C')| \leq \epsilon$ for all $i \in N$.*

3 A Partial Characterization Theorem

In the context of minimum cost spanning tree problems, Bird (1976) is a seminal paper. Bird defined a specific cost allocation rule - the Bird Rule, and showed that the cost allocation specified by his rule belonged to the core of the cost game, thereby providing a constructive proof that the core is always non-empty.[9]

[9] Bird defined his rule for a specific MCST. Since a cost matrix may have more than one MCST, a proper specification of a "rule" can be obtained by, for instance, taking a convex combination of the Bird allocations obtained from the different minimum cost spanning trees.

In this section, we show that even in the directed graph context, the Bird allocations belong to the core of the corresponding game. We then show that if a cost allocation rule satisfies IIC and CS, then the cost allocation of each agent is at least the minimum cost paid by the agent in different Bird allocations. In particular, this implies that such a cost allocation rule must coincide with the Bird Rule on the set of cost matrices which give rise to unique MCA s.

For any arborescence $g \in A_N$, for any $i \in N$, let $\rho(i)$ denote the predecessor of i in g. That is, $\rho(i)$ is the unique node which comes just before i in the path connecting i to the source 0.

DEFINITION **14** *Let C be some cost matrix.*

(i) A Bird allocation of any MCA g is $b_i(g, C) = c_{\rho(i)i}$ for all $i \in N$.

(ii) A Bird rule is given by $B_i(C) = \sum_{g \in M(C)} w_g(C) b_i(g, C)$ for all $i \in N$, where $\sum_{g \in M(C)} w_g(C) = 1$ and $w_g(C) \geq 0$ for each $g \in M(C)$.

Remark 3 *Notice that the Bird rule is a family of rules since it is possible to have different convex combinations of Bird allocations. The set of weights need to be chosen consistently if the Bird Rule is to satisfy IIC. In particular, suppose $M(C) = M(C')$ for two cost matrices C and C'. Then, the restriction that $w_g(C) = w_g(C')$ for each $g \in M(C)$ ensures that the resulting Bird Rule satisfies IIC.*

We first prove that Bird allocations belong to the core of the cost game.

THEOREM **1** *For every cost matrix C and MCA $g \in M(C)$, $b(g, C) \in Co(N, c)$.*

Proof: Consider any $g \in M(C)$. Assume that $b(g, C)$ is not in the core and some $S \subseteq N$ is a blocking coalition. This implies that

$$\sum_{i \in S} b_i(g, C) > c(S). \tag{1}$$

Let E^N be the set of edges used by the MCA g. For every $i \in N$, denote by e_i the unique edge incident on node i in g, and let $E^N_S = \{e_i : i \in S\}$. Now, consider an MCA of coalition S corresponding to cost matrix C, and let E^S be the set of edges used by this MCA. Consider the digraph $g' = (E^N \setminus E^N_S) \cup E^S$. This digraph must be an arborescence for the grand coalition. To see this, note that every $i \in N$ has only one incoming edge in g' - for every agent i, if we have removed the unique incoming edge in g, we have replaced it with a unique edge from E^S. Next, g' cannot have a cycle since E^S is the set of edges in the MCA for S, and this implies that every node in N is connected to the source 0.

Now, the cost of the arborescence g' is

$$c(N) - \sum_{i \in S} b_i(g, C) + c(S) < c(N),$$

where the inequality comes from Inequality (1). This contradicts the fact g is an MCA of the grand coalition. ■

Of course, the Bird Rule satisfies IIC subject to the restriction mentioned in Remark 3. We now prove a partial converse by showing that any cost allocation rule satisfying CS and IIC must specify cost shares which are bounded by the minimum Bird allocation. That is, for each C and each $i \in N$, let

$$b_i^m(C) = \min_{g \in M(C)} b_i(g, C)$$

Then, we have the following theorem.

THEOREM **2** *For every cost matrix C and every cost allocation rule μ satisfying CS and IIC, we have*

$$\mu_i(C) \geq b_i^m(C) \qquad \forall\ i \in N.$$

Moreover, if $M(C)$ is a singleton, then μ coincides with the unique Bird Rule.

Proof: Fix a cost matrix C, and consider any cost allocation rule μ which satisfies CS and IIC. Assume for contradiction that there is $i \in N$ such that $\mu_i(C) = b_i^m(C) - \epsilon$, where $\epsilon > 0$. Let g be an MCA such that $b_i(g, C) = b_i^m(C)$.

Call node p a *successor* of node q in g if the edge $qp \in g$. Let S be the set of all successors of i, and $T = N \setminus \{i\}$. Let the edge $ki \in g$. Note that

$$\sum_{j \in T} \mu_j(C) = c(N) - c_{ki} + \epsilon \tag{2}$$

Note that if $S = \emptyset$, then $c(T) = c(N) - c_{ki} < \sum_{j \in T} \mu_j(C)$. Hence, T is a blocking coalition, contradicting the fact that μ satisfies CS.

So, $S \neq \emptyset$. Define

$$E(C) \equiv \bigcup_{g'' \in M(C)} \{pq : pq \in g''\}.$$

That is, $E(C)$ is the set of all directed edges that belong to *some* MCA corresponding to the cost matrix C.

Let $S_1 = \{j \in S : kj \in E(C)\}$, and $S_2 = S \setminus S_1$. Suppose $j \in S_1$. Let g' be some MCA in which $kj \in g'$. Since $(g' \setminus \{kj\}) \cup \{ij\}$ is also an arborescence, we have $c_{kj} \leq c_{ij}$. Similarly, $c_{ij} \leq c_{kj}$ since g is an MCA. Hence, for all $j \in S_1$,

$$c_{ij} = c_{kj}$$

Notice that if $S_2 = \emptyset$, then the previous equality implies that $c(T) = c(N) - c_{ki} < \sum_{j \in T} \mu_j(C)$, and so T would be a blocking coalition. So, S_2 must be nonempty.

Now, consider the digraph $\bar{g} = g \setminus (\{ki\} \cup \{ij : j \in S\}) \cup \{kj : j \in S\}$. That is, $\bar{g}$ is the digraph in which all edges involving i are deleted from g and all successors of i in g become successors of k in $\bar{g}$. Then, $\bar{g}$ is an arborescence for T. Consider another cost matrix C' constructed as follows.

$$\begin{aligned}
c'_{kj} &= c_{ij} + \frac{\epsilon}{2|S_2|} && \text{if } j \in S_2 \\
c'_{pq} &= c_{pq} && \forall\, pq \in E(C) \\
c'_{pq} &= c(N) + 1 && \text{for all other } pq.
\end{aligned}$$

We first show that C and C' are arborescence equivalent. To see this, note that edges not in $(E(C) \cup \{kj : j \in S_2\})$ cannot be part of any MCA corresponding to C'. Now, assume for contradiction that kj^* belongs to some $\tilde{g} \in M(C')$ where $j^* \in S_2$. Let k^*i be an edge in $\tilde{g}$. Since we have assumed that $c_{ki} = b_i^m(C)$, and since $c'_{ki} = c_{ki}$, we have

$$c'_{ki} \leq c'_{k^*i}$$

Consider the digraph $\hat{g} \equiv (\tilde{g} \setminus \{k^*i, kj^*\}) \cup \{ki, ij^*\}$ (i.e., remove edges k^*i and kj^* from $\tilde{g}$ and insert edges ki and ij^*). Note that every $j \in N$ has a unique incoming edge in $\hat{g}$. Further, $\hat{g}$ cannot have a cycle since having a cycle in $\hat{g}$ must imply that it must include ij^* and/or ki, which in turn implies that there is a path from j^* to k in $\tilde{g}$. But this is not possible since this creates a cycle in $\tilde{g}$. Hence, $\hat{g}$ is an arborescence. But, since $c'_{ij^*} < c'_{kj^*}$ and $c'_{ki} \leq c'_{k^*i}$, the total cost of arborescence $\hat{g}$ is lower than that of $\tilde{g}$, which is a contradiction since $\tilde{g}$ is in $M(C')$.

So, C and C' are arborescence equivalent. Then, IIC implies that $\mu(C) = \mu(C')$.

Now, we get the cost of arborescence $\bar{g}$ in cost matrix C' as

$$\begin{aligned}
\sum_{pq \in \bar{g}} c'_{pq} &= c(N) - c_{ki} - \sum_{j \in S} c_{ij} + \sum_{j \in S} c'_{kj} \\
&= c(N) - c_{ki} - \sum_{j \in S_2} c_{ij} + \sum_{j \in S_2} [c_{ij} + \frac{\epsilon}{2|S_2|}] \\
&= c(N) - c_{ki} + \frac{\epsilon}{2} \\
&= \sum_{j \in T} \mu_j(C') - \frac{\epsilon}{2},
\end{aligned}$$

where the last inequality followed from $\mu(C) = \mu(C')$ and Equation 2. Hence, $c'(T) < \sum_{j\in T} \mu_j(C')$, which contradicts the fact that μ satisfies CS.

Hence, $\mu_j(C) \geq b^m_j(C)$ for all $j \in N$. It follows that if there is a unique MCA corresponding a cost matrix C, then $\mu_i(C) = b^m_i(C)$ for all $i \in N$ since $\sum_{i\in N} \mu_i(C) = \sum_{i\in N} b^m_i(C) = c(N)$. ■

Remark 4 *It is interesting to see why there is no corresponding result for the mcst problem. Let* $N = \{1, 2\}$*, and consider the mcst problem where* $c_{01} = 2$*,* $c_{02} = 3$*,* $c_{12} = 1$*. The "folk" solution for mcst problems then specifies that both players pay* 1.5*, while the Bird solution is that player 1 pays 2 and player 2 pays 1. Since the "folk" solution is reductionist (that is, it satisfies IIC) and satisfies CS, this shows that Theorem 2 does not hold for the mcst problem. Now, consider the MCA problem where* $c_{01} = 2$*,* $c_{02} = 3$*,* $c_{12} = 1$*, and* $c_{21} = a > 0$*. Then, the unique MCA is* $g = \{01, 12\}$*. If the solution is to satisfy IIC, then it cannot depend on the value of* a *since the edge* 21 *is an "irrelevant" edge. This is an additional constraint in the MCA problem, and one which does not arise in the mcst problem because symmetry of the cost matrix implies that* $a = 1$*. The additional constraint restricts the class of admissible solutions.*

The following related theorem is of independent interest. We show that no *reductionist* solution satisfying CS can satisfy either CON or DCM. As we have pointed out in the introduction, this result highlights an important difference between the solution concepts for the classes of symmetric and asymmetric cost matrices.

Theorem **3** *Suppose a cost allocation rule satisfies CS and IIC. Then, it cannot satisfy either CON or DCM.*

Proof: Let $N = \{1, 2\}$. Consider the cost matrix given below

$$c_{01} = 6, c_{02} = 4, c_{12} = 1, c_{21} = 3.$$

Then,

$$b^m_1(C) = 3, b^m_2(C) = 1.$$

Let μ satisfy IIC and CS. Then, by Theorem 2

$$\mu_1(C) \geq 3, \mu_2(C) \geq 1.$$

Now, consider C' such that $c'_{01} = 6 + \epsilon$ where $\epsilon > 0$, and all other edges cost the same as in C. There is now a unique MCA, and so since μ satisfies CS and IIC, by Theorem 2

$$\mu(C') = (3, 4).$$

If μ also satisfies DCM, then we need $3 = \mu_1(C') \geq \mu_1(C) \geq 3$. This implies

$$\mu(C) = (3, 4).$$

Now consider C'' such that $c''_{02} = 4 + \gamma$, with $\gamma > 0$ while all other edges cost the same as in C. It follows that if μ is to satisfy CS, IIC, and DCM, then

$$\mu(C) = (6, 1).$$

This contradiction establishes that there is no μ satisfying CS, IIC, and DCM.

Now, if μ is to satisfy CS, IIC and CON, then there must be a continuous function $f : \Re^2 \to \Re^2$ such that

$$(3, 4) = \mu(C) + f(\epsilon, 0), \text{ and } (6, 1) = \mu(C) + f(0, \gamma)$$

for all $\epsilon, \gamma > 0$ and with $f(0, 0) = (0, 0)$. Clearly, no such continuous function can exist. ■

The last theorem and the earlier remark demonstrate that there is a sharp difference between allocation rules when the cost matrix is symmetric and when it is asymmetric. The literature on minimum cost spanning tree games shows that there are a large number of cost allocation rules satisfying CS, IIC , some appropriate analogue of DCM and\or CON. Clearly, options are more limited when the cost matrix is asymmetric. Nevertheless, we show in the next section that it is possible to construct a rule satisfying the three "basic" properties of CS, DCM and CON. Of course, the rule we construct will not satisfy IIC.

4 A Rule Satisfying CS, DSCM and CON

In this section, we construct a rule satisfying the three "basic" axioms of CS, DSCM and CON. Our construction will use a method which has been used to construct the "folk solution". The rule satisfies counterparts of the three basic axioms in the context of the minimum cost spanning tree framework. However, the rule constructed by us will be quite different. In particular, the "folk solution" satisfies the counterpart of IIC but does not satisfy R for symmetric cost matrices. In contrast, while our rule obviously cannot satisfy IIC, we will show that it satisfies R.

We first describe a recursive algorithm due to Chu and Liu (1965) and Edmonds (1967) to construct an MCA. This algorithm will play a crucial role in the construction of our solution. Although the algorithm is quite different from the algorithms for constructing an MCST, it is still computationally tractable as it runs in *polynomial time.*

4.1 The Recursive Algorithm

It turns out that the typical greedy algorithms used to construct minimum cost spanning trees fail to generate minimum cost arborescences. Figure 1 illustrates this phenomenon.[10] Recall that a unique feature of minimum cost spanning trees is that an MCST must always choose the minimum cost (undirected) edge corresponding to any cost matrix. Notice, however, that in Figure 1, the minimum cost arborescence involves edges $01, 12, 23$. But it does not involve the minimum cost edge 31.

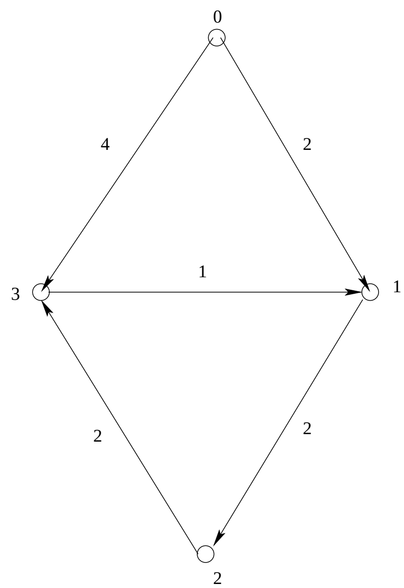

Figure 1: An example where greedy algorithm for MCST problems fails

The recursive algorithm works as follows. In each recursion stage, the original cost matrix on the original set of nodes and the original graph is transformed to a new cost matrix on a new set of nodes and a new graph. The terminal stage of the recursion yields an MCA for the terminal cost matrix and the terminal set of nodes. One can then "go back" through the recursion stages to get an MCA for the original problem. Since the algorithm to compute an MCA is different from the algorithm to compute an MCST, we first describe it in detail with an example.

Consider the example in Figure 2. To compute the MCA corresponding to the cost matrix in Figure 2, we first perform the following operation: for every node, subtract the value of the minimum cost incident edge from the cost of every edge incident on that node. As an example, 31 is the minimum cost incident edge on node 1 with cost 1. Hence, the new cost

[10]The numbers besides each edge represent the cost of the edge. The missing edges have very high cost and do not figure in any MCA.

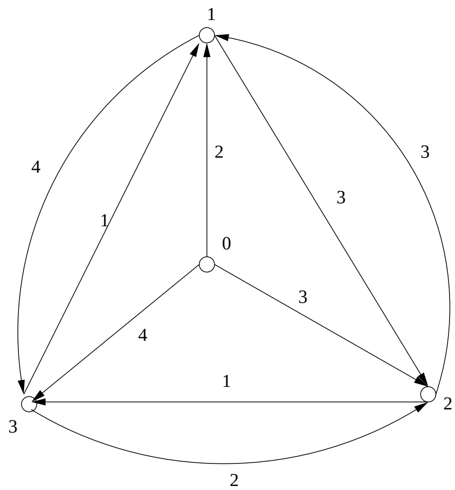

Figure 2: An example

of edge 31 is $1-1=0$, edge 01 is $2-1=1$, and edge 21 is $3-1=2$. In a similar fashion, compute the following new cost matrix C^1:

$$c^1_{01}=1,\ c^1_{21}=2,\ c^1_{31}=0$$
$$c^1_{02}=1,\ c^1_{12}=1,\ c^1_{32}=0$$
$$c^1_{03}=3,\ c^1_{23}=0,\ c^1_{13}=3.$$

Clearly, an MCA corresponding to C^1 is also an MCA corresponding to the original cost matrix. So, we find an MCA corresponding to C^1. To do so, for every node, we pick a zero cost edge incident on it (by the construction of C^1, there is at least one such edge for every node). If such a set of edges form an arborescence, it is obviously an MCA corresponding to C^1, and hence, corresponding to the original cost matrix. Otherwise, cycles are formed by such a set of zero cost edges. In the example, we see that the set of minimum cost edges are 31, 32, and 23. So, 32 and 23 form a cycle. The algorithm then merges nodes 2 and 3 to a single *supernode* (23), and constructs a new graph on the set of nodes 0, 1, and supernode (23). We associate a new cost matrix $\widetilde{C}^1$ on this set of nodes using C^1 as follows: $\tilde{c}^1_{01}=c^1_{01}=1$; $\tilde{c}^1_{0(23)}=\min\{c^1_{02},c^1_{03}\}=\min\{1,3\}=1$; $\tilde{c}^1_{1(23)}=\min\{c^1_{12},c^1_{13}\}=\min\{1,3\}=1$, $\tilde{c}^1_{(23)1}=\min\{c^1_{21},c^1_{31}\}=\min\{2,0\}=0$. The graph consisting of these edges along with the costs corresponding to cost matrix $\widetilde{C}^1$ is shown in Figure 3.

We now seek an MCA for the graph depicted in Figure 3. We repeat the previous step. The minimum cost incident edge on 1 is (23)1 and we choose the minimum cost incident edge on (23) to be 0(23). Subtracting the minimum costs as we did earlier, we get that 0(23) and (23)1 are edges with zero cost. Since these edges form an arborescence, this is an MCA corresponding to cost matrix $\widetilde{C}^1$. To get the MCA for the original cost matrix, we note

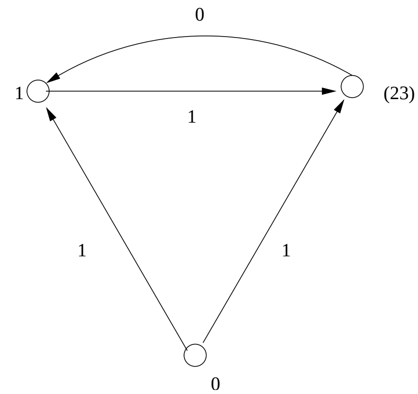

Figure 3: After first stage of the algorithm

that $\tilde{c}^1_{0(23)} = c^1_{02}$. Hence, 0(23) is replaced by edge 02. Similarly, (23)1 is replaced by edge 31. The cycle in supernode (23) is broken such that we get an arborescence - this can be done by choosing edge 23 since 02 is the incident edge on supernode (23). Hence, the MCA corresponding to the original cost matrix is: 02, 23, 31.

We now describe the algorithm formally. We will label the digraph constructed by picking one minimum cost incoming edge to every node in N a *greedy digraph* on N.

DEFINITION **15** *A cost matrix C is a simple cost matrix if there is a corresponding greedy digraph which forms an arborescence.*

Notice that if C is a simple matrix, then any greedy digraph is an MCA. Also, we will use the notion of greedy digraphs and simple matrices for arbitrary sets N'.

Given any cost matrix C, we will say that a set of nodes $I = \{1, \dots, K\}$ form a C-*cycle* if $c_{ii+1} = 0$ for all $i = 1, \dots K-1$ and $c_{K1} = 0$.

- *Stage* 0: Set $C^0 \equiv \widetilde{C}^0 \equiv C$, $N^0 \equiv N \cup \{0\}$, and for each $j \in N$,

$$\Delta^0_j = \min_{i \in N^+ \setminus \{j\}} c^0_{ij}, N^0_j = \{j\}.$$

- *Stage* 1: For each pair $i \in N^0$ and $j \in N$, define $c^1_{ij} = c^0_{ij} - \Delta^0_j$. Construct a partition $\{N^1_1, \dots, N^1_{K^1}\}$ of N^0 such that each N^1_k is either a C^1-cycle of elements of N^0 or a singleton with the restriction that no set of singletons forms a C^1-cycle.[11] Note that since 0 cannot be part of any cycle, 0 is one of the singletons in this partition. Let $N^1_1 \equiv 0$. Denote $N^1 = \{N^1_1, \dots, N^1_{K^1}\}$. For each $k, l \in \{1, \dots, K^1\}$, with $l \neq 1$, define

$$\tilde{c}^1_{N^1_k N^1_l} = \min_{i \in N^1_k, j \in N^1_l} c^1_{ij} = \min_{i \in N^1_k, j \in N^1_l} [\tilde{c}^0_{ij} - \Delta^0_j].$$

[11]If there is some node $j \in N$ such that two or more edges minimize cost, then break ties arbitrarily.

Hence, $\widetilde{C}^1$ is a cost matrix on nodes N^1. For each $k \in \{2, \ldots, K^1\}$, define

$$\Delta_k^1 = \min_{i \notin N_k^1, j \in N_k^1} c_{ij}^1 = \min_{N_l^1 \neq N_k^1} \tilde{c}^1_{N_l^1 N_k^1}.$$

- *Stage t*: For each pair i, j, define $c_{ij}^t = c_{ij}^{t-1} - \Delta_j^{t-1}$. Construct a partition $\{N_1^t, \ldots, N_{K^t}^t\}$ of N^{t-1} such that each N_k^t is either a C^t-cycle of elements of N^{t-1} or a singleton element of N^{t-1} with the restriction that no set of singletons forms a C^t-cycle. Denote $N_1^t \equiv 0$. For each $k, l \in \{1, \ldots, K^t\}$, with $l \neq 1$, define

$$\tilde{c}^t_{N_k^t, N_l^t} = \min_{i \in N_k^t, j \in N_l^t} c_{ij}^t = \min_{N_p^{t-1} \in N_k^t, N_q^{t-1} \in N_l^t} [\tilde{c}^{t-1}_{N_p^{t-1} N_q^{t-1}} - \Delta_q^{t-1}]. \tag{3}$$

  Note that $\widetilde{C}^t$ is a cost matrix on nodes N^t.

  For each $k \in \{2, \ldots, K^t\}$, define

$$\Delta_k^t = \min_{i \notin N_k^t, j \in N_k^t} c_{ij}^t = \min_{N_l^t \neq N_k^t} \tilde{c}^t_{N_l^t N_k^t}.$$

Terminate the algorithm at stage T if $\tilde{C}^T$ is a simple cost matrix on N^T. Since the source cannot be part of any cycle and since N is finite, the algorithm must terminate.

We will sometimes refer to sets of nodes such as N_k^t as *supernodes*. The algorithm proceeds as follows. First, construct a greedy digraph g^T on N^T corresponding to cost matrix $\widetilde{C}^T$. Since $\tilde{C}^T$ is a simple matrix on N^T, g^T must be an MCA for N^T.

Then, unless $T = 0$, "extend" $g^t \in M(\tilde{C}^t)$ to $g^{t-1} \in M(\tilde{C}^{t-1})$ for all $1 \leq t \leq T$ by establishing connections between the elements of N^t and N^{t-1}, until we reach $g^0 \in M(C)$.

If $T = 0$, g^0 is an MCA on N^0 corresponding to cost matrix $\tilde{C}^0$ (and hence C) since g^T is an MCA on N^T corresponding to cost matrix $\widetilde{C}^T$.

Suppose $T \geq 1$. For every $1 \leq t \leq T$, given an MCA g^t corresponding to $\widetilde{C}^t$, we extend it to an MCA g^{t-1} on nodes N^{t-1} corresponding to $\widetilde{C}^{t-1}$ as follows.

(i) For every edge $N_i^t N_j^t$ in g^t, let $\tilde{c}^t_{N_i^t, N_j^t} = \tilde{c}^{t-1}_{N_p^{t-1} N_q^{t-1}} - \Delta_q^{t-1}$ for some $N_p^{t-1} \subseteq N_i^t$ and some $N_q^{t-1} \subseteq N_j^t$. Then, replace $N_i^t N_j^t$ with $N_p^{t-1} N_q^{t-1}$ in graph g^{t-1}.

(ii) Consider every edge $N_p^{t-1} N_q^{t-1}$ added to g^{t-1} in Step (i). Suppose $N_p^{t-1} N_q^{t-1} \in g^{t-1}$ and $N_q^{t-1} \subseteq N_t^k$. Further, suppose N_k^t forms a C^{t-1} cycle with edges

$$H = \{N_{k_1}^{t-1} N_{k_2}^{t-1}, N_{k_2}^{t-1} N_{k_3}^{t-1}, \ldots, N_{k_h}^{t-1} N_{k_1}^{t-1}\},$$

where $k_h = q$. Then, assign to g^{t-1} all edges in H except edge $N_{k_{h-1}}^{t-1} N_{k_h}^{t-1}$.

This completes the extension of g^{t-1} from g^t. It is not difficult to see that g^{t-1} is an MCA for graph with nodes N^{t-1} corresponding to $\widetilde{C}^{t-1}$ (for a formal argument, see Edmonds (1967)). Proceed in this way to $g^0 \in M(C)$.

4.2 Irreducible Cost Matrices

We first briefly describe the methodology underlying the construction of the "folk solution".

Bird (1976) defined the concept of an *irreducible* cost matrix corresponding to any cost matrix C in the minimum cost spanning tree problem. Given any cost matrix C, Bird's irreducible cost matrix has the property that the cost of no edge can be reduced any further if the MCST corresponding to the original matrix C is to remain an MCST of the modified matrix. The irreducible cost matrix C^{BR} is obtained from a symmetric cost matrix C in the following way. Let g be some minimum cost spanning tree for a symmetric cost matrix C.[12] For any $i \in N^+$ and $j \in N$, let $p(ij)$ denote the path from i to j for this tree. Then, the "irreducible" cost of the edge ij is

$$c_{ij}^{BR} = \max_{kl \in p(i,j)} c_{kl} \text{ for all } i, j \in N. \tag{4}$$

Of course, if some edge ij is part of a minimum cost spanning tree, then $c_{ij}^{BR} = c_{ij}$. If some edge ij is not part of any minimum cost spanning tree, then $c_{ij}^{BR} < c_{ij}$. So, while every original minimum cost spanning tree remains a minimum cost spanning tree for the irreducible cost matrix, new trees also minimize the irreducible cost of a spanning tree.

Bird (1976) showed that the cost game corresponding to C^{BR} is *concave*. From the well-known theorem of Shapley (1971), it follows that the Shapley value belongs to the core of this game. Moreover, since $c_{ij}^{BR} \leq c_{ij}$ for all edges ij, it follows that the core of the game corresponding to C^R is contained in the core of the game corresponding to C. So, the cost allocation rule choosing the Shapley value of the game corresponding to C^{BR} satisfies CS. Indeed, it also satisfies Cost Monotonicity[13] and Continuity.

It is natural to try out the same approach for the minimum cost arborescence problem. However, an identical approach cannot possibly work when the cost matrix is asymmetric. Notice that the construction of the irreducible cost matrix outlined above only uses information about the costs of edges belonging to some minimum cost spanning tree. So, the Shapley value of the cost game corresponding to the irreducible cost matrix must also depend only on such information. In other words, the "folk" solution must satisfy IIC. It follows from Theorem 3, that no close cousin of the folk solution can satisfy the desired properties.

A possible explanation for why a somewhat different approach is required is provided by the following example.

Example 1 *Let* $N = \{1, 2\}$, $c_{01} = 6, c_{02} = 5, c_{12} = 4, c_{21} = 8$. *Then, the unique MCA is*

[12] If more than one tree minimises total cost, it does not matter which tree is chosen.

[13] Bergantinos and Vidal-Puga (2007a) show that it satisfies a stronger version of cost monotonicity - a *solidarity* condition which requires that if the cost of some edge goes up, then the cost shares of all agents should (weakly) go up.

$g = \{01, 12\}$. *But, notice that*

$$\max_{kl \in p(0,2)} c_{kl} \equiv c_{01} = 6 > c_{02}$$

In what follows, we focus on the essential property of an irreducible cost matrix - it is a cost matrix which has the property that the cost of no edge can be reduced any further if an MCA for the original matrix is to remain an MCA for the irreducible matrix and the total cost of an MCA of the irreducible cost matrix is the same as that of the original matrix. We say that two cost matrices C and C' are *cost equivalent* if $T(C) = T(C')$; that is two cost matrices are cost equivalent if the total cost of their minimum cost arborescences are equal.

DEFINITION **16** *A cost matrix C^R is an irreducible cost matrix (ICM) of cost matrix C if C and C^R are cost equivalent, $c^R_{ij} \leq c_{ij}$ for all (ij), and there does not exist another cost matrix C' which is cost equivalent to C such that $c'_{ij} < c^R_{ij}$ for some (ij) and $c'_{kl} \leq c^R_{kl}$ for all $(kl) \neq (ij)$.*

Remark 5 *An equivalent definition of C^R is that C and C^R are cost equivalent, $c^R_{ij} \leq c_{ij}$ for all (ij), and every edge belongs to some MCA corresponding to C^R.*

Of course, it is possible that C is an ICM of itself - this happens when no edge cost can be reduced without decreasing the total cost of the MCA. In this case, we will call C an ICM.

There are other important differences between the mcst and MCA problems. For instance, there may be more than one corresponding irreducible cost matrix *even when* the cost matrix C is symmetric. Moreover, the multiple ICMs in the MCA problem may also be asymmetric.[14]

EXAMPLE **2** *Let $N = \{1, 2\}$, $c_{01} = 1, c_{02} = 3$, $c_{12} = c_{21} = 2$. Then, the entire class of ICMs is given by*

$$c^R_{01} = 1, c^R_{02} = 2 + \epsilon, c^R_{12} = 2, c_{21} = 1 - \epsilon, \text{ where } \epsilon \in [0, 1].$$

Let the correspondence $R(C)$ represent the set of irreducible cost matrices corresponding to each cost matrix C. The following is obvious.

Fact: If C is an ICM, then $R(C) = \{C\}$.

Our procedure involves the following. Given any cost matrix C, we use the recursive algorithm to construct a *continuous* and single-valued selection of $R(C)$. *With some abuse of notation, we will denote the (unique) ICM constructed by us as C^R.* However, this will not cause any confusion since we restrict attention to this ICM in the rest of the paper.

We then go on to show that the ICM C^R constructed by us satisfies the following:

[14]We are indebted to Anirban Kar for this observation.

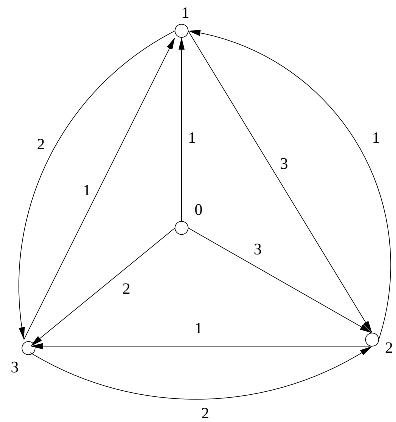

Figure 4: Irreducible cost matrix for cost matrix in Figure 2

1. The irreducible cost matrix C^R is well-defined in the sense that it does not depend on the tie-breaking rule used in the recursive algorithm.

2. The cost game corresponding to C^R is concave.

We will use these properties to show that the rule choosing the Shapley value of the cost game corresponding to C^R satisfies CS, DSCM, and CON.[15]

We first illustrate the construction of the irreducible cost matrix for the example in Figure 2. We do it recursively. So, we first construct an irreducible cost matrix on nodes N^T and corresponding cost matrix $\widetilde{C}^T$ of the last stage T of the algorithm. Figure 3 exhibits the graph on this set of nodes for the example in Figure 2. An irreducible cost matrix corresponding to this cost matrix can be obtained in the following manner. Set the cost of any edge ij equal to the minimum cost incident edge on j. Denoting this reduced cost matrix as $\widetilde{C}^R$, we get $\tilde{c}^R_{01} = \tilde{c}^R_{(23)1} = 0$ and $\tilde{c}^R_{0(23)} = \tilde{c}^R_{1(23)} = 1$. It is easy to verify that this is indeed an irreducible cost matrix. Now, we extend $\widetilde{C}^R$ to an irreducible cost matrix C^R corresponding to the original cost matrix. For any edge ij, if i and j belong to different supernodes N^1_i and N^1_j respectively in Figure 3, then $c^R_{ij} = \tilde{c}^R_{N^1_i N^1_j} + \Delta^0_j$. For example, $c^R_{12} = \tilde{c}^R_{1(23)} + 2 = 1 + 2 = 3$ and $c^R_{03} = \tilde{c}^R_{0(23)} + 1 = 1 + 1 = 2$. For any ij, if i and j belong to the same supernode, then $c^R_{ij} = \Delta^0_j$. For example, $c^R_{23} = 1$ and $c^R_{32} = 2$. Using this, we show the irreducible cost matrix in Figure 4.

We now formalize these ideas of constructing an irreducible cost matrix below.

Fix some cost matrix C. Suppose that, given some tie-breaking rule, the recursive algorithm terminates in T steps. If $T > 0$, then for every $i \in N^+$ and $j \in N^+ \setminus \{i\}$, we say i and j are t-siblings if $i, j \in N^t_k$ for some $N^t_k \in N^t$, and there is no $t' < t$ such that $i, j \in N^{t'}_l$ for some $N^{t'}_l \in N^{t'}$. For every $i \in N^+$ and $j \in N^+ \setminus \{i\}$, if i and j are not t-siblings for any

[15]We show later that this rule also satisfies INV, S and R.

$t \in \{0, \ldots, T\}$, then they are $(T+1)$-siblings. Note that if $T = 0$, then for every $i \in N^+$ and $j \in N^+ \setminus \{i\}$, i and j are 1-siblings. Also, 0 and i will be $(T+1)$-siblings for all $i \in N$.

For every $t \in \{0, \ldots, T\}$, and every $i \in N$,

$$\delta_i^t = \Delta_k^t, \text{ where } i \in N_k^t.$$

Now, for every cost matrix C on nodes N^+, define the cost matrix C^R as follows. For every $i \in N^+$ and $j \in N \setminus \{i\}$,

$$c_{ij}^R := \sum_{t'=0}^{t-1} \delta_j^{t'}, \tag{5}$$

where i and j are t-siblings. So, for instance, if i and j are 1-siblings then $c_{ij}^R = \Delta_j^0$. We will prove subsequently that C^R is an ICM of C.

Since the recursive algorithm breaks ties arbitrarily and the irreducible cost matrix uses the recursive algorithm, it does not follow straightaway that two different tie-breaking rules result in the same irreducible cost matrix. We say that the irreducible cost matrix is *well-defined* if different tie-breaking rules yield the same irreducible cost matrix.

LEMMA **1** *The cost matrix defined through Equation (5) is well-defined.*

Proof: To simplify notation, suppose there is a tie in Stage 0 of the algorithm, between exactly two edges say ij and kj, as the minimum incident cost edge on node j.[16] Hence, $c_{ij} = c_{kj} = \Delta_j^0$. We will investigate the consequence of breaking this tie one way or the other. For this, we break the other ties in the algorithm exactly the same way in both cases.

We distinguish between three possible cases.

CASE 1: Suppose j forms a singleton node N_j^1 in step 1 irrespective of whether the algorithm breaks ties in favour of ij or kj. Then, in either case $\Delta_j^1 = 0$. Moreover, the structure of N^t for all subsequent t is not affected by the tie-breaking rule. Hence, C^R must be well-defined in this case.

CASE 2: There are two possible C^1-cycles - N_i^1 and N_k^1 depending on how the tie is broken. Suppose the tie is broken in favour of ij so that N_i^1 forms a supernode. Then, $\{k\}$ forms a singleton node in Stage 1. But, since $c_{kj}^1 = 0$, $N_i^1 \cup N_k^1$ will form a supernode in step 2 of the algorithm. Notice that if the tie in step 1 was resolved in favour of kj, then again $N_i^1 \cup N_k^1$ would have formed a supernode in step 2 of the algorithm.

Also, the minimum cost incident edges of nodes outside N_i^1 and N_k^1 remain the same whether we break ties in favor of ij or kj. Hence, we get the same stages of the algorithm

[16]The argument can easily be extended to a tie at any step t of the algorithm and to ties between any number of edges.

from stage 2 onwards. So, if $r \notin N_i^1 \cup N_k^1$, then for all $s \in N^+ \setminus \{r\}$, c_{sr}^R cannot be influenced by the tie-breaking rule.

Let $s \in N_i^1 \cup N_k^1$. Then, irrespective of how the tie is broken, $\delta_s^1 = 0$, and hence c_{rs}^R cannot be influenced by the tie-breaking rule for any r.

CASE 3: Suppose N_i^1 is a C^1-cycle containing i and j if we break tie in favor of ij, but there is no C^1-cycle involving k if we break tie in favor of kj. Since the minimum incident edge on N_i^1 corresponds to kj and $c_{kj}^1 = 0$, $\delta_p^1 = 0$ for all $p \in N_i^1$ irrespective of how the tie is broken. So, for all $s \in N_i^1$, c_{rs}^R is unaffected by the tie-breaking rule for all r.

These arguments prove that Equation 5 gives a well-defined cost matrix. ■

In the next lemma, we want to show that the total cost of the minimum cost arborescences corresponding to cost matrices C and C^R is the same.

The proof of this result involves a similar construction which we now describe. In particular, we associate a cost matrix to *each* stage of the algorithm. Apart from being used in the proof of this result, these cost matrices also give an alternate interpretation of the irreducible cost matrix.

Let C be any cost matrix and C^R be the cost matrix defined through Equation 5 corresponding to C.

Consider stage t of the recursive algorithm. The set of nodes in stage t is N^t. Consider $N_i^t, N_j^t \in N^t$ and let them be $\hat{t}$-siblings, i.e., in some stage $\hat{t} \in \{t+1, \ldots, T-1\}$ nodes N_i^t and N_j^t become part of the same supernode for the first time. Now, define

$$\widehat{c}_{N_i^t N_j^t} = \sum_{t'=t}^{\hat{t}-1} \delta_{N_j^t}^{t'}.$$

This defines a cost matrix $\widehat{C}^t$ on nodes N^t for stage t of the algorithm. Note that $C^R = \widehat{C}^0$. From the algorithm, the total cost of an MCA with nodes N corresponding to cost matrix C^t is equal to the total cost of an MCA with nodes N^t corresponding to cost matrix $\widetilde{C}^t$. (See Equation 3). Moreover, $\widehat{C}^t$ is the cost matrix defined through Equation 5 for $\widetilde{C}^t$.

LEMMA **2** *Suppose C^R is the cost matrix constructed through Equation 5 corresponding to cost matrix C. Then, $T(C) = T(C^R)$.*

Proof: We prove the result by using induction on the number of stages of the algorithm. If $T = 0$, then $c_{ij}^R = \Delta_j^0$ for all edges ij. Since C is a simple cost matrix, $T(C) = \sum_{j \in N} \Delta_j^0 = T(C^R)$.

Now, assume the lemma holds for any cost matrix that takes less than t stages where $t > 0$. We show that the lemma holds for any cost matrix C that takes t stages. Consider cost matrix $\widehat{C}^1$, which is the cost matrix defined through Equation 5 for $\widetilde{C}^1$. Note that the algorithm takes $t-1$ stages for cost matrix $\widetilde{C}^1$, applied to nodes in N^1. Hence, by our induction hypothesis and using the fact $T(C^1) = T(\widetilde{C}^1)$, we get

$$T(C^1) = T(\widetilde{C}^1) = T(\widehat{C}^1). \tag{6}$$

Now, consider the cost matrix $\bar{C}$ defined as follows: $\bar{c}_{ij} = 0$ if $i, j \in N_k^1$ for some supernode N_k^1 in Stage 1, and $\bar{c}_{ij} = \hat{c}^1_{N_i^1 N_j^1}$ otherwise. Thus, $\bar{C}$ is a cost matrix on N^0 with edges in supernodes in stage 1 having zero cost and other edges having the same cost as in cost matrix $\widehat{C}^1$. Clearly, $T(\widehat{C}^1) = T(\bar{C})$ (since edges inside a supernode in stage 1 have zero cost in $\bar{C}$). Using Equation 6,

$$T(C^1) = T(\bar{C}). \tag{7}$$

But cost matrices $\bar{C}$ and C^R differ as follows: for any edge ij in the original graph

$$\bar{c}_{ij} + \Delta_j^0 = c_{ij}^R.$$

This implies that

$$T(C^R) = T(\bar{C}) + \sum_{j \in N} \Delta_j^0. \tag{8}$$

Similarly, C^1 and C differ as follows: for any edge ij

$$c_{ij}^1 + \Delta_j^0 = c_{ij}.$$

This implies that

$$T(C) = T(C^1) + \sum_{j \in N} \Delta_j^0. \tag{9}$$

Using Equations 7, 8, and 9,

$$T(C) = T(C^1) + \sum_{j \in N} \Delta_j^0 = T(\bar{C}) + \sum_{j \in N} \Delta_j^0 = T(C^R).$$

This completes the proof. ■

Lemma 3 *The cost matrix C^R constructed via Equation 5 is an ICM.*

Proof: By Lemma 1, C^R is well-defined irrespective of the tie-breaking rule we use. So, fix some tie-breaking rule. By Lemma 2, $T(C) = T(C^R)$. By construction, $c_{ij}^R \leq c_{ij}$ for all edges ij. Hence, it is sufficient to prove that for every $i \in N^+$ and every $j \in N \setminus \{i\}$, the edge ij belongs to some $g \in M(C^R)$.

We prove this by induction on the number of stages T of the algorithm. If $T = 0$, then C is a simple matrix. Then, for every edge ij, $c^R_{ij} = \min_{k \neq j} c_{kj}$. Hence, *every* arborescence g belongs to $M(C^R)$. Thus, every edge in the graph belongs to some MCA $g \in M(C^R)$.

Suppose the lemma is true for any $T < t$, and let $T = t$. Let $\{N^1_1, N^1_2, \ldots, N^1_k\}$ be the nodes in Stage 1 of the algorithm for cost matrix C. Define a new cost matrix $\tilde{C}^1$ on these set of nodes using Equation 3.

Now, as before, denote the ICM of $\tilde{C}^1$ as $\widehat{C}^1$. Then,

- The algorithm takes $(t-1)$ stages when applied to cost matrix $\tilde{C}^1$. By induction, every edge $N^1_i N^1_j$ belongs to some MCA $g \in M(\widehat{C}^1)$.
- For every $p \in N^+$ and $q \in N \setminus \{p\}$, if $p \in N^1_i$ and $q \in N^1_j$, where $i \neq j$, then

$$c^R_{pq} = \widehat{c}^1_{N^1_i N^1_j} + \Delta^0_q, \tag{10}$$

and

$$c^R_{pq} = \Delta^0_q, \tag{11}$$

if $i = j$.

We now note how we can extend any MCA $g \in M(\widehat{C}^1)$ to an MCA $g' \in M(C^R)$. Consider an edge $N^1_i N^1_j \in g \in M(\widehat{C}^1)$. Replace $N^1_i N^1_j$ by *any edge pq* such that $p \in N^1_i$ and $q \in N^1_j$. Further, choose *any set of edges* involving nodes in N^1_j such that they form an arborescence of nodes N^1_j rooted at q. Repeating this procedure for every $N^1_i N^1_j \in g$ gives an arborescence g'. By Equations 10 and 11, the choice of edge pq does not matter in the total cost of arborescence g'. Since $T(C^R) = T(\widehat{C}^1) + \sum_{p \in N} \Delta^0_p$, we conclude that $g' \in M(C^R)$.

Now, pick any edge pq with $p \in N^+$ and $q \in N \setminus \{p\}$. We consider two possible cases.

Case 1: Both p and q belong to the same supernode N^1_j in Stage 1 of the algorithm. Consider $g \in M(\widehat{C}^1)$. Let $N^1_i N^1_j$ be the edge in g. While extending g to $g' \in M(C^R)$, we replace $N^1_i N^1_j$ by an edge rp, where $r \in N^1_i$. Then, while choosing an arborescence rooted at p in N^1_j we choose the edge pq. By our argument earlier, $g' \in M(C^R)$.

Case 2: The node $p \in N^1_i$ and $q \in N^1_j$ where $i \neq j$. By our induction hypothesis, there is an MCA $g \in M(\widehat{C}^1)$ such that $N^1_i N^1_j$ is an edge in g. While extending g to $g' \in M(C^R)$, we replace $N^1_i N^1_j$ by pq. By our argument earlier $g' \in M(C^R)$.

So, in both cases we can construct an MCA $g' \in M(C^R)$ such that an arbitrary edge pq belongs to g. This concludes the proof. ■

4.3 Properties of C^R

The main aim of the next proposition is to show that the cost game corresponding to the ICM constructed by us is *concave*. The proposition also provides an explicit characterization of the marginal cost that any node i imposes on a coalition S not containing i. This characterization will prove useful subsequently.

Proposition 1 *Consider any cost matrix C, any subset $S \subsetneq N$, and any $i \in N \setminus S$. Then the following claims are true for C^R.*

1. *There exists an MCA of coalition $S \cup \{i\}$ corresponding to cost matrix C^R such that i is a leaf of this MCA.*[17]
2. $c^R(S \cup \{i\}) - c^R(S) = \min_{k \in S^+} c^R_{ki}$.
3. *The cost game (N, c^R) is concave.*

Proof: Fix $S \subsetneq N$ and $i \in N \setminus S$.

Claim: There exists $\bar{g} \in M(C^R)$ such that

- For any $j \in S$, the unique path from 0 to j in $\bar{g}$ contains nodes from S^+ only.
- The unique path from 0 to i in $\bar{g}$ contains nodes from $S^+ \cup \{i\}$ only.

Proof of Claim: We prove the claim by induction on N.

Suppose $|N| = 2$,[18] and pick any $j \in N$. Then, it is obvious that there is $\bar{g} \in M(C^R)$ such that $0j \in \bar{g}$. So, the claim is true when $|N| = 2$.

Let the claim be true whenever $|N| \leq P$ for some integer P, and suppose $|N| = P + 1$.

Suppose now that C^R is such that the algorithm terminates in T stages. Then, the T-stage matrix $\widehat{C}^T$ is a simple matrix. Suppose $N^T = \{N^T_1, \ldots, N^T_K\}$, where N^T is of course the set of supernodes in stage T. Recalling that $N^T_1 \equiv 0$, there are two possible cases - either $K > 2$ or $K = 2$.

Case 1 : $K > 2$. Define $S_k = S \cap N^T_k$ for each $k \in \{2, \ldots, K\}$. Take any $j \in S_q$ and any $i \in N^T_p \setminus S_p$. Then, there is an MCA in $M(C^R)$ in which every N^T_k is connected directly to the source. So, let $\bar{g} = \cup^K_{k=2} \bar{g}_k$, where each $\bar{g}_k$ refers to the subgraph of $\bar{g}$ connecting N^T_k to the source. Now, consider any $\bar{g}_k$. We can view this as the MCA for N^T_k and the restriction of C^R to $N^T_k \cup \{0\}$. But, each $|N^T_k| \leq P$. So, from the induction hypothesis, the unique path from 0 to j in $\bar{g}_q$ will contain only nodes in S^+_q. Similarly, the unique path from 0 to i will contain only nodes in $S^+_p \cup \{i\}$. Hence, the claim is true.

[17] A node in a graph is a leaf if it has no successor nodes.

[18] If $|N| = 1$, then S is empty. Then, the claim is obviously true.

Case 2: $K = 2$, so that $N^T = \{0, N\}$. In this case, consider stage $T-1$ of the algorithm, and let $N^{T-1} = \{N_1^{T-1}, \ldots, N_L^{T-1}\}$, where $N_1^{T-1} \equiv 0$. Note that $L > 2$. It is easy to check that there must be $\bar{g} \in M(C^R)$ such that each N_k^{T-1} is connected directly to the source. We can now use arguments analogous to that of Case 1 to establish the claim.

This concludes the proof of the claim.

Now, consider the subgraph of $\bar{g}$ restricted to $S^+ \cup \{i\}$. This is clearly an MCA for $S \cup \{i\}$ with i being a leaf. This proves (1).

Since i is a leaf of some MCA for $S \cup \{i\}$, (2) follows immediately.

To check (3), take any S, T with $S \subsetneq T \subsetneq N$, and let $i \in N \setminus T$. From (2), we get

$$c^R(S \cup \{i\}) - c^R(S) = \min_{k \in S^+} c_{ki}^R \geq \min_{k \in T^+} c_{ki}^R = c^R(T \cup \{i\}) - c^R(T)$$

Hence, (N, c^R) is concave. ■

This proposition proves that the cost game corresponding to the ICM constructed *by us* is concave. We have pointed out earlier that there may be more than one ICM corresponding to a cost matrix C. However, the following is immediate.

Corollary 1 *If C is an ICM, then the cost game (N, c) is concave.*

Proof: If C is an ICM, then $R(C) = \{C\}$. Hence, the corollary follows from Proposition 1. ■

We now show that a "small" change in C produces a small change in C^R. In other words, C^R changes continuously with C.

Lemma 4 *Suppose C and $\bar{C}$ are such that for some edge ij, $\bar{c}_{ij} = c_{ij} + \epsilon$ for some $\epsilon > 0$, and $c_{kl} = \bar{c}_{kl}$ for all edges $kl \neq ij$. If C^R and $\bar{C}^R$ are the irreducible cost matrices corresponding to C and $\bar{C}$, then*

(i) $0 \leq \bar{c}_{kj}^R - c_{kj}^R \leq \epsilon$ *for all* $k \in N^+ \setminus \{j\}$ *and* $-\epsilon \leq \bar{c}_{kl}^R - c_{kl}^R \leq \epsilon$ *for all* $k \in N^+ \setminus \{j, l\}$,

(ii) for all $l \in N \setminus \{j\}$, *for all* $S \subseteq N \setminus \{l, j\}$,

$$\min_{k \in S^+} \bar{c}_{kj}^R - \min_{k \in S^+} c_{kj}^R \geq \min_{k \in S^+} \bar{c}_{kl}^R - \min_{k \in S^+} c_{kl}^R \tag{12}$$

Proof: Say that two cost matrices C and $\bar{C}$ are *stage equivalent* if ties can be broken in the recursive algorithm such that the number of stages and partitions of nodes in each stage are the same for C and $\bar{C}$.

Proof of (i): We first prove this for the case when C and $\bar{C}$ are stage equivalent. Consider node $j \in N$ such that $\bar{c}_{ij} - c_{ij} = \epsilon > 0$. Consider any $k \in N^+ \setminus \{j\}$. Since C and $\bar{C}$ are stage equivalent, if k and j are t-siblings in cost matrix C, then they are t-siblings in cost matrix $\bar{C}$. We consider two possible cases.

Case 1: Edge ij is the minimum cost incident edge of some supernode containing j in some stage t of the algorithm for cost matrix C, and hence for cost matrix $\bar{C}$ since they are stage equivalent. Then δ_j^t increases by ϵ. But δ_j^{t+1} (if stage $t+1$ exists) decreases by ϵ. Hence, the irreducible cost of no edge can increase by more than ϵ and the irreducible cost of no edge can decrease by more than ϵ. Moreover, we prove that for edge kj, the irreducible cost cannot decrease. To see this, note that the irreducible cost remains the same if k and j are t' siblings and $t' \leq t$ or $t' > t+1$. If $t' = t+1$, then the irreducible cost of kj only increases.

Case 2: Edge ij is not the minimum cost incident edge of any supernode containing j in any stage of the algorithm for cost matrix C, and hence for cost matrix $\bar{C}$. In that case, δ_j^t remains the same for all t. Hence $C^R = \bar{C}^R$.

Examining both the cases, we conclude that $0 \leq \bar{c}_{kj}^R - c_{kj}^R \leq \epsilon$ for all $k \in N^+ \setminus \{j\}$. Also, $-\epsilon \leq (\bar{c}_{kl}^R - c_{kl}^R) \leq \epsilon$ for all $l \in N \setminus \{k, j\}$.

We complete the proof by arguing that cost of edge ij can be increased from c_{ij} to $\bar{c}_{ij}$ by a finite sequence of increases such that cost matrices generated in two consecutive sequences are stage equivalent.

Define *rank* of an edge ij in cost matrix C as $rank^C(ij) = |\{kl : k \in N^+ \setminus \{i\}, l \in N \setminus \{k\}, c_{kl} > c_{ij}\}|$. Clearly, two edges ij and kl have the same rank if and only if $c_{ij} = c_{kl}$. Note that if ranks of edges do not change from C to $\bar{C}$, then we can always break ties in the same manner in the recursive algorithm in C and $\bar{C}$, and thus C and $\bar{C}$ are stage equivalent.

Suppose $rank^C(ij) = r > rank^{\bar{C}}(ij) = \bar{r}$. Consider the case when $\bar{r} = r - 1$. This means that a unique edge $kl \neq ij$ exists such that $c_{kl} > c_{ij}$ but $\bar{c}_{kl} = c_{kl} \leq \bar{c}_{ij}$. Consider an *intermediate* cost matrix $\hat{C}$ such that $\hat{c}_{ij} = c_{kl} = \bar{c}_{kl}$ and $\hat{c}_{pq} = c_{pq} = \bar{c}_{pq}$ for all edges $pq \neq ij$. In the cost matrix $\hat{C}$, one can break ties in the algorithm such that one chooses ij over kl everywhere. This will generate the same stages of the algorithm with same partitions of nodes in every stage for cost matrix C and $\hat{C}$. Hence, C and $\hat{C}$ are stage equivalent. But we can also break the ties in favor of edge kl everywhere, and this will generate the same set of stages and partitions as in cost matrix $\bar{C}$. This shows that $\hat{C}$ and $\bar{C}$ are also stage equivalent.[19].

If $\bar{r} < r - 1$, then we increase the cost of edge ij from c_{ij} in a finite number of steps such that at each step, the rank of ij falls by exactly one.

Proof of (ii): For simplicity, we only consider the case where C and $\bar{C}$ are stage-equivalent. As argued in the proof of (i), the argument extends easily to the case when they are not

[19]Note that this does not imply C and $\bar{C}$ are stage equivalent.

stage-equivalent.

Let ij be the minimum cost incident edge of N_k^t. Then, for any $p \in N_k^t$,

$$\bar{c}_{qp}^R > c_{qp}^R \text{ implies } p, q \text{ are } (t+1)\text{-siblings} \tag{13}$$

Also, for all $p \in N_k^t$ and all q which are $(t+1)$-siblings,

$$\bar{c}_{qp}^R = c_{qp}^R + \epsilon$$

So, to prove that Equation 12 is true, we only need to consider any $l \in N_k^t, l \neq j$. Consider any $S \subseteq N^+ \setminus \{l, j\}$, and suppose $\min_{k \in S} \bar{c}_{kl}^R > \min_{k \in S} c_{kl}^R$. Then, from Equation 13, S must contain some q which is a $(t+1)$-sibling of l. Moreover, since the irreducible cost of edges which are t'-siblings are higher than those which are $(t'-1)$-siblings for all t', S cannot contain any siblings which are t'-siblings of l for $t' \leq t$. This is because if S did contain some m which was a t'-sibling with $t' \leq t$, then $\min_{k \in S} \bar{c}_{kl}^R$ would not be attained at a $(t+1)$-sibling of l. But, then Equation 13 establishes that

$$\min_{k \in S} \bar{c}_{kl}^R - \min_{k \in S} c_{kl}^R = \min_{k \in S} \bar{c}_{kj}^R - \min_{k \in S} c_{kj}^R = \epsilon$$

This completes the proof of the lemma. ■

4.4 The Cost Allocation Rule f^*

We identify a cost allocation rule f^* with the Shapley value of the cost game (N, c^R). So,

$$f^*(C) \equiv Sh(N, c^R)$$

A main result of the paper is to show that f^* satisfies CS, DSCM, and CON. We then go on to show that f^* also satisfies S, INV and R.

Theorem **4** *The cost allocation f^* satisfies CS, DSCM, and CON.*

Proof: Take any cost matrix C. Let C^R be the irreducible cost matrix constructed by us. From Proposition 1, the game (N, c^R) is concave. Hence, the Shapley value of (N, c^R) is in the core of the game (N, c^R). Of course, $c_{ij}^R \leq c_{ij}$ for all edges ij. Using Lemma 2 it follows that the core of the game (N, c^R) is contained in the core of the game (N, c). Hence, f^* satisfies CS.

From Proposition 1, $c^R(S \cup \{k\}) - c^R(S) = \min_{j \in S^+} c_{jk}^R$ for any $S \subseteq N \setminus \{k\}$. By Lemma 4, C^R changes continuously with C. Hence, the cost game (N, c^R) also changes continuously with C. The continuity of the Shapley value with respect to the cost game establishes that f^* satisfies continuity.

We use Proposition 1 and Equation 12 to prove that f^* satisfies DSCM. Consider two cost matrices C and $\bar{C}$ as in Lemma 4. From Proposition 1, the marginal costs of any l to a coalition $S \subseteq N \setminus \{l\}$ for C^R and $\bar{C}^R$ are given by

$$\min_{k \in S^+} c^R_{kl} \text{ and } \min_{k \in S^+} \bar{c}^R_{kl}$$

Using the formula for the Shapley value and Equation 12, it is straightforward to verify that DSCM is satisfied. ■

We now show that f^* satisfies INV, S and R. We first show that f^* satisfies INV and S. We then show that *any* rule which satisfies INV, S and DSCM must satisfy R. This will establish that f^* also satisfies R.

PROPOSITION **2** *The allocation rule f^* satisfies INV and S.*

Proof: Consider cost matrices C and $\bar{C}$ such that for some $j \in N$ and for all $i \in N^+ \setminus \{j\}$, $\bar{c}_{ij} = c_{ij} + \epsilon$, and $\bar{c}_{pq} = c_{pq}$ for all $q \in N \setminus \{j\}$ and for all $p \in N^+ \setminus \{q\}$. For any node $k \in N$, if $c_{ik} \leq c_{pk}$ for all $p \in N^+ \setminus \{i\}$, then $\bar{c}_{ik} \leq c_{pk}$. Hence, $\bar{C}^1 = C^1$ from the recursive algorithm and, $\bar{c}^R_{pq} = c^R_{pq}$ for all $q \in N \setminus \{j\}$ and for all $p \in N^+ \setminus \{q\}$. By Part (2) of Proposition 1 and the definition of the Shapley value, we get that $f^*_q(\bar{C}) = f^*_q(C)$ for all $q \in N \setminus \{j\}$. But, $T(\bar{C}) = T(C) + \epsilon$. Hence, $f^*_j(\bar{C}) = f^*_j(C) + \epsilon$. So, f^* must satisfy INV.

We now prove that f^* satisfies S. Consider two agents i and j and cost matrix C such that $c_{ij} = c_{ji}$, $c_{ki} = c_{kj}$ for all $k \notin \{i, j\}$, and $c_{ik} = c_{jk}$ for all $k \notin \{0, i, j\}$.
STEP 1: Let $\hat{t}$ be the last stage of the algorithm such that both i and j form singleton supernodes by themselves. Note that $\Delta^0_i = \Delta^0_j$. If $\hat{t} > 0$, then $\Delta^t_i = \Delta^t_j = 0$ for all $0 < t \leq \hat{t}$.

Without loss of generality, $(i, i^1, \ldots, i^k)$ form a $C^{\hat{t}}$ cycle in Stage $\hat{t}$ for some (super)nodes $i^1, \ldots, i^k$ of Stage $\hat{t}$. If $j \in \{i^1, \ldots, i^k\}$, then $\delta^t_i = \delta^t_j$ for all $t > \hat{t}$. Otherwise, $c^{\hat{t}}_{i^k i} = \Delta^{\hat{t}}_i$. Because of our assumption on C, $c^{\hat{t}}_{i^k i} = c^{\hat{t}}_{i^k j} = \Delta^{\hat{t}}_j$. Hence, j cannot be part of a $C^{\hat{t}}$-cycle.

Denote the supernode containing i in Stage $\hat{t}+1$ to be $i' \equiv N^{\hat{t}+1}_i$. Now, $c^{\hat{t}+1}_{i'j} = 0$. Also, $c^{\hat{t}}_{ii^1} = c^{\hat{t}}_{ji^1}$ implies that $c^{\hat{t}+1}_{ji'} = 0$. Hence, (i) $\delta^{\hat{t}+1}_i = \delta^{\hat{t}+1}_j = 0$ and (ii) i and j form a $C^{\hat{t}+1}$-cycle, and hence, $(\hat{t}+2)$-sibling. Thus, for all $t \leq T-1$, we have $\delta^t_i = \delta^t_j$.

STEP 2: Consider any node $k \notin \{i, j\}$. Suppose k and i are $\bar{t}$-sibling and k and j are $\tilde{t}$-sibling with $\bar{t} \leq \tilde{t}$. By assumption $\bar{t} \geq \hat{t}+1$. By Step 1, if $\bar{t} > \hat{t}+1$, then $\bar{t} = \tilde{t}$. In that case, by Step 1, $c^R_{ki} = \sum_{t=1}^{\bar{t}} \delta^t_i = \sum_{t=1}^{\tilde{t}} \delta^t_j = c^R_{kj}$.

If $\bar{t} = \hat{t}+1$, then Step 1 implies that $\tilde{t} = \bar{t}+1$ as j merges with the supernode containing i in Stage $\hat{t}+1$. But $\delta^{\hat{t}+1}_i = \delta^{\hat{t}+1}_j = 0$, implies that $c^R_{kj} = \sum_{t=1}^{\hat{t}} \delta^t_i$. Hence, by Step 1 again, we conclude that $c^R_{ki} = \sum_{t=1}^{\hat{t}} \delta^t_i = \sum_{t=1}^{\hat{t}} \delta^t_j = c^R_{kj}$.

Step 3: Since for all $k \notin \{i, j\}$, $c^R_{ki} = c^R_{kj}$, $f^*_i(C) = f^*_j(C)$ by the definition of the Shapley value and Proposition 1.

■

Proposition **3** *If an allocation rule satisfies S, INV, and DSCM, then it satisfies R.*

Proof: Let μ be a cost allocation rule which satisfies S, INV, and DSCM. Consider a cost matrix C such that for some $i, j \in N$, we have $c_{ik} = c_{jk}$ and $c_{ki} > c_{kj}$ for all $k \notin \{i, j\}$, and $c_{ji} > c_{ij}$. Consider another cost matrix $\hat{C}$ such that $\hat{c}_{ik} = \hat{c}_{jk} = c_{ik} = c_{jk}$ and $\hat{c}_{ki} = \hat{c}_{kj} = c_{kj} < c_{ki}$ for all $k \neq i, j$ and $\hat{c}_{ij} = \hat{c}_{ji} = c_{ij} < c_{ji}$. By symmetry, $\mu_j(\hat{C}) = \mu_i(\hat{C})$.

Now, let $\epsilon = \min_{k \neq i}[c_{ki} - \hat{c}_{ki}]$. Note that $\epsilon > 0$ by assumption. Consider a cost matrix $\bar{C}$ defined as follows: $\bar{c}_{ki} = \hat{c}_{ki} + \epsilon$ for all $k \neq i$ and $\bar{c}_{pq} = \hat{c}_{pq}$ for all p, q with $q \neq i$. So, we increase cost of incident edges on i from $\hat{C}$ to $\bar{C}$ by the same amount ϵ, whereas costs of other edges remain the same. By invariance, $\mu_i(\bar{C}) = \mu_i(\hat{C}) + \epsilon$ and $\mu_q(\bar{C}) = \mu_q(\hat{C})$ for all $q \in N \setminus \{i\}$.

Hence, $\mu_i(\bar{C}) > \mu_i(\hat{C}) = \mu_j(\hat{C}) = \mu_j(\bar{C})$. Since μ satisfies DSCM, $\mu_i(C) - \mu_j(C) \geq \mu_i(\bar{C}) - \mu_j(\bar{C}) > 0$. This implies that $\mu_i(C) > \mu_j(C)$. ■

An immediate corollary to Proposition 3 is that f^* satisfies R.

Corollary **2** *The allocation rule f^* satisfies R.*

Proof: The allocation rule f^* satisfies S and INV due to Proposition 2 and DSCM due to Theorem 4. By Proposition 3, f^* satisfies R. ■

Notice that Property R requires that if the incoming edges of node i cost strictly more than the corresponding incoming edges for j while corresponding outgoing edges cost the same, then the cost allocated to i should be strictly higher than the cost allocated to j. But, now suppose *both* incoming *and* outgoing edges of i cost strictly more than those of j. Perhaps, one can argue that if the outgoing edges of i cost more than the outgoing edges of j, then i is less "valuable" in the sense that i is going to be used less often in order to connect to other nodes. Hence, in this case too, i should pay strictly more than j. However, it turns out that f^* does not satisfy this modified version of R. This is demonstrated below.

Example **3** *Let $N = \{1, 2, 3\}$. Figure 5 shows a cost matrix C (assume $e > 0$ in Figure 5) and its associated irreducible cost matrix, the latter being shown on the right. It is easy to see that $f^*_1(C) = e, f^*_2(C) = f^*_3(C) = 1/2$. So, for $e < 1/2$, agent 1 pays less than agent 2, though agent 2 has strictly lower incoming and outgoing edge costs than agent 1.*

We do not know whether there are other rules which satisfy the basic axioms and this modified version of R.

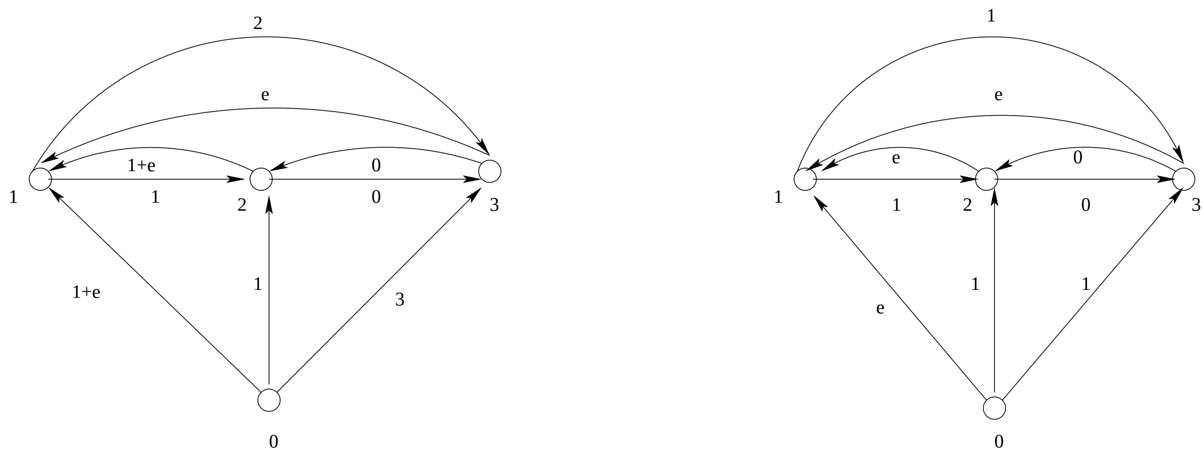

Figure 5: Example illustrating violation of the modified version of Ranking

4.5 Coincidence of f^* with the Folk Solution on Symmetric Cost Matrices

Here, we show that our solution coincides with the folk solution for the MCST problem. This result may seem puzzling in view of Example 2 which showed that even when the cost matrix is symmetric, there may be multiple ICMs all of whom are asymmetric. Of course, the irreducible cost matrix for the MCST problem must necessarily be symmetric. Notice, however that despite the multiplicity of ICMs in the example, the folk solution and our solution both prescribe the cost allocation where 1 pays 1 and 2 pays 2.[20]

Let ϕ denote the folk solution for MCST games. We prove the following.

Theorem **5** *Let C be a symmetric cost matrix. Then for every agent $i \in N$,*

$$f_i^*(C) = \phi_i(C).$$

We prove this theorem by invoking an axiomatic characterization of the folk solution due to Bergantinos and Vidal-Puga (2007a), who show that if a solution satisfies *Independence of Irrelevant Trees, Separability, and Equal Share of Extra Costs* on the set of symmetric cost matrices, then it must coincide with the folk solution. We now show that our solution satisfies these axioms on the domain of symmetric cost matrices.

We begin by formally defining these axioms. To define our first axiom, we need the notion of weak arborescence equivalence. We say that two cost matrices C and C' are *weak arborescence equivalent* if there is an arborescence g such that $g \in M(C) \cap M(C')$, and for every ij in g, we have $c_{ij} = c'_{ij}$. Note that if two cost matrices are arborescence equivalent, then they are also weak arborescence equivalent, but the converse is not true.

[20]The cost matrix in Example 2 is simple, and so our ICM corresponds to $\epsilon = 0$.

Definition 17 *A cost allocation rule μ satisfies Independence of Irrelevant Trees (IIT) if for every pair of cost matrices C and C' which are weak arborescence equivalent, we have $\mu_i(C) = \mu_i(C')$ for all $i \in N$.*

Remark 6 *Note that IIT implies IIC.*

Since our solution does not satisfy IIC on the entire domain of cost matrices, it obviously does not satisfy IIT. However, we show below that our solution does satisfy IIT when the domain is restricted to symmetric cost matrices.

Proposition 4 *Let C, C' be symmetric cost matrices. If C and C' are weak arborescence equivalent, then for every agent $i \in N$,*

$$f_i^*(C) = f_i^*(C').$$

Proof: Suppose C, C' are symmetric cost matrices which are weak arborescence equivalent. Let $g \in M(C) \cap M(C')$. Note that since C and C' are symmetric and weak arborescence equivalent, this means that if $pq \in g$, then $c_{pq} = c_{qp} = c'_{pq} = c'_{qp}$.

We will show that the ICM produced by our procedure is the same for C and C', and this will prove the proposition. We first observe that by breaking ties appropriately, we can always find g using the recursive algorithm (for this, we need to break ties in favor of edges included in g). Second, every MCA corresponding to a symmetric cost matrix corresponds to an MCST of the undirected problem.

We use induction on the number of stages T the algorithm takes to find g in cost matrix C. We do the proof in various steps.

Step 1: Suppose $T = 0$, so that C is a simple cost matrix. So, $c_{ij}^R = \min_{k \in N^+} c_{ki}$. It suffices to show that the minimum incident edge cost of every node does not change from C to C'.

We show that C' is also a simple cost matrix with g being the greedy digraph which is in $M(C')$. Suppose not. Let $ij \in g$, and hence, $c'_{ij} = c_{ij}$. Suppose c'_{ij} is not the minimum incident cost edge to j. Then there is a node $k \in N^+ \setminus \{i, j\}$, such that $c'_{kj} < c'_{ij} = c_{ij}$ and $c'_{kj} \leq c'_{pj}$ for all $p \in N^+ \setminus \{j\}$. By assumption $c'_{kj} = c'_{jk} \neq c_{kj} = c_{jk}$. Hence, $kj, jk \notin g$ since costs of edges in g do not change. We consider two cases:

Case 1: There is no directed path from j to k in g. In that case consider the arborescence $g' = (g \setminus \{ij\}) \cup \{kj\}$. Since there is no path from j to k in g, g' is an arborescence. The difference between the total cost of g and g' in cost matrix C' is $c'_{ij} - c'_{jk} > 0$. Hence, $g \notin M(C')$, a contradiction.

Case 2: There is a directed path $(j, j^1, \ldots, j^r, k)$ from j to k in g. This means i and j lie in the path from 0 to k in g.

Consider any pair of successive edges in the path from i to k, say ij, jj^1. Since C is a simple cost matrix, $c_{ij} \leq c_{j^1 j}$. Since C is symmetric, $c_{j^1 j} = c_{jj^1}$. Hence, $c_{ij} \leq c_{jj^1}$. Continuing in this manner, we must have, $c_{ij} \leq c_{j^r k}$. Since $ij, j^r k \in g$, $c'_{ij} = c_{ij} \leq c_{j^r k} = c'_{j^r k}$. But $c'_{jk} < c'_{ij}$ implies that $c'_{jk} < c'_{j^r k}$. Now consider the arborescence $g' = (g \setminus \{j^r k\}) \cup \{jk\}$. Clearly, g' is an arborescence, and the difference in cost of g and g' is $c'_{j^r k} - c'_{jk} > 0$. This implies that $g \notin M(C')$, a contradiction.

So, in both cases we have reached a contradiction. This establishes that C' is also a simple matrix with g being a greedy digraph which is in $M(C')$. So, the minimum incident edge cost to every node is the same in both C and C'. This completes Step 1.

STEP 2: Suppose $T > 0$. In this step, we show that if we run the algorithm to find the MCA g for cost matrices C and C', the nodes produced in the first stage are the same.

Let $(i^1, i^2, \ldots, i^r, i^1)$ be a C-cycle (i.e., a cycle produced in Stage 0 of the algorithm for cost matrix C). First, note that all edge costs in the cycle are equal. This follows straightaway from symmetry if $r = 2$. Suppose $r > 2$. Since the given cycle is a C-cycle, $c_{i^1 i^2} \leq c_{i^3 i^2} = c_{i^2 i^3}$. Continuing this way, we get $c_{i^1 i^2} \leq c_{i^2 i^3} \leq \cdots \leq c_{i^r i^1}$. But similarly, $c_{i^r i^1} \leq c_{i^1 i^2}$. Hence, $c_{i^1 i^2} = c_{i^2 i^3} = \ldots = c_{i^r i^1}$.

Suppose $r > 2$. Let $i^1 i^2, i^2 i^3, \ldots, i^{r-1} i^r \in g$. Since $c_{i^1 i^2} = c_{i^2 i^3} = \ldots = c_{i^r i^1}$, we can break ties in a manner such that we form a cycle (i^1, i^2, i^1), and then take the minimum cost incoming edge to i^3 as $i^2 i^3$, to i^4 as $i^3 i^4$, ..., to i^r as $i^{r-1} i^r$. This will also recover the MCA g in the algorithm. Thus, we can assume without loss of generality that every C-cycle contains two nodes only.

Let (i, j, i) be a C-cycle. Then, either ij or ji is in g, and so $c_{ij} = c_{ji} = c'_{ij} = c'_{ji}$. We will show that (i, j, i) is a C'-cycle.

If this is not true, then there is $k \in N^+ \setminus \{i, j\}$, $c'_{ki} < c'_{ji} = c_{ji}$.

CASE 1: Suppose $ji \in g$. We consider two possible sub-cases.

- CASE 1A: There is no path from i to k in g. In that case, the digraph $g' = (g \setminus \{ji\}) \cup \{ki\}$ is an arborescence, and the difference in total cost of g and g' in cost matrix C' is $c'_{ji} - c'_{ki} > 0$. Hence, $g \notin M(C')$, a contradiction.
- CASE 1B: There is a path $(i, i^1, \ldots, i^r, k)$ in g. Now, consider the digraph $g' = (g \setminus \{ii^1, i^1 i^2, \ldots, i^r k\}) \cup \{ik, ki^r, i^r i^{r-1}, \ldots, i^2 i^1\}$. Note that $r > 1$ since $ik, ki \notin g$. It is easy to see that g' is also an arborescence. The difference in cost of g and g' in cost matrix C' is $c'_{ii^1} - c'_{ik}$. Since $ii^1 \in g$, $c'_{ii^1} = c_{ii^1}$. Further since (i, j, i) is a C-cycle, $c_{ii^1} \geq c_{ji} > c'_{ki} = c'_{ik}$. Hence, $c'_{ii^1} > c'_{ik}$. This means $g \notin M(C')$, a contradiction.

CASE 2: Suppose $ij \in g$. Let $pi \in g$. Note that since (i, j, i) is a C-cycle, $c_{ji} = c'_{ji} \leq c_{pi} = c'_{pi}$. Hence, $c'_{ki} < c'_{pi}$. We now consider two possible sub-cases.

- Case 2a: There is no path from i to k in g. In the case the digraph $g' = (g \setminus \{pi\}) \cup \{ki\}$ is an arborescence. The difference in total cost of g and g' in cost matrix C' is $c'_{pi} - c'_{ki} > 0$. Hence, $g \notin M(C')$, a contradiction.
- Case 2b: There is a path $(i, i^1, i^2, \ldots, i^s, k)$ in g. Note that $s > 1$ since $ik, ki \notin g$. Now, consider the digraph $g' = (g \setminus \{ii^1, i^1i^2, \ldots, i^sk\}) \cup \{ik, ki^s, \ldots, i^2i^1\}$. Clearly, g' is an arborescence. Moreover, the difference in cost of g and g' in cost matrix C' is $c'_{ii^1} - c'_{ik}$. Since $ii^1 \in g$, $c'_{ii^1} = c_{ii^1}$. Further since (i, j, i) is a C-cycle, $c_{ii^1} \geq c_{ji} > c'_{ki} = c'_{ik}$. Hence, $c'_{ii^1} > c'_{ik}$. Hence, $g \notin M(C')$, which is a contradiction.

Since we reached a contradiction in both cases, (i, j, i) is also a C'-cycle. The choice of (i, j, i) was arbitrary. Hence, the set of C-cycles is the same as the set of C'-cycles. So, the nodes produced in the first stage are the same.

Step 3: Suppose $T > 0$. Let $\widetilde{C}^1$ and $\widetilde{C}'^1$ be the first stage cost matrices of C and C' respectively. By Step 2, both these cost matrices are defined on the same set of nodes. Note that g induces an arborescence for these set of nodes, and denote it by g^1. Since $g \in M(C) \cap M(C')$, $g^1 \in M(\widetilde{C}^1) \cap M(\widetilde{C}'^1)$. The algorithm to find an MCA takes one less stage for cost matrix C^1. Hence, by the induction hypothesis, the ICM produced by our procedure is the same for $\widetilde{C}^1$ and $\widetilde{C}'^1$.

Now, by Step 2, the nodes produced in $\widetilde{C}^1$ and $\widetilde{C}'^1$ are the same. Consider any arbitrary node $j \in N$. We argue that Δ^0_j is the same in C and C'. If j is involved in a C-cycle, by Step 2, it is also involved in a C'-cycle. This means either ij or ji is in g. Hence, $c'_{ij} = c_{ij}$. So, Δ^0_j must be same in C and C'. If j is not involved in a C-cycle, then it is also not involved in C'-cycle. In that case, if $kj \in g$, then Δ^0_j must equal c_{kj} in cost matrix C. Similarly, Δ^0_j must equal c'_{kj} in cost matrix C'. Hence, $c'_{kj} = c_{kj} = \Delta^0_j$. Now, consider any arbitrary edge ij in the original graph. If i and j form a C-cycle, and also a C'-cycle, then the irreducible cost of ij in C and C' is Δ^0_j. If i and j do not form a C-cycle, and also do not form a C'-cycle, then the irreducible cost of ij is irreducible cost of pq plus Δ^0_j, where p is the node in Stage 1 containing i and q is the node in Stage 1 containing j. Since the irreducible cost of pq is the same in $\widetilde{C}^1$ and $\widetilde{C}'^1$, this implies that the irreducible cost of ij is the same in C and C'. This concludes the proof. ■

Now, we define Separability. For this we use the notation $\mu(C^S)$ to denote the cost allocation of agent i in cost matrix C when the set of agents is S and the cost allocation rule is μ.

Definition **18** *A cost allocation rule μ satisfies Separability (SEP) if for all cost matrices C and for all $S \subsetneq N$ such that $c(N) = c(S) + c(N \setminus S)$, we have*

$$\mu_i(C^N) = \begin{cases} \mu_i(C^S) & \text{if } i \in S \\ \mu_i(C^{N \setminus S}) & \text{if } i \in N \setminus S. \end{cases}$$

The proof that f^* satisfies SEP uses a well-known consistency property of the Shapley value demonstrated by Hart and Mas-Colell (1989).

Let φ be any solution concept, and (N, v) a game. Take any $M \subsetneq N$ and define the *reduced game*

$$v_M^{\varphi}(S) = v(S \cup M^c) - \sum_{i \in M^c} \varphi_i(S \cup M^c, v) \text{ for any } S \subseteq M$$

where $M^c \equiv N \setminus M$.

Then, φ satisfies the *HM reduced game property* if for all $M \subsetneq N$,

$$\varphi_i(M, v_M^{\phi}) = \varphi_i(N, v) \text{ for all } i \in T$$

Hart and Mas-Colell (1989) proves that the Shapley value satisfies the HM reduced game property.

PROPOSITION 5 *The cost allocation rule f^* satisfies SEP.*

Proof: Let C be a cost matrix and $M \subsetneq N$ be such that $c(N) = c(M) + c(M^c)$.
Claim 1 : For all $S \subsetneq M$, $c^R(S \cup M^c) = c^R(S) + c^R(M^c)$

Take any $i \in S \subsetneq M$ and $j \in M^c$. Since $c(N) = c(M) + c(M^c)$, there is an MCA g such that $g \cap \{ij, ji\} = \emptyset$. This means i and j are $(T+1)$-siblings, where T is the final stage of the algorithm to find this MCA. So $c_{ij}^R = \sum_{t=0}^{T} \Delta_j^t = c_{0j}^R$. Similarly, $c_{ji}^R = c_{0i}^R$. Hence, there is an MCA corresponding to C^R for $(S \cup M^c)$ which is the union of MCAs for S and M^c, and so the claim is true.

Claim 2 : For all $S \subset M$, $\sum_{i \in M^c} Sh_i(S \cup M^c, c^R|(S \cup M^c)) = c^R(M^c)$ where $c^R|(S \cup M^c)$ is c^R restricted to $S \cup M^c$.

From Claim 1, $c^R(S) + c^R(M^c) = c^R(S \cup M^c)$. Since $c^R|(S \cup M^c)$ is convex, Sh belongs to the core of c^R. This immediately establishes Claim 2.

Claim 3: For all $S \subset M$, $c^R(S) = c_M^{R,Sh}(S)$ where $c_M^{R,Sh}$ is the reduced game with player set M derived from c^R and Sh. This follows from the following.

$$\begin{aligned} c_M^{R,Sh}(S) &= c^R(S \cup M^C) - \sum_{i \in M^c} Sh_i(S \cup M^c, c^R|(S \cup M^c)) \text{ from definition of reduced game} \\ &= c^R(S \cup M^C) - c^R(M^c) \text{ from Claim 2} \\ &= c^R(S) \text{ from Claim 1} \end{aligned}$$

SEP follows from the fact that Sh satisfies the HM reduced game property.

■

We now define our final axiom.

Definition 19 *A cost allocation rule μ satisfies Equal Share of Extra Costs (ESEC) if for any two C, C' such that $c_{0i} = c_0$ for all $i \in N$, $c'_{0i} = c'_0$ for all $i \in N$, $c_0 < c'_0$, and $c_{ij} = c'_{ij} \leq c_0$ for all $i \in N$, $j \in N \setminus \{i\}$, we have*

$$\mu_i(C') = \mu_i(C) + \frac{c'_0 - c_0}{n}.$$

Proposition 6 *The allocation rule f^* satisfies ESEC.*

Proof: We consider a new cost matrix $\hat{C}$ as follows: $\hat{c}_{0i} = c'_0 - c_0$ for all $i \in N$ and $\hat{c}_{ij} = 0$ for all $i \in N$ and for all $j \in N \setminus \{i\}$. Note that $\hat{C}$ is an ICM. Further, $f^*_i(\hat{C}) = \frac{c'_0 - c_0}{n}$ for all $i \in N$. Notice that the worth of any coalition $S \subseteq N$, in the cooperative game $(N, \hat{c})$ is $\hat{c}(S) = c'_0 - c_0$.

Next, apply the recursive algorithm to C and C'. In the case of C, break ties against $0j$ for all j if necessary. Then, since the restrictions of C and C' on $N \times N$ coincide, the structure of supernodes will be the same for both cost matrices. Also, 0 will be the last node to be connected in either case. Hence, $c^R_{ij} = c'^R_{ij}$ for all $i, j \in N$, and $c^R_{0j} = c_0, c'^R_{0j} = c'_0$ for all j.

Then, for any coalition $S \subseteq N$, an MCA corresponding to either of the cost matrices will connect some node $i \in N$ to the source, and connect the remaining nodes from i. It is straightforward to verify that $c'^R(S) = c^R(S) + (c'_0 - c_0)$. Hence, $c'^R(S) = c^R(S) + \hat{c}(S)$.

Since the Shapley value satisfies Additivity, it follows that for all $i \in N$, $f^*_i(C') = f^*_i(C) + f^*_i(\hat{C}) = f^*_i(C') + \frac{c'_0 - c_0}{n}$. This proves that f satisfies ESEC. ■

Remark 7 *Notice that f^* satisfies SEP and ESEC on the entire domain of cost matrices, but IIT only on the domain of symmetric cost matrices.*

The proof of Theorem 5 follows straightaway. By Propositions 4, 5, and 6, our cost allocation rule satisfies IIT, SEP and ESEC on the set of symmetric cost matrices. By Bergantinos and Vidal-Puga (2007a), the only rule that satisfies these axioms on the set of symmetric cost matrices is the folk solution. Hence, our cost allocation rule must coincide with the folk solution on the set of symmetric cost matrices.

5 Characterization Result

In this section, we provide a characterization result for f^* when the domain of the allocation rule is restricted to $\mathcal{C}^{sim}_N$, the set of simple cost matrices.

Theorem 6 *Suppose the domain of the allocation rules is restricted to $\mathcal{C}^{sim}_N$. Then, the following statements are equivalent.*

1. *An allocation rule satisfies INV and NN.*

2. *An allocation rule is the unique Bird allocation rule.*

3. *An allocation rule satisfies IOC and SS.*

4. *An allocation rule is f^*.*

Proof: Consider any cost matrix $C \in \mathcal{C}_N^{sim}$. Clearly, there is a unique Bird allocation in C with

$$b_i(C) = \min_{j \in N^+, j \neq i} c_{ji} \equiv \Delta_i^0$$

The ICM C^R of C obtained from our procedure satisfies $c_{ij}^R = \Delta_j^0$ for all $j \in N$ and for all $i \in N^+ \setminus \{j\}$. But $f_j^*(C) = \Delta_j^0$ for all $j \in N$. Hence, f^* coincides with the Bird allocation rule. So, (2) and (4) are equivalent.

It is easy to see that f^* satisfies SS, NN, IOC and hence INV, on the domain $\mathcal{C}_N^{sim}$. Now, suppose a cost allocation rule μ satisfies INV and NN. For any cost matrix $C \in \mathcal{C}_N^{sim}$, we intend to show that $\mu_i(C) = \Delta_i^0$ for all $i \in N$, and this will show that μ is f^*. Consider another cost matrix $\bar{C}$ such that for all $j \in N$ and for all $i \in N^+ \setminus \{j\}$ we have $\bar{c}_{ij} = c_{ij} - \Delta_j^0$. Clearly, $\bar{C} \in \mathcal{C}_N^{sim}$. By repeated application of invariance, $\mu_i(C) = \mu_i(\bar{C}) + \Delta_i^0$ for all $i \in N$. The total cost of an MCA for $\bar{C}$ is zero. Hence, by NN, $\mu_i(\bar{C}) = 0$ for all $i \in N$. Hence, $\mu_i(C) = \Delta_i^0$ for all $i \in N$.

Now, suppose a cost allocation rule μ satisfies IOC and SS. Consider any cost matrix $C \in \mathcal{C}_N^{sim}$. Fix any $k \in N$. Consider two cost matrices $\bar{C}$ and C^0 as follows: $c_{ij}^0 = 0$ for all edges ij, and $\bar{c}_{ik} = c_{ik}$ for all $i \in N^+ \setminus \{k\}$ and $\bar{c}_{ij} = 0$ for all $j \in N \setminus \{k\}$ and for all $i \in N^+ \setminus \{j\}$. It is easy to see that $C^0, \bar{C} \in \mathcal{C}_N^{sim}$. By SS, $\mu_i(C^0) = 0$ for all $i \in N$. By IOC, $\mu_i(\bar{C}) = \mu_i(C^0) = 0$ for all $i \in N \setminus \{k\}$. Hence, $\mu_k(\bar{C}) = b_k(\bar{C})$. Applying IOC again, we get $\mu_k(\bar{C}) = \mu_k(C) = b_k(C)$. So, (3) implies (2). ■

6 Conclusion

In this paper, we study the cost allocation problem in *directed* spanning network problems when the cost matrix is asymmetric - the cost associated with the directed edge ij is not necessarily the same as the cost edge ji. This distinguishes our framework from that of the well-studied minimum cost spanning tree problems. We adopt an axiomatic approach to the cost allocation problem. Our first major result establishes the difference in the two frameworks. It is well-known that in the MCST problem, there are a large number of continuous rules picking out an allocation in the core of the cost game and other properties such as Cost Monotonicity and an invariance condition which requires only that the allocation depends on the costs of edges figuring in minimum cost spanning trees. In contrast, we show that no rule can satisfy analogous properties in our framework. Our second major result is the construction of an allocation rule which satisfies some "basic" axioms. This allocation rule can be viewed as a "natural" extension of the folk solution for MCST problems since it coincides with the latter on the domain of symmetric cost matrices. We show that the rule constructed by us satisfies additional properties. We also provide an axiomatic characterization of our rule on a restricted domain of cost matrices.

However, we do not have a characterization of our rule on the complete domain. Neither do we know about the existence of other rules satisfying the basic axioms. These issues are left for future work.

References

Bergantinos, G. and A. Kar (2010): "On Obligation Rules for Minimum Cost Spanning Tree Problems," *Games and Economic Behavior*, 69, 224-237.

Bergantinos, G. and J. J. Vidal-Puga (2007a): "A Fair Rule for Minimum Cost Spanning Tree Problems," *Journal of Economic Theory*, 137, 326–352.

——— (2007b): "The Optimistic TU Game in Minimum Cost Spanning Tree Problems," *International Journal of Game Theory*, 36, 223–239.

Bird, C. G. (1976): "On Cost Allocation of a Spanning Tree: A Game Theoretic Approach," *Networks*, 6, 335–350.

Bogomolnaia, A., R. Holzman, and H. Moulin (2010): "Sharing the Cost of a Capacity Network," *Mathematics of Operations Research*, 35, 173–192.

Bogomolnaia, A. and H. Moulin (2010): "Sharing a Minimal Cost Spanning Tree: Beyond the Folk Solution," *Games and Economic Behavior*, 69, 238-248.

Branzei, R., S. Moretti, H. Norde, and S. Tijs (2004): "The P-Value for Cost Sharing in Minimum Cost Spanning Tree Situations," *Theory and Decision*, 56, 47–61.

——— (2005): "The Bird Core for Minimum Cost Spanning Tree Problems Revisited: Monotonicity and Additivity Aspects," Working Paper, Tilburg University (CentER DP).

Chu, Y. J. and T. H. Liu (1965): "On the Shortest Arborescence of a Directed Graph," *Science Sinica*, 14, 1396–1400.

Dutta, B. and A. Kar (2004): "Cost Monotonicity, Consistency and Minimum Cost Spanning Tree Games," *Games and Economic Behavior*, 48, 223–248.

EDMONDS, J. (1967): “Optimum Branchings,” *Journal of Reseseach National Bureau Standards*, 71B, 233–240.

FELTKAMP, V., S. MUTO, AND S. TIJS (1994): “On the Irreducible Core and the Equal Remaining Obligations Rule of Minimum Cost Spanning Extension Problems,” Working Paper, Tilburg University (CentER DP).

HART, S. AND A.MAS-COLELL (1989): “ Potential, Value and Consistency ”,*Econometrica*, 57, 589-614.

HERZOG, S., S.SHENKER, AND D.ESTRIN (1997): “Sharing the Cost of Multicast Trees: An Axiomatic Analysis,” *IEEE/ACM Transactions on Networking*, 847–860.

KAR, A. (2002): “Axiomatization of the Shapley Value on Minimum Cost Spanning Tree Games,” *Games and Economic Behavior*, 38, 265–277.

NORDE, H., S. MORETTIE, AND S. TIJS (2001): “Minimum Cost Spanning Tree Games and Population Monotonic Allocation Schemes,” *European Journal of Operational Research*, 154, 84–97.

SHAPLEY, L. (1971): “Cores of Convex Games,” *International Journal of Game Theory*, 1, 11–26.