The Iterative Minimum Cost Spanning Tree Problem

MSc Thesis *(Afstudeerscriptie)*

written by

Femke Bekius
(born November 20th, 1987 in Amsterdam, The Netherlands)

under the supervision of **Dr. Ulle Endriss**, and submitted to the Board of Examiners in partial fulfillment of the requirements for the degree of

MSc in Logic

at the *Universiteit van Amsterdam.*

Date of the public defense:	**Members of the Thesis Committee:**
September 25th, 2013	Prof.dr Krzysztof Apt
	Dr. Olivier Cailloux
	Dr. Ulle Endriss
	Prof.dr Benedikt Löwe
	Prof.dr Guido Schäfer

INSTITUTE FOR LOGIC, LANGUAGE AND COMPUTATION

Abstract

The minimum cost spanning tree problem consists in constructing a network of minimum cost that connects all agents to the source and distributes the cost among the agents in a fair way. We develop a framework for the *iterative* minimum cost spanning tree problem. In the iterative setting, agents arrive over time and desire to be connected to a source in different rounds in order to receive a service from the source. We provide an algorithm for the iterative minimum cost spanning tree problem in order to connect the agents from the different rounds to the source in a minimal way. Moreover, we discuss the complexity of the algorithm. To divide the cost of the constructed network among the agents in a fair way we propose different charge rules. One class of charge rules is defined in such a way that the inefficiency of the network, caused by agents joining in different rounds, is equally divided among the agents who use the network. A second class of charge rules charges the incoming agents as much as possible such that previously connected agents can be reimbursed. However, we want to avoid that agents are better off by construction their own network. Furthermore, we prove that the charge rules satisfy several properties. This provides the basis for comparing the charge rules and allows for assessment of their fairness in a particular situation.

Acknowledgements

First and foremost, I would like to thank my supervisor Ulle Endriss. You were the one who pointed me on the subject which was a great move. I was interested from the beginning and became more enthusiastic along the way. Your comments made me think of new concepts and directions which led to the final stage as it is now. Moreover, your support and flexibility made it a pleasure to have you as a supervisor.

I would also like to thank my thesis committee: Krzysztof Apt, Olivier Cailloux, Benedikt Löwe and Guido Schäfer, for their time reading my thesis and useful comments and questions.

Thanks to all MoL-students who were sitting with me till late in the MoL-room, and forced me to think of something else during lunch or dinner break. Special thanks to Anthony, Aybüke, Dieuwke, Heleen, Hugo and Sebastian for all their advices, suggestions and being there with and for me.

Aino, Annike, Anouk, Babette, Jitske, Merel, Nina, Philine en Sanne dank voor jullie lieve berichtjes, telefoontjes, etentjes en kopjes koffie! Nienke en Jolien bedankt voor het lezen van mijn introductie en het rotsvaste vertrouwen. Pap en mam, dank voor het altijd achter me staan, niet alleen tijdens het schrijven van mijn scriptie, maar gedurende de gehele master.

Contents

Chapter 1

Introduction

Optimizing a situation in order to keep costs as low as possible and fairly dividing these costs among the relevant agents is common to many real-life situations.

Consider the following situation: a group of people who live in different parts of the country and work at the same place usually all drive in their own car. It would reduce the daily travel costs and be more environmentally friendly if they would share the cars and come to work together. They wonder what the optimal route for them to drive would be and discuss the following issues among each other: whose car(s) should be used; who should share a car with whom; and who should be picked up first, second, etc. Furthermore, they wonder how to divide the cost of the drive among them in a fair way. In order to decide this different factors can be taken into consideration. Some people will drive in the car longer than others, some cars will be completely filled while others will have empty places left. Moreover, some people will take their own car and share it with other people and others will leave their car at home.

There are multiple possibilities for providing a driving schedule in this situation. Different disciplines can be consulted in order to find an *optimal solution* which minimizes the total cost and fairly divides it.

Providing an algorithm in an efficient way, for constructing an optimal driving scheme in the example above, lies in the field of operations research. Furthermore, in this area the computational complexity of these algorithms is of interest. In the economics literature the focus lies on aspects such as network cost sharing and mechanisms that try to explain how networks form. The discipline is closely related to game theory which provides tools that are particularly useful for allocating costs. Game theory can be seen as part of both operations research and economics: it provides a link between both research fields (Bergantiños and Lorenzo, 2004). (Computational) Social Choice Theory is concerned with fair division of goods between several agents. Above, the total cost for driving the people to their destination has to be divided in a fair way among them. In doing so we have to take into consideration the different situations of the people, e.g., the total time each one is driving the car.

The studies conducted in this thesis are concerned with a problem which lies on the boundary of the disciplines listed before, the Minimum Cost Spanning Tree (mcst) problem. Since the mcst problem interfaces with several disciplines it can be called an *interdisciplinary problem*. The mcst problem can be illustrated as follows: assume there is a group of agents located at different geographical points. They all desire a particular service which can only be provided by a common supplier, called the *source*. The agents are served through connections which entail some cost. They can be connected to the source *directly* or *indirectly* via another agent. Moreover, they do not care whether their connection is direct or indirect (Bergantiños and Vidal-Puga, 2007a). In the previous example, the cost of driving a car from one person to another can be determined and the source can be represented by their shared destination. Moreover, the agents do not care whether they have to pick up someone else or can travel directly to their work (Norde et al., 2004).

Assuming that the connection costs between the agents and the agents and the source are known, we would like to construct a network of minimum cost which connects all agents to the source, either directly or indirectly. Furthermore, given the cost of the network, we want to allocate the cost over the different agents in a fair way. The minimum cost spanning tree problem can thus be split into two sub-problems:

1. Constructing a network with minimum cost.

2. Dividing the cost of the network over the agents in a fair way.

Real-life situations in which the mcst problem arises abound. One can think of several towns which draw power from a common power plant (Bergantiños and Vidal-Puga, 2010), or houses which need to be connected to a water supply via pipelines (Bergantiños and Lorenzo, 2004, 2008a). If the houses are situated in the mountains an extra constraint has to be taken into account since the pipelines can only be constructed between a house and another if the second house is located strictly below the first one (Moretti et al., 2001). Another example can be found in the situation where the Russian natural gas producer Gazprom sends gas from Russia to Europe through the Ukraine. In order for Europe to reach the source of natural gas cheaply, Ukraine is used as a passing-through country and is compensated by transit fees from Gazprom (Trudeau, 2013a). Internet connections and cable TV are examples of devices/commodities served via network structures. The communication companies are dealing with the problem of constructing a network and dividing the total cost among the users of the service (Fernández et al., 2009). Furthermore, there are applications of the mcst problem in anthropology, biology and linguistics (Graham and Hell, 1985).

Usually it is assumed that the agents construct a minimum cost spanning tree together and share the cost among them in a specific way. In particular, it is assumed that all agents are there from the beginning. However, the following example illustrates that this need not always be the case.

Bergantiños and Lorenzo (2004) present an example which constitutes the point of departure for this thesis. They consider a situation which took place in the villages in an area called Ourense in Spain. Each village in the area has its own water supply. However, they do not always have a sufficient amount of water during summer. The inhabitants of the villages informed the valley authority about the situation who then took action. It built a dam and constructed pipelines such that each village was connected to the dam. Moreover, it provided a water deposit for each village. The corresponding costs were covered by the valley authority. Access to the water was free, the only cost the inhabitants of the villages had to cover were the cost of the pipelines which connected the individual houses to the water supply, either directly or indirectly via another house. After the inhabitants of the villages were informed about the procedure the following scenarios took place:

- People who lived close to the water supply decided to connect immediately and paid the cost of the created pipelines.
- Some people started to talk to each other and agreed on cooperating. They shared the cost of the pipelines which were used to connect all of them to the source.
- Some people decided not to connect to the source. They found it too expensive and said they did not need the extra water.

For the people wished to connect to the source, the valley authority started constructing the pipelines. After some time, when the system was seen to perform well, other people who were not yet connected decided that they wanted to be connected to the water supply after all. They argued that at that moment the connection cost for them were lower than before. This was the case since they could now connect via one of the previously connected houses. However, the previously connected people started to complain since they covered the cost for the pipelines from their houses to the water supply before and other people would now benefit from them. They argued that the new people should be charged for this. The valley authority replied by saying that connecting via previously connected houses is allowed since the pipelines are property of the valley authority.[1]

[1] Bergantiños and Lorenzo (2004) believe that the valley authority did not expect new houses wanting to be connected to the water supply at a later time.

Situations as the one in Ourense give rise to the subject for this thesis: we aim to provide a solution for the *iterative* minimum cost spanning tree problem. The problem deals with situations in which agents arrive over time and desire to be connected to the source in different rounds.[2] This is different from the *classical* minimum cost spanning tree problem where all agents arrive at the same time.

There are different reasons why agents do not always connect to the source at the same time. The main reason lies in the fact that the costs for connecting to the source are simply too high. Agents might hope that costs decrease over time, e.g., because of new technologies are developed, or because, as in the previous example, other agents connect earlier. Another reason why not all agents connect to the source at the same time might consist in it being physically impossible. For example, in a situation where new houses are built these have to be connected at a later stage, since they were simply not there in the first connecting round(s).

1.1 Approach

In the literature different approaches for finding a solution to the minimum cost spanning tree problem have been pursued. Two main approaches, the cooperative and non-cooperative game approach, can be distinguished. The former assumes that people have the intention to cooperate if they can be better off by doing so. The latter assumes that people do not cooperate and make their decisions independently. Bergantiños and Lorenzo (2004) argue that the mcst problem of the villages in Ourense calls for a non-cooperative game approach. The fact that some people decided not to connect to the water supply in the beginning made them wonder whether this was inefficiency on the side of the agents or whether there was some underlying rationale behind their behavior. Assuming that agents are not inefficient but that it is just impossible for them to be connected to the water supply immediately allows for adhering to the cooperative game approach. In this thesis, we will, as opposed to Bergantiños and Lorenzo, focus on the cooperative game approach.[3]

An implicit assumption in the literature on the minimum cost spanning tree problem is that agents want to be connected to the source, even if they

[2]The concept 'iterative' is related to the notion of 'online' in cooperative cost sharing games: players arrive over time and reveal their input data, e.g., cost of the edges, to agents who arrived earlier only at the moment of arrival (Brenner and Schäfer, 2010). See Section 3.1.2 for a brief discussion of the online model.

[3]Bird (1976), Claus and Kleitman (1973) and Granot and Huberman (1981) were among the first who associated the mcst problem with (cooperative) game theory. Aarts (1994) gives an introduction to the mcst problem associated with game theory. For a non-cooperative game approach we refer to Bergantiños and Lorenzo (2004, 2008a), Fernández et al. (2009) and Bergantiños and Vidal-Puga (2010).

have to pay the highest possible cost (Bergantiños and Vidal-Puga, 2007a; Bergantiños and Lorenzo, 2004). In addition, there is a planner who decides the cost allocation. Typically, the planner is interested in a fair distribution (Bergantiños and Vidal-Puga, 2010). Furthermore, the question of allocating costs is generally studied in the context of complete information (Sharkey, 1995). Hence, it is known which agents connected to the source before and which agents desire to be connected at a later stage. In addition, we assume that the agents in one round know the connection cost between themselves, between themselves and the source, and between themselves and the agents who connected before. Moreover, we make the assumption that the time at which agents desire to be connected to the source is fixed and that it is impossible to join earlier, although an agent may artificially delay this arrival.

The goal of this thesis is studying the iterative mcst problem and providing solutions in line with a cooperative game theoretic approach.

1.2 Outline

The structure of the thesis is as follows: in Chapter 2 we discuss the classical mcst problem and provide an overview of the existing literature on this problem. Two algorithms for constructing the network are presented and five different charge rules for allocating the cost over the agents are explained. Moreover, we list several properties which can be satisfied by the charge rules and provide an overview of which properties are satisfied by the rules presented. In Chapter 3 we introduce the framework for the iterative mcst problem. We present the algorithm for constructing the network and determine the complexity of the algorithm. Furthermore, we adapt the classical properties presented in Chapter 2 and introduce new properties, so-called *iterative properties*. In Chapter 4 we describe a particular approach, the *fair sharing of inefficiencies* approach, for allocating the cost of the network constructed in the iterative mcst problem. This approach leads to four different charge rules which will be evaluated according to the properties they satisfy. In Chapter 5 we describe a second approach, called the *reimbursement of previously connected agents* approach. This approach leads to a charge rule which will be evaluated according to the properties that are satisfied by it as well. In both chapters we discuss whether the rules satisfy more properties in special network structures which are particularly interesting in the iterative mcst problem. In Chapter 6 we conclude the thesis and provide directions for future work. Moreover, we compare the charge rules introduced in Chapters 4 and 5.

Chapter 2

The classical mcst problem

This chapter is an introduction to the classical minimum cost spanning tree problem. Furthermore, it gives an overview of a large part of the literature existing in this field. Section 2.1 starts with a brief introduction to cooperative game theory and graph theory. The necessary concepts used in the thesis will be presented. In addition, the framework for the classical mcst problem will be introduced in this section. In Section 2.2, the first part of the classical mcst problem, constructing a network of minimum cost which connects all agents to the source, will be considered. We introduce different algorithms and, moreover, state the complexity of these algorithms. Section 2.3 deals with the second part of the mcst problem: dividing the total cost of the constructed network over the agents in a fair way. In the literature several charge rules, also called cost allocation rules, for the classical mcst problem are defined and discussed. We present an overview of the most important ones regarding the cooperative game theory approach. Whether the charge rule provides a fair allocation over the agents of the cost of the constructed network can be evaluated by considering the properties satisfied by the charge rule. Section 2.4 consists of a selection of various properties discussed in the literature. Moreover, we divide them into different classes. Finally, we present an overview of which properties are satisfied by the charge rules introduced in Section 2.3.

2.1 Framework

This section consists of three subsections. In Section 2.1.1 some preliminaries in cooperative game theory are presented. In Section 2.1.2 some preliminaries in graph theory are given and in the Section 2.1.3 the framework for the classical minimum cost spanning tree problem is introduced. The readers familiar with the basics of cooperative game theory and graph theory can skip the first two subsections and are suggested to continue with Section 2.1.3.

2.1.1 Cooperative game theoretic preliminaries

A *cooperative game*, referred as a transferable utility (TU) game,[1] is defined to be a pair (N, v) consisting of a set of agents $N = \{1, \ldots, n\}$ and a utility function $v : 2^N \to \mathbb{R}_+$, which assigns a value to every coalition of players $S \subseteq 2^N$. We require that $v(\emptyset) = 0$ and in general $v(S)$ can be interpreted as the best outcome that a coalition S can achieve. A utility function is *monotone* if for all $S \subseteq T \subseteq N$.

$$v(S) \leq v(T).$$

A utility function is *superadditive* if for all $S, T \subseteq N$ with $S \cap T = \emptyset$ we have

$$v(S) + v(T) \leq v(S \cup T).$$

In other words, a pair of disjoint coalitions is best off when merging into one bigger coalition. To be more precise, the utility of the union of two coalitions is bigger than or equal to the sum of the utility of the separate coalitions (Airiau, 2012).

A *solution* of a TU game (N, v) is a set of vectors $x \in \mathbb{R}^{|N|}$, also called *payment vectors*, which assigns a payoff x_i to every agent $i \in N$ (Sharkey, 1995). Different solution concepts are proposed in the literature (Granot and Huberman, 1981). We will present two concepts, the core and the Shapley value, which will be applied in the charge rules we define in Section 2.3.

The *core* of the TU game (N, v) is defined as follows:

$$core(N, v) = \Big\{ x \in \mathbb{R}^{|N|} \mid \sum_{i \in N} x_i = v(N) \text{ and } \sum_{i \in S} x_i \geq v(S), \text{ for all } S \subseteq N \Big\}.$$

A solution is in the core if no coalition S can gain by forming a different coalition, i.e., no agent is better off by forming another coalition than his current one which we assume to be the grand coalition N.

Given a finite set N, let Π_N denote the set of all permutations of N. Given a permutation $\pi \in \Pi_N$, let $Pre(i, \pi)$ denote the set of elements of N which come before i in the order given by π, i.e.,

$$Pre(i, \pi) = \{ j \in N \mid \pi(j) < \pi(i) \}.$$

The *Shapley value*, first defined by Shapley (1953b), for agent $i \in N$ of a TU game (N, v) is defined as

$$Sh_i(N, v) = \frac{1}{n!} \sum_{\pi \in \Pi_N} \Big(v(Pre(i, \pi) \cup \{i\}) - v(Pre(i, \pi)) \Big).$$

[1] Notice that cooperative game theory is also considered with non-transferable utility games, but this is not of interest here.

Equivalently,

$$Sh_i(N,v) = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(n-|S|-1)!}{n!} \Big(v(S \cup \{i\}) - v(S) \Big).^2$$

The Shapley value averages each agent's payoff over all possible orderings. The value of an agent i in a coalition S is the average marginal value over all possible orders in which the agents may join the coalition (Airiau, 2012).

A TU game (N, v) is *concave* if, for all $S, T \subseteq N$ and $i \in N$ such that $S \subseteq T$ and $i \notin T$,

$$v(S \cup \{i\}) - v(S) \geq v(T \cup \{i\}) - v(T).$$

The following theorem links the two solution concepts defined above given that the cooperative game is concave.

Theorem 1. *The Shapley value belongs to the core if the cooperative game is concave (Shapley, 1953a).*

2.1.2 Graph theoretic preliminaries

A *graph* $G = (V, E)$ consists of a set of vertices V, also called nodes, and a set of edges E, also called links. A *weighted graph* is a graph with numerical labels, called *weights*, on the edges.

In the thesis we will only consider *simple graphs*, i.e., graphs having no loops or multiple edges. Thus, $E \subseteq \{(i,j) \mid i,j \in V \text{ and } i \neq j\}$. Moreover, the graphs are undirected, i.e., if $(i,j) \in E$ then also $(j,i) \in E$, and the edge weights are nonnegative.

We call a graph *complete* when all its vertices are pairwise adjacent, thus $E = \{(i,j) \mid i,j \in V \text{ and } i \neq j\}$. A *tree*[3] is a connected graph T that does not contain any cycles. Notice that a tree with n vertices has $n-1$ edges.

Given a graph $G = (V, E)$ and a pair of different vertices $i, j \in V$, a *path* from i to j in G is a sequence of different edges $\{(i_{s-1}, i_s)\}_{s=1}^p$ that satisfy $(i_{s-1}, i_s) \in E$ for all $s \in \{1, \ldots, p\}$, $i_0 = i$ and $i_p = j$. Two vertices $i, j \in V$ are *connected* in G, if there is a path in G from i to j. If $(i,j) \in E$ we say that i and j are *directly connected* in G.

An *adjacency matrix* $A = (a_{ij})_{i,j \in V}$ for a simple graph $G = (V, E)$ with $|V| = n$ is an $n \times n$ matrix where $a_{ij} = 1$ represents that vertices i and j are connected in G. If vertices i and j are not connected, then $a_{ij} = 0$. Notice that every adjacency matrix is symmetric, i.e., $a_{ij} = a_{ji}$ (West, 2000).

[2] In general, computing the Shapley value is not efficient, i.e., not possible in polynomial time in the number of agents. We refer to Airiau (2012) for some representations of the game which allow for computing the Shapley value in an efficient way.

[3] In this thesis we will use the terms *tree* and *network* interchangeably.

2.1.3 Minimum cost spanning tree problem

In this section the minimum cost spanning tree problem is introduced. As presented in Chapter 1, the mcst problem consist of two sub-problems:

1. Constructing a network of minimum cost connecting the prospective users.

2. Dividing the cost of the network in a fair way over these users.

Before stating the mcst problem formally we introduce some concepts.

Let $\mathcal{N} = \{1, 2, \dots\}$ be the set of possible agents and $N = \{1, \dots, n\} \subset \mathcal{N}$ is a finite set of *agents* who desire to be connected to the source. In graph theoretical terms, this is called the set of vertices. The *source* is a special vertex, denoted by 0. The constructed graphs in the mcst problem have vertices from the set $N_0 = N \cup \{0\}$. Our interest lies in graphs where each vertex in N is connected to the source, either directly or indirectly. The set of all graphs over N_0 is denoted by $\mathcal{G}^N$ and the set of all graphs over N_0 such that every vertex is connected to the source is denoted as $\mathcal{G}_0^N$.

An agent i is connected to the source if there exist a path from i to the source. We call an edge $(0, i)$ a *direct edge between agent i and the source.* For two agents i and j we say that they are connected if there is a path between i and j which does *not* include the source.

Given a weighted graph $G = (N_0, E)$,[4] where the weight of an edge represents its cost, we define a *cost matrix* $C = (c_{ij})_{i,j \in N_0}$. This matrix is similar to the adjacency matrix. In this case, entry c_{ij} corresponds to the cost of edge (i, j) in G. Given a cost matrix C, we assume $c_{ij} = c_{ji} \geq 0$ for all $i, j \in N_0$ and $c_{ii} = 0$ for all $i \in N_0$. Since $c_{ij} = c_{ji}$ the graph consists of undirected edges, thus $(i, j) = (j, i)$. The set of all cost matrices over N is denoted as $\mathcal{C}^N$. Cost matrices are thus nonnegative,[5] symmetric matrices of order $(n + 1) \times (n + 1)$ (Kar, 2002). Given two cost matrices C and C' we say that $C \leq C'$ if $c_{ij} \leq c'_{ij}$ for all $i, j \in N_0$ (Bergantiños and Vidal-Puga, 2007a). Given a cost matrix C for graph $G = (N, E)$, the *restricted* cost matrix for a subset $S \subseteq N$ is denoted by $C|_S$ and consist of the entries c_{ij} such that $i, j \in S$.

A *minimum cost spanning tree problem*, abbreviated as mcst problem, is a pair (N_0, C) where N is a finite set of agents, 0 is the source, and $C \in \mathcal{C}^{N_0}$ is the corresponding cost matrix. Given a mcst problem (N_0, C), the mcst problem induced by C for $S \subseteq N$ is $(S_0, C|_{S_0})$, denoted by (S_0, C). Given a mcst problem (N_0, C) and a weighted graph $G = (N_0, E) \in \mathcal{G}_0^N$ we define

[4]Considering the mcst problem we will always mean *weighted graphs* when we talk about graphs.

[5]Recall that we assume the cost of the edges to be nonnegative, which is a standard assumption in the literture on mcst problems.

the *cost* associated with G as

$$c(N_0, C, G) = \sum_{(i,j)\in E} c_{ij}.$$

We write $c(G)$ instead of $c(N_0, C, G)$ when there is no ambiguity.

A *minimum cost spanning tree* for (N_0, C), abbreviated to an *mt*, is a graph $T \in \mathcal{G}_0^N$ such that $c(T) = \min_{G\in\mathcal{G}_0^N} c(G)$. Notice that a graph T which connects all agents to the source in a minimal way will always be a tree. If the graph contains a cycle we can always remove one edge of this cycle such that all agents are still connected to the source. It has been proven, among others by Prim and Kruskal, that such an mt always exists. However, it does not have to be unique.[6] Given a mcst problem (N_0, C) we denote the cost associated with any mt in (N_0, C) as $m(N_0, C)$.

Bird (1976) was the first who associated a cooperative game (N, v_C) with a mcst problem (N_0, C). This is called a *mcst game* where $v_C(S) = m(S_0, C)$ for each $S \subseteq N$ and $v_C(\emptyset) = 0$. The corresponding *Shapley value* for agent i of the game (N, v_C) equals,

$$Sh_i(N, v_C) = \sum_{S\subseteq N\setminus\{i\}} \frac{|S|!(n-|S|-1)!}{n!}\Big(m((S\cup\{i\})_0, C) - m(S_0, C)\Big).$$

Since the Shapley value satisfies efficiency, i.e., $\sum_{i\in N} Sh_i(N, v) = v(N)$ (Driessen, 1988), the sum over all agents $i \in N$ of $Sh_i(N, v_C)$ equals the minimum cost of any mt in (N_0, C), i.e., $m(N_0, C)$. The Shapley value divides equally the cost of an edge e between all agents who need this edge (Moulin, 2013).

Granot and Huberman (1981) showed that the core of a mcst game is never empty. It is thus possible to find a stable allocation for the mcst problem, i.e., no group of agents can be better off by forming a different coalition. For a proof of the non-emptiness of the core see the lecture notes of Airiau (2012).

2.2 Algorithms

Given a minimum cost spanning tree problem (N_0, C), a tree with minimum cost can be constructed by different algorithms. Prim's algorithm (Prim, 1957) and Kruskal's algorithm (Kruskal, 1956) are most commonly used algorithms in the literature. For example, for defining a charge rule for the mcst problem, Prim's algorithm is used by Bird (1976) and Dutta and Kar (2004). Kruskal's algorithm is used by Moretti et al. (2005), Tijs et al. (2006), Brânzei et al. (2004) and Feltkamp et al. (1994). Both algorithms are *greedy algorithms*: they make a locally optimal choice at each stage and hope to

[6] We refer to the Section 2.2 for an explanation.

output an optimal solution in the end. Both Prim's and Kruskal's algorithm actually do output an optimal solution for the mcst problem. In this section we will present these two algorithms and state their complexity.

For an explanation of the algorithms we follow Çiftçi and Tijs (2007). Prim's algorithm is vertex-oriented and can be described as follows: in every iteration of the algorithm, an agent who is not yet connected to the source constructs an edge between her and the source or one of the agents connected in a previous step of the algorithm. The not yet connected agent with the cheapest edge to the source or to one of the previously connected agents is the one who may construct the edge. The algorithm is formally presented in Algorithm 1.

The set S denotes the set of vertices which are connected to the source (consisting of j's). The set E denotes the edges which are used to connect the vertices in S to the source. Agent j_i^* represents the agent in S to who agent i can connect in the cheapest way, thus j^* is an agent who connected in a previous round or it is the source. In the initial step j^* represents the source 0. Agent i^* is then the agent who has the cheapest connection cost to j_i^* over all agents i who are not connected to the source yet.

Algorithm 1 Prim's algorithm

Input: N vertices, source 0 and cost matrix C.
Ouput: An mt tree T that connects all vertices in N to 0: $T := (S, E)$.

```
S ← {0}, E ← ∅                                    # Initialization
for i ∈ N do
    j_i* ← 0
end for

while N ≠ ∅ do
    i* ← arg min_{i∈N\S} c_{i j_i*}
    S ← S ∪ {i*}                                  #Add new agent to graph
    E ← E ∪ (i*, j_{i*}*)                          #Add new edge to graph
    N ← N \ {i*}
    for i ∈ N do
        if c_{ii*} < c_{i j_i*} then
            j_i* ← i*                             #Update costs
        end if
    end for
end while
```

Theorem 2 states that the tree constructed by Prim's algorithm is a minimum cost spanning tree, i.e., it connects all the agents to the source with minimal cost. This can be proved by induction on the vertices of the tree and the idea is the following: first, prove that the edge e with least weight is included in some minimum tree. Second, consider a graph $G' := G/e$, which is the graph resulting from the contraction of edge e. There is a one-to-one correspondence between the set of spanning graphs of G and the set of spanning graphs of G'. Hence, it suffices to show that $T' := T/e$ is an minimum cost spanning tree of G'. We refer to Bondy and Murty (2008, p. 147) for a complete proof.

Theorem 2. *Every graph constructed by Prim's algorithm is a minimum cost spanning tree.*

On the other hand, Kruskal's algorithm is edge oriented. It first orders the edges in increasing order according to their cost. Then, the cheapest edge is selected and added to the spanning tree. Followed by the second cheapest edge, etc. This continues in such a way that an edge is only added if it does not create a cycle with the previously added edges. Another way of saying it is that an edge is only added if it connects two different components. The algorithm terminates when the graph consist of one component, i.e., all vertices in N are connected to 0. Moreover, the number of edges then equals $|N|$. The algorithm is formally presented in Algorithm 2.

The set E denotes the set of edges which are added to the tree and A counts the number of edges in the tree. The list L contains all possible edges ordered by increasing weight. We use the operations HEAD(L) and TAIL(L) to make sure that each time we check the edge with lowest cost that is not checked before. The following operations are used to verify whether the vertices are contained in different components of the graph. The operation MAKE-SET(i) creates a set i whose only member is the vertex i. The operation FIND-SET(i) returns a pointer to the set which contains i. The operation UNION(i, j) creates a new set by taking the union of the set which contains i and the set which contains j.

Theorem 3 states that Kruskal's algorithm constructs a spanning tree with minimum cost. The idea of the proof is as follows: consider that the algorithm indeed outputs a tree T and prove that this tree is a spanning tree of minimum weight T^*. As long as $T \neq T^*$ consider an edge e in T which is not in T^* and build a minimum spanning tree that completely agrees with T. We refer to Theorem 2.2.3 of West (2000, p. 96) for a complete proof.

Theorem 3. *Every graph constructed by Kruskal's algorithm is a minimum cost spanning tree.*

Remark 1. Notice that when all the weights, i.e., costs, on the edges are different in a given complete graph, then both Prim's and Kruskal's algorithm output the same and unique minimum cost spanning tree. Whenever two edges have the same weight, the minimum cost spanning tree does not have to be unique. It is possible that two different edges can be added to the tree at some point in the algorithm, and selecting the one or the other might result in a different spanning tree in the end. The algorithms therefore might output a different tree. However, the cost of the trees are the same.

Algorithm 2 Kruskal's algorithm

Input: N vertices, source 0 and cost matrix C.
Ouput: An mt tree T that connects all vertices in N to 0: $T = (N_0, E)$.

```
E ← ∅, A ← 0                                         # Initialization
for i ∈ N_0 do
  MAKE-SET(i)
end for

for i, j ∈ N_0 do
  L ← Ordering of edges by increasing weight.
end for
while A ≠ |N| do
  (i, j) ← HEAD(L)                       #Pick the edge with lowest weight
  L ← TAIL(L)                                  #Remove edge from list
  if FIND-SET(i) ≠ FIND-SET(j) then
    UNION(i, j)                      #Put vertices i and j in the same set
    E ← E ∪ (i, j)                               #Add new edge to graph
    A ← A + 1                               #Count the number of edges
  end if
end while
```

While the upper bound on the complexity of Prim's algorithms is $O(|V|^2)$, the upper bound is $O(|E| \log |V|)$ for Kruskal's algorithm (Graham and Hell, 1985). If we have more sophisticated data structures, i.e., a particular ordering of the input, then the complexity can be lower. Martel (2002), Campos and Ricardo (2008) and Fredman and Tarjan (1987) give lower complexity bounds for Prim's algorithm. Furthermore, Graham and Hell (1985) give lower complexity bounds for Kruskal's algorithm. When the edges are pre-sorted by weight Prim's and Kruskal's algorithm have similar running times (West, 2000).

Other algorithms which provide a minimum cost spanning tree are the Boruvka algorithm (Bergantiños and Vidal-Puga, 2011), the V-algorithm (Çiftçi and Tijs, 2007) and the Subtraction algorithm (Norde et al., 2004). Graham and Hell (1985) give an overview of the evaluation of the algorithms for the mcst problem of which Boruvka's algorithm was actually the starting point (Boruvka, 1926).

Now that we have constructed a spanning tree of minimum cost we solved the first sub-problem of the mcst problem. In the next section we will explore the second part of the problem, i.e., the distribution of the cost of the tree over the agents in a fair way.

2.3 Charge rules

In the literature various charge rules are defined. The aim of these charge rules is to define a fair allocation in order to divide the cost of the constructed tree over the agents in N. Charge rules are generally designed according to a cooperative or non-cooperative game approach. Since in this thesis we focus on a cooperative game approach we will consider charge rules that use concepts from cooperative game theory. The following definition provides the formal definition of a charge rule (Bergantiños and Vidal-Puga, 2007a).

Definition 1 (Charge rule)**.**
Given any mcst problem (N_0, C) a charge rule is a function y mapping the mcst problem to a payment vector,

$$y(N_0, C) \in \mathbb{R}^{|N|}$$

such that $\sum_{i \in N} y_i(N_0, C) = m(N_0, C)$,[7] where $y_i(N_0, C)$ denotes the cost allocated to agent i.

We next introduce five different charge rules. In the literature on the mcst problem these charge rules are the ones to which is referred the most and we therefore consider them as the most important.

Bird rule

The Bird rule (Bird, 1976), denoted by B, is defined through Prim's algorithm. The agents connect sequentially to the source following Prim's algorithm. Each agent then pays the cost of the edge which connects him to the source, either directly or indirectly, in the constructed tree.

The Bird rule is defined for two instances. First assume that there exists an unique mt T. Given $i \in N$, let i^0 be the first node in the unique path in T from i to the source. The Bird rule is then defined for each $i \in N$ as

$$B_i(N_0, C) = c_{i^0 i}.$$

Thus, the rule assigns to agent i the cost of the edge connecting him to its immediate predecessor in the minimum cost spanning tree. Secondly, assume that there is more than one mt T. In this case the Bird rule can be defined as the average over the trees associated with Prim's algorithm (Bergantiños and Vidal-Puga, 2007a). Dutta and Kar (2004) defined this as follows: given a permutation $\pi \in \Pi_N$, $B^\pi(N_0, C)$ is the allocation obtained by using Prim's algorithm. Possible ties are broken by selecting the first agent in the ordering of π. Formally,

$$B(N_0, C) = \frac{1}{n!} \sum_{\pi \in \Pi_N} B^\pi(N_0, C).$$

[7]This condition is called budget balance (BB), see Section 2.4.

Notice that the Bird rule depends on the minimum cost spanning tree constructed by the algorithm. Since no one will pay more than the cost of her direct edge this rule satisfies core stability (CS), i.e., no one is better off by constructing their own network.[8]

Example 1. Let $N = \{1, 2, 3\}$ and let the complete graph be given in Figure 2.1.

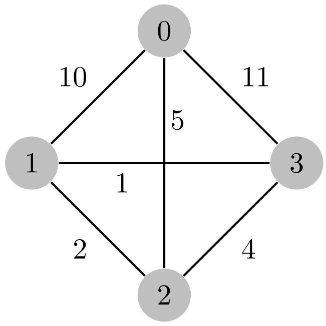

Figure 2.1: Complete graph with cost matrix C.

The Bird rule gives the following allocation vector: $B(N_0, C) = (2, 5, 1)$.

Kar rule

The Kar rule (Kar, 2002), denoted by K, is defined to be the Shapley value of the game (N, v_C), i.e.,

$$K(N_0, C) = Sh(N, v_C).$$

Notice that, unlike the Bird rule, the Kar rule is independent of the tree constructed by the algorithm. The charge rule only depends on the cost matrix C. This can be seen as an advantage since the algorithm can output more than one tree, but the rule will in all cases give the same allocation vector. The aim for defining the Kar rule was because the rule does satisfy cost monotonicity (CM), i.e., a decrease in the cost of an edge cannot harm the adjacent agents. Different from the Bird rule, which does not satisfy CM, the Kar rule does not satisfy CS. Another disadvantage is that the Kar rule allows for negative cost shares. However, depending on the situation, negative cost shares can be accepted (Trudeau, 2012, 2013a). An axiomatic characterization of the Kar rule can be found in the work of Kar (2002) or Trudeau (2013a).

Example 2. Given the mcst problem as presented in Figure 2.1, the Kar rule gives the following allocation vector: $K(N_0, C) = (\frac{20}{6}, \frac{-1}{6}, \frac{29}{6})$.

[8]We refer to Section 2.4 for the formal definition of CS.

Dutta-Kar rule

The Dutta-Kar rule (Dutta and Kar, 2004), denoted by DK, is similar to the Bird rule, defined through Prim's algorithm (Bergantiños and Vidal-Puga, 2007a). After each step of the algorithm the charge rule decides the cost allocation for the agent who is connected to the source in the previous step. Let t^k be the maximum of the cost among all edges constructed in previous steps 1 to $k-1$. Formally, let $t^0 = 0$ and $t^k = \max(t^{k-1}, c_{a^k b^k})$, where (a^k, b^k) is the edge selected by Prim's algorithm in step k. Moreover, b^k is the agent who is connected in round k. Then,

$$DK_{b^{k-1}}(N_0, C) = \min(t^{k-1}, c_{a^k b^k}),$$

and

$$DK_{b^n}(N_0, C) = t^n.$$

Notice that the cost allocation of the agent added to the tree in round $k-1$, i.e., b^{k-1}, is decided at step k. If in any step there is more than one possibility for an edge $(a^k b^k)$, then let π be a strict permutation N and use π as a tie-breaking rule.[9] Let Π_N be the set of all strict permutations of N. Then, the cost allocation using the Dutta-Kar rule is obtained by taking the average of the cost allocations obtained for each permutation $\pi \in \Pi_N$. That is,

$$DK(N_0, C) = \frac{1}{n!} \sum_{\pi \in \Pi_N} DK^{\pi}(N_0, C).$$

Similar to the Bird rule, the Dutta-Kar rule depends on the constructed tree. The main purpose for defining the Dutta-Kar rule was that the rule satisfies both CS and CM. The first property is not always satisfied by the Kar rule and the second one is not always satisfied by the Bird rule (Dutta and Kar, 2004).

Example 3. Given the mcst problem as presented in Figure 2.1, following Prim's algorithm, we select first edge $(0, 2)$, then edge $(2, 1)$ and at last edge $(1, 3)$. This gives $t^0 = 0$, $t^1 = \max(0, 5)$, $t^2 = \max(5, 2)$ and $t^3 = \max(5, 1)$. The cost share for agent 2 then equals $\min(5, 2) = 2$, for agent 1 the cost share equals $\min(5, 1) = 1$ and for agent 3 the cost share equals $t^3 = 5$. Thus, the Dutta-Kar rule gives the following allocation vector: $DK(N_0, C) = (1, 2, 5)$.

Folk solution

The folk solution, denoted by φ, has been studied by several researchers and was invented under different names.[10]

[9] The rule selects the agent b^k which comes first in the ordering according to π.

[10] The following charge rules coincide with the folk solution: β^{π} (Bergantiños and Vidal-Puga, 2011), V-value (Çiftçi and Tijs, 2007), P-value (Brânzei et al., 2004), ERO-value (Feltkamp et al., 1994), and Shapley value of the (weighted) optimistic TU game (Bergantiños and Lorenzo, 2008b; Bergantiños and Vidal-Puga, 2007b).

The folk solution is the Shapley value of the game (N, v_{C^*}), where C^* is the *irreducible cost matrix*. The idea behind the irreducible cost matrix is to reduce the cost of each edge as much as possible, with the constraint that the total cost of connecting all agents to the source remains unchanged (Trudeau, 2013c). Formally, for any cost matrix C, we can define the irreducible cost matrix C^* for $i, j \in N_0$ by

$$c^*_{ij} = \min_{p_{ij} \in P_{ij}(N_0)} \left(\max_{e \in p_{ij}} (c_e) \right),$$

where p_{ij} denotes a path from i to j and $P_{ij}(N_0)$ denotes the set of all possible paths over N_0 between i and j (Trudeau, 2013a). Thus, for any edge (i, j) find the path between i and j for which the edge with highest cost has the lowest value and assign this value to entry c^*_{ij} in the irreducible matrix (Bergantiños and Vidal-Puga, 2007a).

A criticism leveled against the dependence of the folk solution on the irreducible cost matrix is that we lose information by reducing the cost matrix. Since we lower the cost of some of the edges it might be that edges with previously high cost now have lower cost and thus when the charge rule is applied this will not represent the real situation. Similar to the Kar rule the folk solution does not depend on the constructed tree. The main advantage of the folk solution is that it satisfies population monotonicity (PM), i.e., when more agents desire to be connected to the source the cost shares for the existing agents become lower. Bergantiños and Vidal-Puga (2007a) give an argument in favor of the folk solution by proving that any rule that does not depend on the irreducible form does not satisfy strict cost monotonicity (SCM), i.e., the requirement that a decrease of any cost of the edges should not harm any agent. Moreover, the folk solution is, in contrast to the Kar rule, computable in polynomial time (Bogomolnaia and Moulin, 2010).

A characterization of the folk solution can be found in the work of Trudeau (2013a), and Bogomolnaia and Moulin (2010) present a closed-form expression of the folk solution.

Example 4. Given the mcst problem as presented in Figure 2.1, the irreducible cost matrix C^* can be computed. This gives the complete graph in Figure 2.2.

Cycle-complete solution

The cycle-complete solution, denoted by CC, was devised in response to the critique of the folk solution. The charge rule selects a core allocation but throws away less information than the folk solution does (Trudeau, 2012). The cost matrix is defined in a way similar to the irreducible cost matrix, but uses cycles instead of paths. Given a graph $G = (V, E)$, for an edge $(i, j) \in E$, the solution searches for the cycle that goes through i and j and for which the

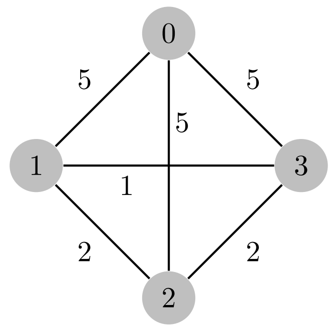

Figure 2.2: Complete graph with irreducible cost matrix C^*.

The folk solution gives the following allocation vector: $\varphi(N_0, C^*) = (\frac{5}{2}, 3, \frac{5}{2})$.

edge with the highest cost has the lowest value. If this value is smaller than the original connection cost c_{ij}, it will be assigned to edge (i, j). Formally, given an irreducible cost matrix C^* we can define the cycle-complete cost matrix $\overline{C}$ for $i, j \in N$ as follows:

$$\bar{c}_{ij} = \max_{k \in N \setminus \{i,j\}} \left(c_{ij}^{N_0 \setminus k} \right)^*$$

$$\bar{c}_{0i} = \max_{k \in N \setminus \{i\}} \left(c_{0i}^{N_0 \setminus k} \right)^*.$$

The cycle complete solution is then defined to be the Shapley value of the game $(N, v_{\overline{C}})$ (Trudeau, 2013c).

Compared to the folk solution less reductions of the cost of the edges take place in the cycle-complete solution. Therefore, the charge rule is more responsive to changes in costs and asymmetries than the folk solution. The cycle-complete solution is less responsive than the Kar rule since the Kar rule does not reduce any of the cost of the edges. However, unlike the Kar rule the cycle-complete solution satisfies CS and is computable in polynomial time (Bogomolnaia and Moulin, 2010; Ando, 2012).

Trudeau (2013c) provides characterizations of the cycle-complete and folk solution. By defining two different properties, of which one is satisfied by the cycle-complete and one satisfied by the folk solution, he makes clear that the two solutions differ in their approach.

Example 5. Given the same mcst problem as presented in Figure 2.1, the cycle-complete cost matrix $\overline{C}$ can be computed. This gives the complete graph in Figure 2.3.

The following overview shows the cost allocation vectors of the mcst problem presented in Figures 2.1, 2.2 and 2.3 for the Bird, Kar, Dutta-Kar, folk and cycle-complete solutions. The left column consist of rules using cost matrix

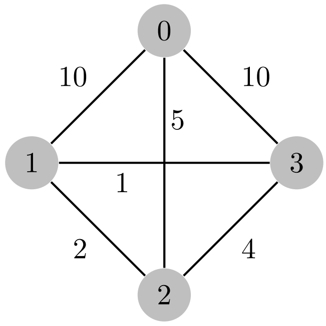

Figure 2.3: Complete graph with cycle-complete cost matrix $\overline{C}$.

The cycle-complete solution gives the following allocation vector: $CC(N_0, \overline{C}) = (\frac{21}{6}, 0, \frac{27}{6})$.

C, the middle column consist of the rule using the irreducible cost matrix C^* and the right column consist of the rule using the cycle-complete cost matrix $\overline{C}$. The allocations for the agents differ considering the different rules.

$B(N_0, C) = (2, 5, 1)$	$\varphi(N_0, C^*) = (\frac{5}{2}, 3, \frac{5}{2})$	$CC(N_0, \overline{C}) = (\frac{21}{6}, 0, \frac{27}{6})$
$DK(N_0, C) = (1, 2, 5)$		
$K(N_0, C) = (\frac{20}{6}, \frac{-1}{6}, \frac{29}{6})$		

Now that we discussed several charge rules the question is which one gives a fair allocation of the cost over the agents. In the next section we present different properties and give an overview of which properties are satisfied by the previously discussed charge rules.

2.4 Properties

We consider the properties satisfied by the charge rule in order to judge whether a certain charge rule is a fair rule. Since there are different interpretations of fairness and different papers discuss different situations, many properties have been invented. Some properties occur in the same form but with different names in different papers.

In the following sections we present several properties. Some of them had already been mentioned while describing the charge rules, but in these sections the formal definitions will be stated. To have a better overview we divided them in different classes depending on the purpose of the property. The idea of classifying the properties comes from Trudeau (2013b) who also uses the division of the properties in the following classes: stability, comparative and simplifying properties. The first class states some basic properties. The second class considers the stability of the problem, dealing with the concern that agents will not freely agree to cooperate. The third class consists of properties based on the comparison between different agents. The fourth one consists of properties which try to simplify the problem. In an additional class we list some properties which do not fit in any of the classes introduced before. The last section provides an overview of the properties satisfied by the charge rules discussed in Section 2.3.

2.4.1 Basic properties

Given a charge rule y we define the following basic properties. Among others they are standard assumptions in the literature on cost sharing mechanism design (Brenner and Schäfer, 2010; Tazari, 2005; van Zwam, 2005).

NPT No Positive Transfer:[11] an agent is not paid for receiving a connection to the source.
For all mcst problems (N_0, C), for all $i \in N$, we have

$$y_i(N_0, C) \geq 0.$$

BB Budget Balance:[12] the total cost share obtained from all agents is equal to the total cost.
For all mcst problems (N_0, C), we have

$$\sum_{i \in N} y_i(N_0, C) = m(N_0, C).$$

2.4.2 Stability properties

Given a charge rule y we define the following stability properties (Bergantiños and Vidal-Puga, 2008; Trudeau, 2013b).

CS Core Stability: no group of agents will be better off by constructing their own network instead of paying what the charge rule proposes for each of them.
For all mcst problems (N_0, C) and $S \subseteq N$, we have

$$\sum_{i \in S} y_i(N_0, C) \leq m(S_0, C).$$

PM Population Monotonicity: no agent is worse off with the entrance of new agents. In particular, this property prevents incentives to veto the entrance of new agents.
For all mcst problems (N_0, C), $S \subseteq N$ and $i \in S$, we have

$$y_i(N_0, C) \leq y_i(S_0, C).$$

[11] This property is also called *Positivity* (Kar, 2002; Bergantiños and Vidal-Puga, 2007a, 2008; Trudeau, 2013b).

[12] In some sources in the literature, the condition is called *Efficiency* (Kar, 2002; Angel et al., 2006).

Remark 2. PM implies CS (Trudeau, 2013b).[13]

2.4.3 Comparative properties

Given a charge rule y we define the following comparative properties (Bergantiños and Vidal-Puga, 2007a, 2008; Trudeau, 2013b).

CM Cost Monotonicity: a decrease in the cost of a link does not harm the adjacent agents. In particular, this property prevents the agents from taking advantage by reporting false connection costs.
For any two mcst problems (N_0, C) and (N_0, C') such that $c_{ij} < c'_{ij}$ for some $i \in N$ and $j \in N_0$, and $c_e = c'_e$ otherwise, we have

$$y_i(N_0, C) \leq y_i(N_0, C').$$

The following property is particularly interesting for the iterative mcst problem since in the iterative case the available connections to the source may change, depending on the the agents who joined before. Hence, the cost to the source might change. We therefore introduce the following property.

CM_0 Source Cost Monotonicity: a decrease in the cost of a link to the source does not harm any agent.
For any two mcst problems (N_0, C) and (N_0, C') such that $c_{0i} \leq c'_{0i}$ for all $i \in N$ and $c_e = c'_e$ otherwise, we have

$$y(N_0, C) \leq y(N_0, C').$$

SCM Strong Cost Monotonicity:[14] a decrease in the cost of a link does not harm any agent.
For any two mcst problems (N_0, C) and (N_0, C') such that $C \leq C'$, we have

$$y(N_0, C) \leq y(N_0, C').$$

Remark 3. SCM implies CM (Trudeau, 2013b) and SCM implies CM_0.

The following property focuses on the situation in which every agent has high connection cost to the source compared to the connection cost between agents. The optimal way to build the network is to connect one agent to the source directly and let the other agents connect to the source via this agent. Then, a charge rule satisfies the following property if it divides the cost of the link to source equally over the agents.

[13]Notice that the charge rule should satisfy BB in order for this to hold.

[14]This property is called *Solidarity* by Bergantiños and Vidal-Puga (2007a); Trudeau (2013b).

ESEC Equal Share of Extra Cost: given any two mcst problems (N_0, C) and (N_0, C'), and given $c_0, c_0' \geq 0$. Let $c_{0i} = c_0$ and $c_{0i}' = c_0'$ for all $i \in N$, $c_0 < c_0'$ and $c_{ij} = c_{ij}' \leq c_0$ for all $i, j \in N$, then for all $i \in N$, we have

$$y_i(N_0, C') = y_i(N_0, C) + \frac{c_0' - c_0}{|N|}.$$

IOC Independence of Other Cost: an agent's cost share depends only on his adjacent costs.
For any two mcst problems (N_0, C) and (N_0, C'), and all $i \in N$ such that $c_{ij} = c_{ij}'$ for all $j \in N_0 \setminus \{i\}$, we have

$$y_i(N_0, C) = y_i(N_0, C').$$

Remark 4. IOC is not satisfied by any charge rule discussed in Section 2.3 (Bergantiños and Vidal-Puga, 2007a).

The previously stated properties compared the cost allocation for one agent given two different mcst problems. The next three properties compare the situation between two agents given the same mcst problem.

SYM Symmetry: symmetric agents are treated equally.
For all mcst problems (N_0, C) and any pair of agents $i, j \in N$ such that for all $k \in N_0 \setminus \{i, j\}$ we have $c_{ik} = c_{jk}$, it is the case that

$$y_i(N_0, C) = y_j(N_0, C).$$

RNK Ranking: lower connection cost translates into lower cost shares.[15]
For all mcst problems (N_0, C), if $c_{ik} \leq c_{jk}$ for all $k \in N_0 \setminus \{i, j\}$, then we have

$$y_i(N_0, C) \leq y_j(N_0, C).$$

ET Equal Treatment: if the cost of a link changes, then the cost shares of the corresponding agents change by the same amount.
For any two mcst problems (N_0, C) and (N_0, C') and $i, j \in N_0$ such that $c_{ij} > c_{ij}'$ and $c_e = c_e'$ else, we have

$$y_i(N_0, C) - y_i(N_0, C') = y_j(N_0, C) - y_j(N_0, C').$$

A Anonymity: an allocation of the cost to the agents does not depend on their name.
For all mcst problems (N_0, C) and a permutation π of N_0 with $\pi(0) = 0$, we have

$$\pi\left(y(N_0, C)\right) = y(N_0, \pi C).$$

[15]Bogomolnaia and Moulin (2010) present several strict versions of this property.

IIT Independence of Irrelevant Trees: if two mcst problems (N_0, C) and (N_0, C') are *tree-equivalent*, i.e., there exist a tree T such that T is an mt for both (N_0, C) and (N_0, C') and $c_{ij} = c'_{ij}$ for all $(i, j) \in T$, then we have

$$y(N_0, C) = y(N_0, C').$$

Remark 5. SCM implies IIT (Bergantiños and Vidal-Puga, 2007a).

2.4.4 Simplifying properties

Given a charge rule y we define the following simplifying properties. This means that we can consider the mcst problem as smaller sub-problems which give the same cost allocation (Bergantiños and Vidal-Puga, 2007a; Trudeau, 2013b).

SEP Separability: if two disjoint subsets of agents connect independently to the source, then the cost shares can be computed separately for those subsets of agents.
For all mcst problems (N_0, C) and $S \subseteq N$ satisfying $m(N_0, C) = m(S_0, C) + m((N \setminus S)_0, C)$, we have

$$y_i(N_0, C) = \begin{cases} y_i(S_0, C) & \text{if } i \in S \\ y_i((N \setminus S)_0, C) & \text{if } i \in N \setminus S. \end{cases}$$

GI Group Independence: this property is similar to SEP, but it can only be applied if S and $N \setminus S$ are completely independent, i.e., no group in S has any gain if it cooperates with any group in $N \setminus S$.
For all mcst problems (N_0, C), if $S \subseteq N$ is such that for all $i \in S$ and $j \in N \setminus S$, $c_{ij} \geq \max\{c_{0i}, c_{0j}\}$, then

$$y_i(N_0, C) = \begin{cases} y_i(S_0, C) & \text{if } i \in S \\ y_i((N \setminus S)_0, C) & \text{if } i \in N \setminus S. \end{cases}$$

Remark 6. PM implies SEP and SEP implies GI (Trudeau, 2013b).

The following property focuses on the situation in which every agent has high connection cost to the source compared to the connection cost between agents. A charge rule satisfying the following property can compute the cost allocation for several simpler problems.

PS Problem Separation: for all mcst problems (N_0, C) such that C contains no irrelevant edge, i.e., there exist no edge $c_{ij} > \max\{c_{0i}, c_{0j}\}$ for all $i \neq j \in N$, then for all $i \in N$,

$$y_i(N_0, C) = y_i(N_0, \hat{C}) + y_i(N_0, \tilde{C}) - y_i(N_0, \dot{C}).$$

Where, $\hat{c}_{0i} = c_{0i}$ and $\hat{c}_{ij} = 0$ for all $i, j \in N$,
$\tilde{c}_{0i} = \max_{i,j \in N_0} c_{(i,j)}$ and $\tilde{c}_{ij} = c_{ij}$ for all $i, j \in N$,
$\dot{c}_{0i} = \max_{i,j \in N_0} c_{(i,j)}$ and $\dot{c}_{ij} = 0$ for all $i, j \in N$.

IIE Independence of Irrelevant Edges: no cost share depends on irrelevant edges. Thus if an edge is not used, increasing the cost of this edge should not influence the cost shares of the agents.
For any two mcst problems (N_0, C) and (N_0, C'), if $\max\{c_{0i}, c_{0j}\} \leq c_{ij} < c'_{ij}$ and $c_e = c'_e$ else, then

$$y(N_0, C) = y(N_0, C').$$

RA Restricted Additivity: if two mcst problems share an mt T, then the sum of the problems can be split into the two smaller problems.
For any two mcst problems (N_0, C) and (N_0, C'), if there exists $T \in T^*(C) \cap T^*(C')$, where $T^*(C)$ is the set of all mt's for the cost matrix C, and an order of the edges $\pi : T \to \{1, \dots, |N|\}$ such that for any $e, e' \in T$, if $\pi(e) \leq \pi(e')$, we have $c_e \leq c_{e'}$ and $c'_e \leq c'_{e'}$, then

$$y(N_0, C + C') = y(N_0, C) + y(N_0, C').^{16}$$

2.4.5 Other properties

Given a charge rule y we define the following other properties (Bergantiños and Vidal-Puga, 2008; Bogomolnaia and Moulin, 2010; Trudeau, 2013c).

CON Continuity: small changes in agents' connection cost do not lead to big changes in the amount they have to pay.
For all $N \subseteq \mathcal{N}$, $y(N_0, \cdot\,)$ is a continuous function of $\mathcal{C}^N$.

POL Polynomial Complexity: the cost share for each agent is computable in polynomial time.
For all mcst problems (N_0, C), $y(N_0, C)$ is computed in polynomial time.[17]

The following two properties take care of the situation when no one has a cheaper connection to the source than agent i and everyone can connect through agent i at no cost (Trudeau, 2013c). The first property claims that the cost of the direct link for agent i to the source is equally distributed among all agents. The second property argues that agent i should not be charged any of the cost of the direct link to the source.

ESCR Equal Share of Cost Reduction: for any two mcst problems (N_0, C) and (N_0, C') such that $c_{0i} \leq c_{0j}$ and there is a free path[18] p_{ij} for all

[16] Restricted Additivity implies Piecewise Linearity, i.e., if for any two cost matrices there is a common ranking of the edges from cheapest to most expensive, then addivity holds (Trudeau, 2013b).

[17] Ando (2012) proves that the Shapley value of a game is computable in polynomial time if the underlying graph $G(C, \alpha)$, i.e., graph with edge set $\{e \in E \mid c_e \leq \alpha\}$, is a chordal graph for all $\alpha \in \mathbb{R}_+$.

[18] This is a path from i to j such that $c_e = 0$ for all $e \in p_{ij}$.

$j \in N_0 \setminus \{i\}$, $c'_{0i} = c_{0i} - x$ and $c'_e = c_e$ otherwise, it is the case that for all $j \in N_0$ with $x \in [0, c_{0i}]$,

$$y_j(N_0, C') = y_j(N_0, C) - \frac{x}{|N|}.$$

FSCR Full Share of Cost Reduction: for any two mcst problems (N_0, C) and (N_0, C') such that $c_{0i} \leq c_{0j}$ and there is a free path p_{ij} for all $j \in N_0 \setminus \{i\}$, $c'_{0i} = c_{0i} - x$ and $c'_e = c_e$ otherwise, it is the case that for all $j \in N_0$ with $x \in [0, c_{0i}]$,

$$y_i(N_0, C') = y_i(N_0, C) - x \quad \text{and} \quad y_j(N_0, C') = y_j(N_0, C).$$

2.4.6 Comparison of charge rules

The following table summarizes which properties are satisfied by the charge rules discussed in Section 2.3. Whenever one of the boxes is left empty, this means that, to our knowledge, it is not proven that this property is satisfied by the particular charge rule. Proofs of the results can be found in the following papers: Kar (2002); Bergantiños and Vidal-Puga (2007a, 2008, 2009); Bogomolnaia et al. (2010); Bogomolnaia and Moulin (2010); Trudeau (2012, 2013b,c).

Some results are not stated in one of the papers mentioned before and are proven by us. This concerns the properties designed to distinguish the Kar rule and the folk solution, i.e., ESCR, FSCR, A, ET, RNK, IIE, PS, GI, RA. These properties are not evaluated for the Bird and Dutta-Kar rule. The proofs follow from the charge rule satisfying other properties or, in most of the cases that a property is not satisfied, a simple counterexample with two agents suffices. Moreover, we evaluted the property CM_0 for the Brid rule, Dutta-Kar rule and folk solution.

Property	B	K	DK	φ	CC
NPT	✓	–	✓	✓	–
BB	✓	✓	✓	✓	✓
CS	✓	–	✓	✓	✓
PM	–	–	–	✓	–
CM	–	✓	✓	✓	✓
CM_0	–		–	✓	
SCM	–	–	–	✓	–
ESEC	✓	✓	–	✓	✓
IOC	–	–	–	–	–
SYM	✓	✓	✓	✓	✓
RNK	✓	✓	–	✓	✓
ET	–	✓	–	–	
A	✓	✓	✓	✓	✓
IIT	–	–	–	✓	
SEP	–	–	–	✓	
GI	✓	✓	–	✓	✓
PS	–	✓	–	–	–
IIE	✓	✓	✓	✓	
RA	✓	–		✓	
CON	–	✓	–	✓	✓
POL	✓	–	✓	✓	✓
ESCR	–	–	–	✓	–
FSCR	–	✓	–	–	✓

From the results presented in the table we observe that the folk solution satisfies most of the properties. Therefore, given the charge rules discussed in Section 2.3, the folk solution may be considered as the most fair rule.

Chapter 3

The iterative mcst problem

In this chapter the iterative variant of the minimum cost spanning tree problem will be defined. This version of the mcst problem consist in agents who desire to be connected to the source in different rounds and thus unlike the classical case in which all agents want to connect to the source at the same time.

In Section 3.1 the framework for the iterative mcst problem is described. In Section 3.2 the algorithm for constructing a minimum cost spanning tree in the iterative case is given. Moreover, the complexity of the algorithm is discussed. Thereafter, in Section 3.3, we adapt the properties discussed in the previous chapter for the classical case to the iterative case, i.e., we change them to round-dependent properties. Furthermore, we specify some properties which are particularly interesting in the iterative case. Finally, we present different network structures based on real-life examples of the mcst problem.

3.1 Framework

In this section the framework for the mcst problem in the iterative situation is introduced. In addition, we discuss two other frameworks that also present an iterative setting and are related to the mcst problem. Thereafter, some additional concepts are defined.

3.1.1 The mcst problem

The framework for the iterative mcst problem is based on the framework for the classical mcst problem. As in the classical case, N is a finite set of agents and 0 is the source. In the iterative case agents want to be connected to the source in different rounds. In the first round a group of agents enters the network, then, in the second round a new group of agents enters the network, etc. This continues until round K which denotes the total number of rounds.

Let $V^k \subseteq N$ denote the set of agents who desire to be connected to the source in round $k \leq K$. We assume that no agent wants to be connected to the source in round 0 and therefore $V^0 = \{0\}$. The union of all agents who desire to be connected to the source in rounds $1, \dots, k$, is denoted, for $k = 0$, by $\mathbf{V}^1 = V^0 = \{0\}$, and for $k \leq K$, by

$$\mathbf{V}^{k+1} = \bigcup_{i=1}^{k} V^i.$$

In addition, similar to the classical case, we denote $\mathbf{V}_0^{k+1} = \mathbf{V}^{k+1} \cup \{0\}$.

Agents entering in round k of the iterative mcst problem can connect to the source in three ways:

(i) Directly, not using any other agent,

(ii) Indirectly, via another agent who connected in a previous round,

(iii) Indirectly, via another agent of their own round.

The following definition captures these three ways of connecting to the source in two possible actions. Agents of round k connect either directly or indirectly to the shrunk source 0^{k-1}.

Definition 2 (Shrunk source)**.**
The source for round $k \leq K$ is constructed by shrinking all vertices in $\mathbf{V}_0^k$ to one vertex, denoted by 0^{k-1}.[1] Regarding the edges, we remove the ones between the agents in $\mathbf{V}_0^k$ and keep the ones between the agents in V^k and $\mathbf{V}_0^k$ with minimal cost, i.e., for all $i \in V^k$, $(0^{k-1}, i)$ is such that $c_{0^{k-1}i} = \min\{c_{ij} \mid j \in \mathbf{V}_0^k\}$. For $k = 1$, the shrunk source is defined as $0^{k-1} = 0^0 = 0$.[2]

The next definition explains how to deal with the cost matrix C in the different rounds.

Definition 3 (Restricted cost matrix)**.**
The cost matrix C restricted to $V_{0^{k-1}}^k$, denoted by $C|_{V_{0^{k-1}}^k}$, is defined to be a $(|V^k| + 1) \times (|V^k| + 1)$ matrix $\hat{C}$ such that $\hat{c}_{ij} = c_{ij}$ if $i, j \in V^k$ and $\hat{c}_{0^{k-1}i} = \min_{j \in \mathbf{V}_0^k} c_{ij}$ for $i \in V^k$.

For notational convenience we will usually write C instead of $C|_{V_{0^{k-1}}^k}$ whenever it is clear from context that we mean the cost matrix restricted to the set $V_{0^{k-1}}^k$.

Similar to the classical mcst problem, the iterative mcst problem consists of two sub-problems:

[1] For a formal definition of shrinking vertices in a graph to one vertex we refer to the lecture notes of Schrijver (2012).

[2] Another way of saying it is that the shrunk source 0^{k-1} represents the tree constructed from round 1 until $k - 1$.

1. Constructing a network of minimum cost.

2. Dividing the cost over the agents who use the constructed network.

The difference with the classical mcst problem consists in the fact that the two sub-problems have to be solved after each round. The formal definition of the iterative mcst problem is given below.

Definition 4 (Iterative mcst problem)**.**
Given a partition[3] of the agents N into disjoint sets $V^1, \ldots, V^K$ and a cost matrix C for N_0, the iterative mcst problem is presented as follows

$$(V^1_0, V^2_{0^1}, \ldots, V^K_{0^{K-1}}, C).$$

The minimum cost associated with the iterative mcst problem is equal to the sum of the minimum costs associated with the different rounds, for $k \leq K$,

$$m(V^1_0, V^2_{0^1}, \ldots, V^K_{0^{K-1}}, C) = \sum_{k=1}^{K} m(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}}).$$

Here, $(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}})$ is the mcst problem in round k, i.e., the agents entering in round k desire to be connected to the source 0^{k-1}.

Furthermore, $m(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}})$ represents the minimum cost of the tree constructed in round $k \leq K$. Then, as in the classical case,

$$m(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}}) = \min_{T^k \in \mathcal{G}^{V^k}_{0^{k-1}}} c(T^k),^4$$

where T^k is the *tree* constructed in round k and $c(T^k) = \sum_{(i,j)\in T^k} c_{ij}$. The tree is constructed as in the classical mcst problem, e.g., by Prim's algorithm. The algorithm presented in Section 3.2 shows how the tree will be constructed for the iterative mcst problem when considering all rounds.

We will now explain the different possible trees used in this thesis for the iterative mcst problem. The *global tree* is denoted by G^k and represents the graph constructed after round $k \leq K$ of the iterative mcst problem. The global tree G^k is constructed by the algorithm given a partition of the agents in different rounds and cost matrix C as input. It represents all agents from the different rounds and their connections to the source or to each other. Thus, if we take the last constructed tree, the global tree is the tree where all the sources $0^1, \ldots, 0^{K-1}$ are unfolded. Generally, the global tree does not have to be a minimum cost spanning tree as in the classical mcst problem. Since the agents are entering the network in different rounds it is

[3]A partition of N into sets $V^1, \ldots, V^K$ is such that $V^i \cap V^j = \emptyset$ for all $i \neq j$, $i, j \in \{1, \ldots, K\}$ and $V^1 \cup \cdots \cup V^K = N$.

[4]Recall that we will write this as $m(V^k_{0^{k-1}}, C)$.

possible that the agents with the lowest connection cost to the source do not enter in the first round. Therefore, agents, who entered in the first round cannot connect via the agents with low direct connection cost. This results in the fact that some edges with high cost need to be constructed in order to connect the agents from the first round. The high costs thus have to be covered and the constructed network after round k is therefore not minimal. The tree which represents the problem in the classical case where all agents desire to be connected to the source at the same time is called the *optimal tree*. In round $k \leq K$ the optimal tree is equal to the tree constructed by the algorithm for the iterative case when all the agents of rounds $1, \ldots, k$ would have joined in the same round.[5] The optimal tree will always be a minimum cost spanning tree. The corresponding mcst problem is defined by $(\mathbf{V}_0^{k+1}, C)$ and the minimum cost $m(\mathbf{V}_0^{k+1}, C)$ is defined as in the classical mcst problem.

Notice that in the iterative case the optimal situation with respect to having minimal cost of the global tree is to let all agents enter the network in the same round, since only then we are sure that the spanning tree has minimum cost.[6] However, we assume that the time at which agents enter the network is fixed[7] and that the optimal situation thus cannot be forced.

As in the classical case, given as input a partition of the agents into disjoint sets and cost matrix C, a *solution* for the iterative mcst problem consists of two parts. One part is given by a global tree constructed by the algorithm and the other part is an allocation of the cost over the agents given by the charge rule. A *charge rule* for the iterative mcst problem takes as input the partition of the agents into disjoint sets, the global tree and cost matrix C, and outputs a cost share for each agent. For some charge rules, e.g., for the folk solution, it is not necessary to consider the global tree. In that case having the partition of the agents and the cost matrix as input is sufficient. Since both the solution and the charge rule output the cost share for the considered agents we will often use these notions interchangeably.

The following definition gives the charge rule for the iterative mcst problem formally. We assume the charge rule to completely cover the costs which have to be made to construct the tree in the iterative case. Moreover, an agent is only charged if she is actually using the network. Thus, agents which desire to be connected in future rounds are not charged anything yet.

Definition 5 (Charge rule)**.**
Given any iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ a charge rule is a function $\hat{y}$ mapping the iterative mcst problem for a given round $k \leq K$ to a

[5] Notice that it can happen, by a favorable partition of the agents, that the optimal tree is constructed at the end of round k.

[6] We assume here that optimality refers to having least possible cost. There are of course other definitions of optimality such as having least possible agents through whom one connects, but this is not of interest here.

[7] Although agents may artificially delay their arrival.

vector of payments,

$$\hat{y}(V_0^1, \ldots, V_{0^{k-1}}^k, C|_{\mathbf{V}_0^{k+1}}) \in \mathbb{R}^{|\mathbf{V}^{k+1}|}$$

such that for all $k \leq K$,

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C|_{\mathbf{V}_0^{k+1}}) = \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C|_{V_{0^{j-1}}^j})$$

and $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C|_{\mathbf{V}_0^{k+1}}) = 0$ for all $i \in V^q$ with $k < q \leq K$.

The first condition is called budget balance (BB) and is one of the basic properties we desire to be satisfied by a charge rule, see Section 3.3. The second condition is a standard assumption in the literature on mcst problems, in that only agents who are connected to the source are charged. Moreover, it is assumed that the source is not charged or paid anything (Bergantiños and Vidal-Puga, 2007a, 2008; Trudeau, 2013c,a; Bogomolnaia and Moulin, 2010; Moretti et al., 2001; Dutta and Kar, 2004).[8] The following example presents a charge rule which divides the cost over all agents who are using the network equally.

Example 6. Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ with minimum cost $\sum_{k=1}^K m(V_{0^{k-1}}^k, C)$, the following charge rule divides the costs equally over the corresponding agents, for $i \in \mathbf{V}^{k+1}$, for $k \leq K$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|}.$$

This rule decides in each round k the cost share of the agents who desire to get a connection to the source in round k and the ones who were already connected in previous rounds. A simple example is the following: let $N = \{1, 2, 3, 4\}$ be partitioned in $V^1 = \{1\}, V^2 = \{2\}$ and $V^3 = \{3, 4\}$. The complete graph is presented in Figure 3.1.

Then, the minimum cost associated with the different rounds are as follows: $m(V_0^1, C) = 4$, $m(V_{0^1}^2, C) = 5$ and $m(V_{0^2}^3, C) = 4$. This gives,

$$\hat{y}_i(V_0^1, C) = 4 \qquad \text{for } i \in \{1\}$$

$$\hat{y}_i(V_0^1, V_{0^1}^2, C) = \frac{4+5}{2} = 4.5 \qquad \text{for } i \in \{1, 2\}$$

$$\hat{y}_i(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = \frac{4+5+4}{4} = 3.25 \qquad \text{for } i \in \{1, 2, 3, 4\}.$$

The rule thus charges agent 1 again in the second round, she has to pay 0.5. This seems unfair since her connections do not change. In addition, agents 3 and 4 together pay 6.5 in the third round while they can construct a connection to the source by themselves with cost 4. They are thus better off by constructing their own network and not following the charge rule.

[8] However, one could see the source as the government or company who provides the source and therefore should be reimbursed. In this thesis we do not consider this case.

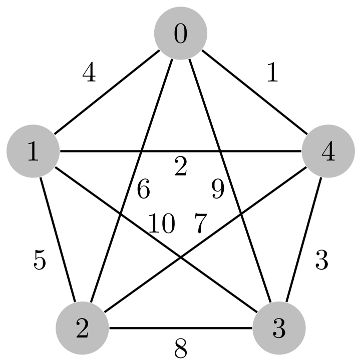

Figure 3.1: The complete graph of Example 6.

3.1.2 Related frameworks

In this section two other frameworks, related to the previously presented framework for the iterative mcst problem, are discussed.

Brenner and Schäfer (2010) propose an online model for general demand cost sharing games. “In an online setting, upon the arrival of a new agent, the mechanism has to take instantaneous and irreversible decisions without any knowledge of agents that arrive in the future.” They consider cooperative cost sharing games and a general demand setting, i.e., the mechanism determines which level of service is granted to which agent and for which price. Moreover, agents require not only one but more levels of service. Notice that this is different from our setting in which agents can only arrive once. Each agent has a valuation vector which indicates how much the agent values the service levels she demands. The valuation vectors are private information. In addition, each agent has a bidding vector which indicates how much the agent is willing to pay for the services. A general demand cost sharing mechanism takes the bid vectors from the agents and outputs a service allocation and payment for the concerned services for all agents. Brenner and Schäfer assume that agents act strategically and aim to maximize utility for the services. Utility for an agent is defined as the sum over all allocated services of the difference between the valuation and the price of a particular service. Moreover, an agent cannot lie about the characteristics or the arrival times of her request. The algorithm that they propose decides for each request, which consist of an agent asking for a particular service at a certain time, the price at that moment and checks whether the bid of the agent is greater or equal than the decided price. If this is the case, the agent receives the service for the selected price. If this is not the case, the agent does not receive the corresponding service.

Bergantiños and Lorenzo (2004, 2008a) propose a framework for a non-cooperative game approach of the mcst problem. Bergantiños and Lorenzo

(2004) provide a non-cooperative extensive form game Γ and Bergantiños and Lorenzo (2008a) extend this game to a budget restricted non-cooperative extensive form game Γ_α. In each stage of the game agents have two possible strategies, either connect to the source (directly or indirectly via agents connected in a previous stage), or stay unconnected. They ignore the possibility of connecting through other agents from the same stage which is the main difference with our approach. In addition, they assume that agents use stationary strategies, i.e., the strategies of the agents depend only on the agents who connected previously and not on the order in which they connected. A utility function in Γ is defined as follows: let z be a terminal node of Γ in stage t and assume that the game ends at stage t. Then, either agent i is connected to another agent or source i^* which gives utility $u_i(z) = -c_{ii^*}$, or agent i is unconnected and thus $u_i(z) = -\alpha$, with $\alpha > c_{0j}$ for all $j \in N$ according to Bergantiños and Lorenzo (2004). However, Bergantiños and Lorenzo (2008a) allow α to be smaller. By definition of Γ, a subgame $(t, R, (t_i)_{i \in R})$ can be defined for each stage t, where R represents the agents connected before stage t and t_i denotes the stage in which agent i decided be connected. A disadvantage of this approach is that the tree formed in stage t does not have to be a minimum cost spanning tree. Fernández et al. (2009) assume that a minimum cost spanning tree for the subset of agents who want to be connected in stage t is constructed at stage t. They introduce a class of profiles, the opportune moment strategies. Each agent desires to be connected at the stage in which she can connect via the cheapest link among all feasible links.

3.1.3 Additional concepts

The optimal situation for the iterative mcst problem is the situation in which all agents join at the same time,[9] or the situation in which agents accidentally enter in an order such that the minimum cost spanning tree is constructed. When we are not in the optimal situation, the minimum cost spanning tree for the classical case might not be constructed by the algorithm for the iterative mcst problem. In that case the constructed tree will have a certain inefficiency compared to the mcst constructed for the classical mcst problem. The inefficiency of the global tree depends on the partition of the agents in different rounds and the corresponding cost matrix C.

Definition 6 (Inefficiency of a tree)**.**
Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, the inefficiency I of a tree constructed in round $k \leq K$ is defined to be the difference between the

[9]This situation is equal to the iterative mcst problem taken as a classical mcst problem.

minimum cost of the global tree and the minimum cost of the optimal tree,

$$I(G^k) = \sum_{j=1}^{k} m(V^j_{0^{j-1}}, C) - m(\mathbf{V}^{k+1}_0, C).$$

The following lemma states that if there is no inefficiency in a certain round, then the inefficiency of the global trees in the rounds before is equal to 0. Notice that in the first round the inefficiency is always equal to zero.

Lemma 1. *If for $k \leq K$, $I(G^k) = 0$, then $I(G^q) = 0$ for all $q \leq k$.*

Proof. It suffices to prove that the lemma holds for $q = k - 1$, since then the lemma follows for all $q \leq k$ by induction.
Assume $I(G^{k-1}) \neq 0$, then by definition $m(\mathbf{V}^k_0, C) < \sum_{j=1}^{k-1} m(V^j_{0^{j-1}}, C)$.
It follows that,

$$\begin{aligned} m(\mathbf{V}^{k+1}_0, C) &\leq m(\mathbf{V}^k_0, C) + m(V^k_{0^{k-1}}, C) \\ &< \sum_{j=1}^{k-1} m(V^j_{0^{j-1}}, C) + m(V^k_{0^{k-1}}, C) = \sum_{j=1}^{k} m(V^j_{0^{j-1}}, C). \end{aligned}$$

Hence, $I(G^k) \neq 0$. □

In addition, we define the inefficiency of the tree constructed in round $k \leq K$ compared to the tree constructed in the previous round. The definition captures the extra cost of the network due to the agents who joined in round k compared to the cost caused when they would have joined in round $k - 1$.

Definition 7 (Inefficiency of a round)**.**
The inefficiency of round k is defined to be the difference between the situation in which agents of V^k enter separately and the situation in which they enter together with the agents of V^{k-1}. The inefficiency between round 0 and round 1, and round $k - 1$ and round k is defined as follows:

$$\begin{aligned} &I(G^1_0) = 0 \\ &I(G^k_{k-1}) = m(V^k_{0^{k-1}}, C) + m(V^{k-1}_{0^{k-2}}, C) - m((V^k \cup V^{k-1})_{0^{k-2}}, C) \end{aligned}$$

A charge rule R for the classical mcst problem is denoted by y^R in the iterative mcst problem. The definition is similar to the definition of a charge rule for the classical mcst problem, the difference lies in the corresponding source.

Definition 8 (Charge rule in round k, depending on R)**.**
Given any iterative mcst problem $(V^1_0, \ldots, V^K_{0^{K-1}}, C)$ and a charge rule R, a charge rule y^R for a mcst problem $(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}})$, for $k \leq K$, is defined to be a function mapping the mcst problem in round k to a vector of payments,

$$y^R(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}}) \in \mathbb{R}^{|V^k|}$$

such that for all $k \leq K$,

$$\sum_{i \in V^k} y_i^R(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}}) = m(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}})$$

and $y_i^R(V^k_{0^{k-1}}, C|_{V^k_{0^{k-1}}}) = 0$ for $i \in V^q$ with $k \neq q$.

The following lemma compares two situations and proves that the minimum cost of the tree constructed in situation one is less than or equal to the minimum cost of the constructed tree in situation two. Situation one describes agents $S \subseteq V^k$, who desire to be connected in the same round to the existing network, i.e., being connected to 0^{k-1}, either directly or indirectly via another agent in S. Situation two describes agents in $S \subseteq V^k$ when they do not use the existing network and connect as a group to the original source. The result follows from the fact that in case of connecting to the shrunk source the cost of direct edges are lower or equal than when connecting to the original source. The proof is straightforward and follows from the definitions. The corollary then states that, if a charge rule R satisfies BB, then the sum of the cost shares of the agents in S is at most as much in the iterative case as it is in the classical case.

Lemma 2. *For $S \subseteq V^k$ with $S \neq \emptyset$ and $k \leq K$,*

$$m(S_{0^{k-1}}, C) \leq m(S_0, C).$$

Proof. Given a cost matrix $\hat{C}$ corresponding to the set $S_{0^{k-1}}$ and a cost matrix $\tilde{C}$ corresponding to the set S_0 we have

$$\hat{c}_{ij} = \tilde{c}_{ij} \text{ for } i, j \in S$$

and

$$\hat{c}_{0^{k-1}i} \leq \tilde{c}_{0i} \quad \text{for } i \in S,$$

since $\hat{c}_{0^{k-1}i} = \min\{c_{\ell i} \mid \ell \in \mathbf{V}_0^k\}$. Hence, $\hat{C} \leq \tilde{C}$ by definition.
Thus, by definition of $m(S_{0^{k-1}}, C)$ and $G = (S_{0^{k-1}}, E)$, we have

$$\begin{aligned} m(S_{0^{k-1}}, C) &= \min\{c(S_{0^{k-1}}, \hat{C}, G) \mid G \text{ is a tree}\} \\ &= \min\{\sum_{(i,j) \in E} \hat{c}_{ij} \mid G \text{ is a tree}\} \\ &\leq \min\{\sum_{(i,j) \in E} \tilde{c}_{ij} \mid G \text{ is a tree}\} \\ &= \min\{c(S_0, \tilde{C}, G) \mid G \text{ is a tree}\} \\ &= m(S_0, C). \end{aligned}$$

Therefore, the lemma holds. □

Given that $m(S_0, C) = \sum_{i\in S} y_i(S_0, C)$.

Corollary 1. *For $S \subseteq V^k$, with $S \neq \emptyset$ and $k \leq K$, then*

$$\sum_{i\in S} y_i^R(S_{0^{k-1}}, C) \leq \sum_{i\in S} y_i^R(S_0, C).$$

Remark 7. Since Corollary 1 holds for $S \subseteq V^k$ we have the following result if the charge rule R satisfies CM_0,[10]

$$y_i^R(V^k_{0^{k-1}}, C) \leq y_i^R(V^k_0, C).$$

3.2 Algorithm

The algorithm for the iterative mcst problem is an adaptation of Prim's algorithm for the classical mcst problem. Since agents are joining in different rounds the algorithm used in the classical case should be executed each time a new group of agents desires to be connected to the source. The general idea is the following: after a set of agents V^1 enters in round 1 we run Prim's algorithm on V^1 and source 0. When a new set of agents V^2 then desires to be connected to the source in round 2 we first search for the cheapest edge for all agents $i \in V^2$ between i and any agent j of the previous rounds, i.e., for $j \in V^1$. Then, we shrink the source node and the nodes representing the agents of round 1 to one node and call this node 0^1. Thereafter, we run Prim's algorithm again, this time with the agents in V^2 and source 0^1. These steps will be repeated until there are no more agents who desire to be connected to the source. Notice that in each round agents will be added to the network and therefore no connections between agents or agents and the source of previous rounds will change.[11]

In the next section the algorithm is presented in a formal way and explained providing an example. Then, in Section 3.2.2, the complexity of the algorithm is discussed.

3.2.1 Pseudocode

In Algorithm 3, B denotes the set of agents who are already connected to the source (consisting of j's). A denotes the set of agents who need to be connected to the source in a particular round (consisting of i's). The agent j_i^* is the previously connected agent, or the source itself, to whom agent i

[10] We refer to Section 3.3.2 for an adaptation of the properties from the classical mcst problem to the iterative mcst problem.

[11] One could think of allowing for changing the previously constructed network, e.g., in case the network is cheaper in the long term. However, in this thesis we do not consider this since in the example of houses connecting to a water source reconstructing some pipelines will not prevent that the high costs of the beginning have to be covered.

has the cheapest connection cost. The agent i^* then is the not yet connected agent who has the cheapest connection cost to one of the j's, or the source, of all of the i's. Note that i^* can possibly denote more than one agent. If that is the case, just select one of them. Regarding the notation introduced in the previous section we output G^k after round k and the agents in $\mathbf{V}^{k+1}$ are connected to the source after the algorithm has been applied in round k.[12]

Algorithm 3 Iterative minimum cost spanning tree

Input: A partition of N vertices in $V^1, \ldots, V^K$ and a cost matrix C.
Ouput: A global tree $G^K = (N_0, E)$ that connects all vertices in $V^1, \ldots, V^K$ to the source 0.

```
V^0 ← {0}, E ← ∅, B ← {0}                    # Initialization
for k from 1 to K do
  A ← V^k                                    #Introduce new agents
  for i ∈ A do
    j*_i ← arg min_{j∈B} c_ij
  end for
  while A ≠ ∅ do
    i* ← arg min_{i∈A} c_{i j*_i}
    A ← A \ {i*}
    B ← B ∪ {i*}                             #Add new agent to graph
    E ← E ∪ (i*, j*_i)                       #Add new edge to graph
    for i ∈ A do
      if c_{ii*} < c_{i j*_i} then
        j*_i ← i*                            #Update costs
      end if
    end for
  end while
end for
```

The algorithm takes at most $|N| = n$ rounds and each round takes at most $|V^k|$ steps. The set of edges E together with the vertices N form the global tree after round K.

Notice that in each round $k \leq K$ a minimum cost spanning tree is constructed regarding the agents of round k and source 0^{k-1}, but the global tree constructed after round K does not have to be a minimal one. The following example will illustrate this case.

[12] The algorithm can be seen as a dynamic graph algorithm since the solution will be efficiently maintained after an update of the input, i.e., a new set of agents enters, rather than having to recompute the solution from the beginning. However, the solution does not have to be optimal. We refer to Eppstein et al. (1996) for an introduction in dynamic graph algorithms.

Example 7. Given the complete graph with corresponding costs on the edges as shown in Figure 3.2. Let $N = \{1, 2, 3, 4\}$ and let the agents be partitioned in sets $V^1 = \{4\}$, $V^2 = \{1, 2\}$ and $V^3 = \{3\}$.

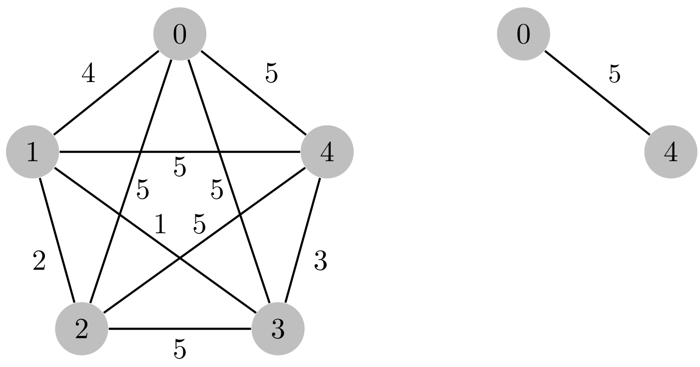

Figure 3.2: The complete graph before and mcst of round 1.

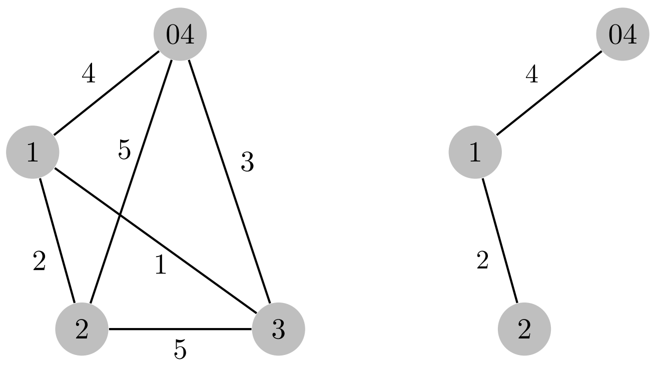

Figure 3.3: The complete graph before and mcst of round 2.

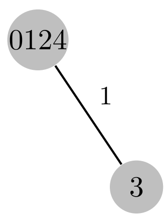

Figure 3.4: Complete graph before and mcst after round 3.

Figures 3.2, 3.3 and 3.4 show the complete graphs before each round and the mcst in each round constructed by the algorithm. The following figure shows the global tree which is constructed by the algorithm after round 3 for the iterative mcst problem.

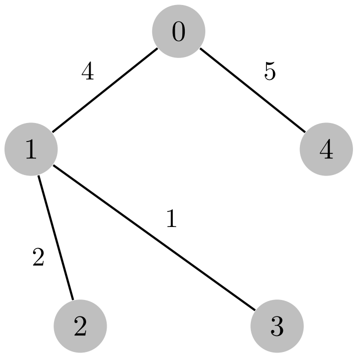

Figure 3.5: Global tree constructed after round 3.

The cost of the global tree presented in Figure 3.5 equals $m(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = 12$. However, the global tree constructed for the iterative mcst problem in this example by the algorithm after the last round is not equal to the minimum cost spanning tree in the optimal case. Consider the following situation: given a different partition of the agents, for example, let agents 1 and 3 connect to the source in the first round and let agents 2 and 4 connect in the second round. The algorithm then would have constructed the minimum cost spanning tree shown in Figure 3.6, which is equal to the optimal tree, i.e., the situation in which agents 1, 2, 3 and 4 desire to be connected in the same round. The cost of the optimal tree, $m(\{1,2,3,4\}, C) = 10$, which is less than the cost needed to construct the global tree of Figure 3.5.

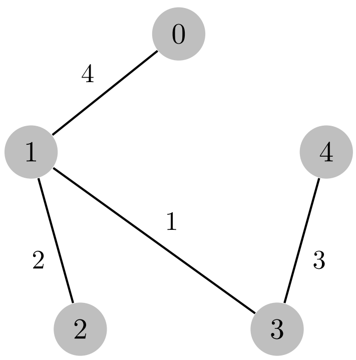

Figure 3.6: Mcst for $V^1 = \{1,3\}$ and $V^2 = \{2,4\}$.

3.2.2 Complexity

In the previous section the algorithm for finding a minimum cost spanning tree for the iterative mcst problem was presented. In this section we will discuss the complexity of this algorithm. Before stating the complexity result an overview of the different steps in the algorithm is given.

INPUT: Partition of N in disjoint sets $V^1, \ldots, V^K$ and $(n+1) \times (n+1)$-cost matrix C.

Start from $k = 1$.

1. For $i \in V^k$, list the cost of edges $(0^{k-1}, i)$ such that

$$c_{0^{k-1}i} = \min_{j \in \mathbf{V}_0^k} c_{ij}.$$

2. Shrink the vertices in $V_{0^{k-2}}^{k-1}$ to the shrunk source 0^{k-1}.
3. Run Prim's algorithm on $V^k \cup \{0^{k-1}\}$.

Repeat till $k = K$.

OUTPUT: Global tree G^K.

One could think that the complexity of the algorithm for the iterative mcst problem is lower than the complexity of Prim's algorithm for the classical mcst problem. For example, consider the case in which in each round exactly one agent wants to be connected to the source. In that case, Prim's algorithm becomes very easy since there is only one possible edge to choose in each round. However, the first step of the algorithm for the iterative mcst problem requires that all the edges between the new agents and all previously connected agents have to be checked. Therefore, the algorithms for the classical and the iterative case share the same upper bound on the complexity which will be proved in the next theorem. Before this result we will state and explain the complexity of the different steps of the algorithm for the iterative mcst problem.

Step 1 has complexity $O(|V^k| \cdot |\mathbf{V}_0^k|)$. For each agent $i \in V^k$ we have to check $|\mathbf{V}_0^k|$ edges.

Step 2 has complexity $O(1)$. Notice that this step is not a necessary step in the algorithm. This is the reason why it does not appear in the pseudo code of the algorithm. However, we do believe that this way of presenting the problem is insightful. After each round we consider the agents from the previous rounds equal and the agents who will join in the next round can connect to the source via one of those agents. We therefore consider the agents of the previous rounds, including the source, as one new source.

Step 3 has complexity $O(|V^k|^2)$. In the first step of Prim's algorithm we start from the source, search through $|V^k|$ edges for $i \in V^k$ and pick the edge with lowest cost, say $(0^{k-1}, i^*)$. Then we add this edge to the tree. Also, we remove i^* from the list. In the second step, we search through $|V^k| - 1$ edges (i^*, j) for $j \in V^k \setminus \{i^*\}$. If we find for j an edge with lower cost than the cost of the edge $(0^{k-1}, j)$, then we update the list with agent i^* for j instead of 0^{k-1}. For agents $j \in V^k \setminus \{i^*\}$ we pick the edge with lowest cost and add this one to the tree. This continues until step $|V^k|$ in which only 1 edge has to be checked. Hence, for connecting $|V^k|$ agents to source 0^{k-1}, $\frac{|V^k|(|V^k|+1)}{2}$ edges have to be checked. Therefore, the complexity of the algorithm in round k equals,

$$O(|V^k| \cdot |\mathbf{V}_0^k|) + O(1) + O(|V^k|^2).$$

Moreover, the complexity of the algorithm for all rounds equals,

$$O\left(K \cdot \max_{k \leq K}(|V^k| \cdot |\mathbf{V}_0^k| + |V^k|^2)\right).$$

Hence, by using Prim's algorithm in each round, we can conclude that the complexity of the algorithm for the iterative mcst problem is at most quadratic in the number of agents.[13]

Theorem 4. *The iterative minimum cost spanning tree algorithm can be executed in quadratic time (like Prim's algorithm for the classical mcst problem).*

Proof. By induction on the number of rounds $k \leq K$.
For $K = 1$ the theorem is trivially true since then the algorithm of the iterative mcst problem is equal to the classical one.
Assume that until round $k-1$ the complexity of the iterative mcst problem is in the same order as the complexity of the classical mcst problem, i.e., when the agents of rounds 1 until $k-1$ join altogether.
Let $K = k$ and let from round 1 to $k-1$, $|\mathbf{V}^k| = m \leq n$ agents be connected to the source. Assume that in round k we have $|V^k| = p$ agents who desire to get connected to the source with $m + p = n$. Then, we need to show that the complexity of (i) is in the same order as the complexity of (ii):

(i) connecting p agents in round k to the source in a minimal way. This should be added to the complexity of the m agents connecting to the source in the rounds 1 until $k-1$.

(ii) connecting $m + p$ agents to the source in round 1.

The way to show this is counting the number of edges we have to check in order to select the edge with minimal cost in both cases.

[13] Using more sophisticated data structures as input instead of the adjacency matrix, the running time of Prim's algorithm can be faster than quadratic. For example, using adjacency lists or adjacency lists and Fibonacci heaps, the complexity is $O(|V| \log |E|)$ and $O(|E| + |V| \log |V|)$ respectively (Campos and Ricardo, 2008; Fredman and Tarjan, 1987; Martel, 2002). However, in this thesis this is not of interest.

(i) For connecting m agents to the source in rounds 1 to $k-1$ we have by I.H. that $\frac{m(m+1)}{2}$ edges have to be checked. Then, connecting p agents in round k requires that $pm + \frac{p(p+1)}{2}$ edges have to be checked. Thus, in total, after round k the number of checked edges is equal to $\frac{m(m+1)}{2} + pm + \frac{p(p+1)}{2}$.

(ii) For connecting $m+p$ agents to the source in round 1, $\frac{(m+p)(m+p+1)}{2}$ edges have to be checked. Eliminating the brackets of this sum gives $\frac{(m+p)(m+p+1)}{2} = \frac{m(m+1)}{2} + pm + \frac{p(p+1)}{2}$.

Notice that in both cases we cannot do better than checking the edges presented before since we have to check each edge at least once. On the other hand, we will not do worse since this is how the algorithm specifies to do it. Therefore, we can conclude that in both cases the number of edges that have to be checked is equal and thus the algorithm for the iterative mcst problem can be executed in quadratic time. □

3.3 Properties

In this section different properties which we would like the charge rule for the iterative mcst problem to satisfy after each round will be presented. First, we list the basic properties from Chapter 2 for the iterative case. Then, we show how to adapt the properties introduced for the classical case, i.e., stability, comparative, simplifying, and other properties, to round-dependent properties. Finally, we define iterative properties, i.e., the properties that are interesting particularly for the iterative mcst problem.

3.3.1 Basic properties

The following properties are standard assumptions we have on the charge rule $\hat{y}$.

NPT No Positive Transfer: for all iterative mcst problems $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for all $k \leq K$, if $i \in \mathbf{V}^{k+1}$, we have

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) \geq 0.$$

BB Budget Balance: for all iterative mcst problems $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for all $k \leq K$, we have

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

Recal that BB is part of the definition of any charge rule, see for example Definition 5. In the sequel we will continue to list BB as a property only when we want to specifically emphasize this assumption.

A weaker version of BB is the property β-Budget Balance.[14] We desire the charge rule to satisfy full BB in each round. However, one can argue that if the agents pay more than the cost of the minimum tree this will not harm the construction of the network and can therefore be seen as a property we would accept too.

β-BB β-Budget Balance: the total cost share obtained from all agents up to round k deviates by a factor of at most $\beta \geq 1$ from the total cost. For all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k \leq K$,

$$\sum_{j=1}^{k} m(V_{0^{j-1}}^j, C) \leq \sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) \leq \beta \cdot \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

A charge rule not satisfying NPT means that an agent could be paid for using the network. When BB is not satisfied by the charge rule, the total cost of the constructed tree is not covered and it is therefore not possible to construct the network as some agents might not be able to connect to the source. Both situations are excluded since we do not consider them as fair.

3.3.2 Classical properties

The following properties for the charge rule in the iterative mcst problem are similar to the properties presented for the classical mcst problem in Chapter 2 (Bergantiños and Vidal-Puga, 2007a; Trudeau, 2013b). However, we need to adapt the properties to the framework for the iterative mcst problem introduced in the first section of this chapter. For each class of properties[15] presented in Chapter 2 one or two properties are listed below, for the ones not listed here the modification is similar. We require the properties to be satisfied after each round.

CS Core Stability: for all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k \leq K$ and $S \subseteq V^k$, we have

$$\sum_{i \in S} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) \leq m(S_0, C).$$

Here we assume that agents can only ask people from their own round to form a coalition and construct their own network without using the existing

[14] In the literature this property is sometimes called Weak-Budget-Balance or Cost Recovery (Leonardi and Schäfer, 2004; Tazari, 2005).

[15] Stability, comparative and simplifying properties.

network. The agents from previous rounds are already connected and charged for being connected to the source. It therefore does not make sense to ask them to reconnect and incur costs again. The agents who will enter in later rounds are not available yet.

PM Population Monotonicity: for all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k \leq K$, $S \subseteq V^k$ and $i \in S$, we have

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) \leq \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, S_{0^{k-1}}, C).$$ [16]

CM_0 Source Cost Monotonicity: for all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ and $(V_0^1, \ldots, V_{0^{K-1}}^K, C')$, for all $k \leq K$, such that $c_{0i} \leq c'_{0i}$ for $i \in V^k$ and $c_e = c'_e$ else, we have

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) \leq \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').$$

SYM Symmetry: for all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k \leq K$, and any pair of agents $i, j \in V^k$ such that for all $\ell \in \mathbf{V}_0^{k+1} \backslash \{i, j\}$ we have $c_{i\ell} = c_{j\ell}$, it is the case that

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \hat{y}_j(V_0^1, \ldots, V_{0^{k-1}}^k, C).$$

IIE Independence of Irrelevant Edges: for all iterative mcst problems $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ and $(V_0^1, \ldots, V_{0^{K-1}}^K, C')$, for all $k \leq K$, if $\max\{c_{0i}, c_{0j}\} \leq c_{ij} < c'_{ij}$ and $c_e = c'_e$ else, then

$$\hat{y}(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \hat{y}(V_0^1, \ldots, V_{0^{k-1}}^k, C').$$

Remark 8. Notice that for CS, PM and SYM in the classical mcst problem the agents i, j (or coalition S) are taken from the set N. However, for the iterative mcst problem we consider $i, j \in V^k$ for $k \leq K$. For example, for symmetry, if agents have the same adjacent connection costs but are joining in different rounds, then it does not have to be the case that their cost share is the same. In the iterative mcst problem we do not call them symmetric.

3.3.3 Iterative properties

In this section we state some properties which are particularly interesting in the iterative mcst problem where agents are joining in different rounds. The aim is to find a charge rule for the iterative mcst problem which satisfies each of the following properties after each round. For the classical mcst problem these properties will be vacuously true since all agents desire to be connected to the source at the same time.

[16] In the literature on cost sharing mechanism design this property is called *cross-monotonicity* and received a lot of attention because of the difficulty of this property being satisfied in a cost-sharing scheme (Tazari, 2005; Leonardi and Schäfer, 2004; Gupta et al., 2007; van Zwam, 2005).

OP One Payment:[17] agents only have costs in the round they are joining, and in later rounds they might be reimbursed.

Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for all $2 \leq k \leq K$ and $i \in \mathbf{V}^k$, we have,

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) \leq \hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C).$$

R-OS R-Optimal Stability: in case there is no inefficiency of the constructed network, the cost share for each agent after round k is equal to the cost share of the optimal problem given by a charge rule R for the classical case.

Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$ and a charge rule R, for all $k \leq K$ and $i \in \mathbf{V}^{k+1}$, if $I(G^k) = 0$, then we have,

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C).$$

JIT Join in Time:[18] agents will join the network in the round they truly desire to be connected to the source, meaning that there is no incentive to join later.[19]

Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for no $k < K$, there exist $i \in V^k$ such that

$$\hat{y}_i(V_0^1, \dots, V_{0^{K-1}}^K, C) >$$
$$\hat{y}_i(V_0^1, \dots, (V^k \setminus \{i\})_{0^{k-1}}, \dots, (V^K \cup \{i\})_{0^{K-1}}, C).$$

The property says that if an agent desires to be connected to the source in round k and decides to join in the last round K, and everyone else will connect in the same round, then this agent cannot be charged less in round k than in round K.

[17]This property can also be called PM over rounds, like Bergantiños and Gómez-Rúa (2010) call a similar property PM over groups (PMG).

[18]This property is related to the property called *strategyproofness* in the literature on cost sharing mechanisms, i.e., for every player bidding truthfully is a dominant strategy (Brenner and Schäfer, 2010; van Zwam, 2005; Gupta et al., 2007; Leonardi and Schäfer, 2004). However, JIT is different since our framework does not include utilities and bidding vectors. We could say that an agent's utility is 0 if she is not connected to the source and her utility is infinity if she is connected to the source. We cannot say anything about the utility of an agent who desires to be connected but thinks the price is too high compared to this agent waiting some time while not being connected and paying a lower price later. Therefore, we cannot talk about strategyproofness in the way it is done in the literature on general cost sharing mechanisms and gave it another name.

[19]An agent will certainly not join earlier since we assume this to be physically impossible, for example, when houses are not built yet. Moreover, the chance of being able to connect via a cheaper way is higher by joining later since by then there are more other agents through which one could connect to the source.

G-JIT Group-Join in Time:[20] if no coordinated entering of the network, where agents may join later but not earlier, of a coalition $S \subseteq V^k$ can strictly decrease the cost shares of some agent in S without strictly increasing the cost share of some other agent(s) in S.[21]

Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k < K$ and all coalitions $S \subseteq V^k$, if there exists some agent $i \in S$ such that

$$\hat{y}_i(V_0^1, \ldots, V_{0^{K-1}}^K, C) >$$
$$\hat{y}_i(V_0^1, \ldots, (V^k \setminus \{S\})_{0^{k-1}}, \ldots, (V^K \cup \{S\})_{0^{K-1}}, C),$$

then there exist some agent $j \in S$ such that

$$\hat{y}_j(V_0^1, \ldots, V_{0^{K-1}}^K, C) <$$
$$\hat{y}_j(V_0^1, \ldots, (V^k \setminus \{S\})_{0^{k-1}}, \ldots, (V^K \cup \{S\})_{0^{K-1}}, C).$$

The following property is a weaker version of JIT. The added constraint says that the cost of the network should either be the same or become larger when an agent joins later, i.e, the inefficiency of the global tree should increase or remain the same. More precisely, if an agent joins later and the cost of the network goes down, thus the inefficiency decreases, this means something positive. In that case at least someone should be charged less in order to maintain BB. One could ask why not all agents join later and have the lowest possible cost for constructing the network. This would bring us in the optimal situation, but is in contradiction with the assumption that the round in which agents desire to be connected to the source is fixed.

W-JIT Weak-Join in Time: agents will join in the round they truly desire to be connected to the source, assuming that the inefficiency of the global tree does not decrease by an agent joining later.

Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, there exists no $k < K$ and no agent $i \in V^k$ such that if,

$$m(V_0^1, \ldots, V_{0^{K-1}}^K, C) \leq$$
$$m(V_0^1, \ldots, (V^k \setminus \{i\})_{0^{k-1}}, \ldots, (V^K \cup \{i\})_{0^{K-1}}, C), \tag{3.1}$$

[20]This property is related to the notion of group-strategyproofness in the literature on cost sharing mechanisms (Brenner and Schäfer, 2010; van Zwam, 2005). Moreover, if group-join in time is satisfied, then join in time is satisfied as well.

[21]Notice that we could also allow for taking S from different rounds V^k for $k < K$ and even allow for the agents entering later in different rounds V^q for $k < q \leq K$. However, we do not see this as a realistic situation since agents who want to form a coalition should not be connected yet and therefore desire to be connected in the same round. Moreover, we assume coalitions to be formed because of agents being able to benefit from each other while connecting to the source, hence they will enter in the same round at a later point.

then,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{K-1}}^K, C) >$$
$$\hat{y}_i(V_0^1, \ldots, (V^k \setminus \{i\})_{0^{k-1}}, \ldots, (V^K \cup \{i\})_{0^{K-1}}, C).$$

Notice that 3.1 can also be expressed as follows: let the global tree for the iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ be denoted by G^K and let the global tree for the iterative mcst problem $(V_0^1, \ldots, (V^k \setminus \{i\})_{0^{k-1}}, \ldots, (V^K \cup \{i\})_{0^{K-1}}, C)$ be denoted by G^{K_i}, then 3.1 can also be stated as $I(G^K) \leq I(G^{K_i})$.

Impossibility result

Ideally the charge rule for the iterative mcst problem satisfies all the basic, classic and iterative properties. However, if we assume the basic properties and the classic property CS to hold, then we cannot assure that JIT is satisfied. Therefore, we have the impossibility result stated in Theorem 5. This result is interesting since these properties seem quite natural to be satisfied at the same time by a charge rule we would call fair. We assume BB to hold by definition of a charge rule and NPT is also one of the basic conditions. CS will prevent that agents construct their own network and JIT will preclude that agents enter the network at a later time than they truly desire to enter.

Theorem 5. *No charge rule $\hat{y}$ for the iterative mcst problem that satisfies NPT, BB and CS will satisfy JIT.*

Proof. Assume $\hat{y}$ satisfies NPT, BB and CS. Consider the mcst problem presented in Figure 3.7 and let $V^1 = \{2\}$ and $V^2 = \{1\}$. Then, in the first round, $\hat{y}_2(V_0^1, C) = 3 + \epsilon$. We cannot charge agent 2 either more or less because of BB. In the second round agent 1 will connect to the source directly. The cost of the global tree after the second round is $m(V_0^1, C) + m(V_{0^1}^2, C) = 3 + \epsilon + 1 = 4 + \epsilon$. We cannot charge agent 1 more than 1 because of CS. Therefore, because of BB the cost share of agent 2 after the second round is at least $3 + \epsilon$, i.e., $\hat{y}_2(V_0^1, V_{0^1}^2, C) \geq 3 + \epsilon$. Consider the situation in which agent 1 and 2 join together, thus $V^1 = \{\emptyset\}$ and $V^2 = \{1, 2\}$. Then, the minimum cost of the constructed tree is $m(V_0^1, V_{0^1}^2, C) = 3$. Because of CS we cannot charge agent 1 more than 1. Because of NPT we should charge agent 1 at least 0, so $0 \leq \hat{y}_1(V_0^1, V_{0^1}^2, C) \leq 1$. Because of BB we should charge agent 2 at least 2 but not more than 3, so $2 \leq \hat{y}_2(V_0^1, V_{0^1}^2, C) \leq 3$. Hence, in the first situation $\hat{y}_2(V_0^1, V_{0^1}^2, C) \geq 3 + \epsilon$ and in the second situation where agent 2 decides to join later $\hat{y}_2(V_0^1, V_{0^1}^2, C) \leq 3$. Thus, for $\epsilon > 0$,

$$\hat{y}_2(V_0^1, V_{0^1}^2, C) \geq 3 + \epsilon > 3 = \hat{y}_2((V^1 \setminus \{2\})_0, (V^2 \cup \{2\})_{0^1}, C),$$

which shows that $\hat{y}$ does not satisfy JIT. $\square$

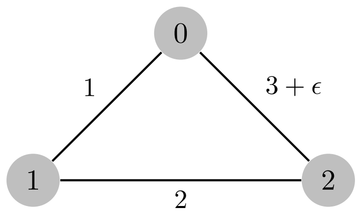

Figure 3.7: Counterexample for Theorem 5.

The following examples are presented in order to show that all properties are needed for the impossibility result. The first charge rule satisfies the properties NPT, CS and JIT but not BB, the second one satisfies NPT, BB and JIT, but not CS. For the combination of properties CS, BB and JIT but not NPT, we present a rule which satisfies the three properties in the case of two agents. However, for the case of three agents it might not be possible to find such a rule. The problem has presumably to do with the fact that the total cost of the network can decrease when agents enter later. This is the reason why W-JIT is defined.

Example 8. NPT and CS.
The charge rule allocates 0 to every agent. Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for all $k \leq K$ and $i \in \mathbf{V}^{k+1}$, we have

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = 0.$$

NPT and CS are clearly satisfied, BB is not if the connection costs are not all 0. JIT is satisfied since each agent is charged the same in each round, therefore it is not beneficial to join later.

Example 9. NPT and BB.
The charge rule allocates the cost of the global tree constructed in round k to the agents entering in round k and the agents who entered in previous rounds are charged nothing. Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, for all $k \leq K$ and $i \in V^k$,

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = \begin{cases} 0 & \text{if } i \in V^q, q < k \\ \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|V^k|} & \text{if } i \in V^q, q = k. \end{cases}$$

NPT and BB are clearly satisfied, but CS is not, since agents in V^k should pay for all the cost of the agents connecting in previous rounds. In some round, this is more than the cost as when they would connect to the source themselves without using the existing network and therefore are better off by constructing their own network. JIT is satisfied, since if an agent joins in round $k < K$, she is charged 0 in round K. If she then decided to join later in round K she is charged $\frac{\sum_{j=1}^K m(V_{0^{j-1}}^j, C)}{|V^K|} > 0$.

Example 10. BB and CS.
The following charge rule works for the case $n = 2$. Charge every agent i her direct connection cost to the source. If another agent j connects via agent i in the same or a later round, then subtract the difference between the edge that j uses and j's direct cost to the source from agent i's direct cost to the source. Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for all $k \leq K$ and $i \in V^k$ and $j \in V^q$ for $k \leq q \leq K$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = c_{0i} - c_{0j} + c_{ij}.$$

BB is satisfied since the cost of the direct edges to the source which are not used are subtracted from the agents which are used to connect through. CS is satisfied since no agent pays more than her direct connection cost to the source. This is the only thing to check for the case $n = 2$ since BB holds. JIT is satisfied since an agent can only be reimbursed when she joins earlier and thus is available to connect through. If $n \geq 3$ the charge rule does not satisfy CS. The following example will illustrate this.

Example 11. Let $V^1 = \{3\}$, $V^2 = \{1\}$ and $V^3 = \{2\}$ and let the complete graph be given in Figure 3.8. Then,
$\hat{y}_1(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = 1$,
$\hat{y}_2(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = 6$,
$\hat{y}_3(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = 3 + \epsilon - 6 + 3 = \epsilon$.
One can check that in this case CS does hold in each round since no one pays more than their direct cost to the source. If now agent 1 decides to join later together with agent 2, then $V^1 = \{3\}$ and $V^2 = \{1, 2\}$ and one can compute that the cost shares for the agents are the same as in the previous situation. However, CS does not hold since,

$$\hat{y}_1(V_0^1, V_{0^1}^2, C) + \hat{y}_2(V_0^1, V_{0^1}^2, C) = 1 + 6 = 7 > 5 = m(\{1, 2\}_0, C).$$

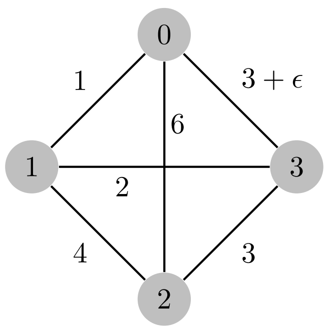

Figure 3.8: Counterexample for $n = 3$ in Example 11.

3.4 Network structures in real-life examples

In this section, we consider some specific network structures. These network structures are interesting because they represent real-life situations and some of them are particularly interesting for the iterative mcst problem.

Considering the iterative properties defined in the section above, it might be the case that some properties are not satisfied by a charge rule in general, but are satisfied in a particular network structure.

We state a number of different network structures below. The first one can be motivated by the fact that the source is further away from each agent than the agents from each other. Therefore, connecting to the source directly is expensive and connecting via other agents can be done by having relatively low cost.[22] The second network structure deals with the fact that some connections between agents cannot be made, for example because of a mountain being in between two houses and therefore it being actually impossible to construct a pipeline (Moretti et al., 2001). In the complete graph these edges are given cost ∞ to make clear that they cannot be used. All other edges have the same cost. The third class of networks assumes that when new agents join the network, for example because a new district with newly built houses is completed, they have low connection cost between each other and high cost to the source and other houses which were connected before. The last two network structures have a restriction on the way in which agents may connect through other agents. For the second but last, the restriction is on the number of other agents which can connect via an agent (Bogomolnaia et al., 2010; Moulin and Laigret, 2011; Moulin, 2013). Assume, for example, that a house can only have a certain number of cables being placed under the floor. For the last network, the restriction is on the length of the path between every agent and the source (Bergantiños et al., 2012). It might be the case that because of the long distance and lots of agents in between, when, for example, a signal has to travel between the source and a particular agent, the agent will not be able to receive it completely or with the desired strength. The following are the network structures we consider.

I. High cost to the source directly, low cost between agents. For example,

$$\min\{c_{0i} \mid i \in \mathbf{V}^{K+1}\} > \max\{c_{j\ell} \mid j, \ell \in \mathbf{V}^{K+1}\}.$$

Or, if we have one fixed cost for all agents for connecting to the source directly, e.g., for all $i \in \mathbf{V}^{K+1}$, $c_{0i} = \gamma > \max\{c_{j\ell} \mid j, \ell \in \mathbf{V}^{K+1}\}$.

II. Some connections between agents cannot be made, i.e., the cost of these edges equal ∞, and the other connection costs between agents and between agents and source are all the same.

[22] This assumption is widely used in the literature on the classical mcst problem (Bergantiños and Vidal-Puga, 2007a, 2008; Bergantiños and Lorenzo, 2008b; Trudeau, 2013a,b,c).

For some $i, j \in \mathbf{V}_0^{K+1}$ we have $c_{ij} = \infty$, otherwise we have $c_{ij} = 1$.

III. Agents connecting in the same round have low cost between each other, the connection costs to source and other agents are high.

For all $i, j \in V^k$ such that $k \leq K$, the c_{ij} are equal, and for all $\ell \in \mathbf{V}_0^{K+1} \setminus V^k$ we have $c_{i\ell} > c_{ij}$.

IV. Restriction on the number of agents that can connect via any other agent.[23]

For all $i \in \mathbf{V}^{K+1}$ we have $d(i) \leq s$ for $s \in \mathbb{N}$, where $d(i)$ is the number of agents that connect through agent i to the source.

V. Restriction on the length of the path between every agent and the source.[24]

For all $i \in \mathbf{V}^{K+1}$ we have for the path $\{(i_{s-1}, i_s)\}_{s=1}^{p}$ where $i_0 = i$ and $p = 0$ that $|\{(i_{s-1}, i_s)\}_{s=1}^{p=0}| \leq w$ for $w \in \mathbb{N}$.

In network structures I and III the inefficiency will be relatively low and in network structure II the inefficiency will always be 0. The situation in which the inefficiency is high occurs when agents enter the network in an unfortunate way. For example, when the ones with relatively high direct cost to the source join early and the ones with relatively low direct cost to the source join later. In network structure I this cannot be the case since everyone has the same cost to the source. In network structure II the cost of the constructed network in round k will always be equal to the number of agents who desire to be connected to the source in rounds 1 until k since the cost of each constructed edge equals one. This network will therefore have no inefficiency. A constraint on the partition of the agents in different rounds is that there should always be one possible edge for an agent to the source, one of the previously connected agents or one of the agents entering in the same round. In network structure III, everyone will connect to each other within one round and one agent connects to the source directly or to one of the previously connected agents. This will also not cause much inefficiency. On the contrary, network structures IV and V cause inefficiency since we put a constraint on the way in which agents can connect through each other. Hence, the minimal way in which the agents can connect through each other might not be possible to achieve. Notice that it is an NP-hard combinatorial optimization problem to find a minimum cost spanning tree with a structure as presented by the last two cases (Tazari, 2005; Bergantiños et al., 2012;

[23]Moulin and Laigret (2011) introduce an allocation of the cost of a network under connectivity constraints.

[24]Bergantiños et al. (2012) prove that the core of such a problem could be empty. Moreover, they provide a cost allocation rule for this specific minimum cost spanning tree problem, called k-hop mcst problems.

Moulin and Laigret, 2011). Furthermore, network structures IV and V cannot be expressed in the framework for the iterative mcst problem introduced in this chapter since the constraints are not expressible via cost matrix C. It would be interesting to consider the possibility of enlarging the framework in such a way that we can express these constraints. Moreover, it might be possible to perform minimal networks satisfying these constraints with lower complexity in the iterative setting. We suggest this as a direction for further research.

Chapter 4

Fair sharing of inefficiencies

In this chapter we will consider the so-called *fair sharing of inefficiencies* approach to the distribution of costs of the constructed network over the agents in the iterative mcst problem. In the first section the general idea behind and motivation for the approach will be explained. Fair sharing of inefficiencies leads to four charge rules which will be defined in Section 4.2. In Section 4.3 the basic, classic and iterative properties stated in Chapter 3 will be evaluated. We will see that some of the iterative properties are not satisfied in general. However, we consider the special network structures presented in Chapter 3 for those iterative properties and prove that some of them are satisfied in particular network structures.

4.1 Motivation

In this section the motivation behind the *fair sharing of inefficiencies* approach will be explained. Thereafter, the procedure for allocating the cost among the agents is introduced.

Fair sharing of inefficiencies divides the extra cost, obtained because agents join in different rounds, equally after each round over the agents who are connected to the source until that round. Because of our assumption that the time at which agents desire to be connected to the source is fixed, an agent cannot be penalized for being late (or early). It is therefore possible to argue that it is fair to charge every agent an equal part of the cost caused by the inefficiency of the network. The part of the cost which should be paid by an agent can depend on several factors such as, for example, on an agent's cost share decided by a charge rule for the classical mcst problem. It does not matter how much the agents are charged in previous rounds, as long as the cost shares, in particular the inefficiency parts, in round k are equal for all agents who are connected to the source in round k.

The procedure for allocating the cost over the agents in round $k \leq K$ starts by selecting a charge rule R for the classical case. Thereafter, a solution is selected. This solution equally distributes the inefficiency of the network in round k over the agents who have entered the network until round k. In each round, every new agent is charged what she would have been charged in the optimal case,[1] plus an equal part of the inefficiency of the network in round k. Each previously connected agent is charged the difference between the amount she was charged in round $k-1$ and the amount she is charged in the solution in round k. An 'equal distribution' of the cost caused by the inefficiency of the constructed network can be defined in different forms and will therefore lead to different charge rules.

Each solution corresponds to a way of equally distributing the inefficiencies. They are similar in the sense that they first decide the cost shares of the agents who joined until round k by a charge rule R for the optimal situation. Then, the extra cost, i.e., the cost caused by the inefficiency of the constructed network, will be divided over those agents. The allocation of the extra costs over the agents can be accomplished in several ways. It depends on what one considers to be an equal distribution of the extra cost because of the fact that agents enter the network in different rounds. Each solution takes a different point of view on how to divide the inefficiency equally. The first solution is defined in such a way that everyone's extra cost share is the same in each round. The second solution is defined such that everyone's extra cost share is proportional to their cost determined by the charge rule R. When one has high costs according to the existing rule, then one will also have high extra costs. The third solution is defined in such a way that if one's cost is twice as high as someone else's cost, then the extra cost should be twice as low. The extra cost shares are thus, similar to the second solution, allocated to the agents depending on the cost they have to pay according to the charge rule R. The fourth solution does not depend on one of the charge rules for the classical mcst problem and is defined such that every agent is charged the same.

4.2 Charge rule

The charge rule for allocating the cost of the constructed network will be applied after each round. After new agents join the network the charge rule provides the allocation of the costs for the new agents by setting their cost share equal to their cost share in one of the solutions. For the agents who are connected in previous rounds the charge rule does the same, but since they have already been charged in the previous round, the rule gives the additional cost share or reimbursement. The agents who will connect in later rounds are not charged anything yet.

Given an iterative mcst problem and a charge rule R for the classical

[1] Recall that this is the situation in which all agents who entered until round k would have been there at the same time.

mcst problem,[2] the solutions described in the previous section lead to four charge rules (a), (b), (c) and (d).

Definition 9 (Charge rules for Fair sharing of inefficiencies)**.**
Given any iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for $k \leq K$, and agents $i \in \mathbf{V}^{k+1}$ the charge rule $\hat{y}$ is defined as follows:

(a) $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|}$.

(b) $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C)$, where $\beta = \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{m(\mathbf{V}_0^{k+1}, C)}$.

(c) $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \cdot \delta_i$, such that for $i, j \in \mathbf{V}^{k+1}$ we have $\frac{y_i^R(\mathbf{V}_0^{k+1}, C)}{y_j^R(\mathbf{V}_0^{k+1}, C)} = \frac{\delta_j}{\delta_i}$ and $\sum_{i \in \mathbf{V}^{k+1}} \delta_i = |\mathbf{V}^{k+1}|$.[3]

(d) $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|}$.[4]

The charge rule can also be defined by specifying the extra cost q_i^k for agent i in round $k \leq K$ given i's cost share in round $k-1$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \begin{cases} \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) + q_i^k & \text{if } i \in \mathbf{V}^k \\ \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) & \text{if } i \in V^k \end{cases},$$

where,

$$q_i^k = \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C).$$

Notice that $q_i^k > 0$ means that agent i, who connected before round k, is charged in round k, and $q_i^k \leq 0$ means that agent i is not charged anything in round k, she will eventually be reimbursed.

Remark 9. If $i \in V^k$ for $k \leq K$, then the cost share in round K equals:

$$\hat{y}_i(V_0^1, \ldots, V_{0^{K-1}}^K, C) = \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \sum_{j=k+1}^{K} q_i^j.$$

[2]Recall that we denote a charge rule R for the classical mcst problem by y^R in the definitions and proofs.

[3]One could also decide on δ_i for $i \in \mathbf{V}^{k+1}$ according to some other factors. For example, if someone's income is twice as high, then the extra cost share should be twice as high.

[4]Notice that this charge rule does not depend on R and thus strictly speaking does not belong to the cooperative game approach.

Example

An example, similar to the one presented in Figure 2.1, demonstrates charge rule (a). For charge rule R, the folk solution is selected. The iterative mcst problem is given by the following partition of the agents: $V^1 = \{1\}$ and $V^2 = \{2,3\}$ with cost matrix $C = \begin{pmatrix} 0 & 10 & 5 & 11 \\ 10 & 0 & 2 & 1 \\ 5 & 2 & 0 & 4 \\ 11 & 1 & 4 & 0 \end{pmatrix}$.

Then,

$$\begin{aligned}
\hat{y}_1(V_0^1, C) &= y_1^R(\mathbf{V}_0^2, C) + \frac{I(G^k)}{|\mathbf{V}^2|} = 10 + \frac{0}{1} = 10 \\
\hat{y}_1(V_0^1, V_{0^1}^2, C) &= y_1^R(\mathbf{V}_0^3, C) + \frac{I(G^k)}{|\mathbf{V}^3|} = \frac{5}{2} + \frac{5}{3} = \frac{25}{6} \\
\hat{y}_2(V_0^1, V_{0^1}^2, C) &= y_2^R(\mathbf{V}_0^3, C) + \frac{I(G^k)}{|\mathbf{V}^3|} = 3 + \frac{5}{3} = \frac{28}{6} \\
\hat{y}_3(V_0^1, V_{0^1}^2, C) &= y_3^R(\mathbf{V}_0^3, C) + \frac{I(G^k)}{|\mathbf{V}^3|} = \frac{5}{2} + \frac{5}{3} = \frac{25}{6}.
\end{aligned}$$

Thus, in the first round agent 1 is charged 10, and in the second round agent 1 is charged $4\frac{1}{6}$. Therefore, in the second round she will receive $q_1^2 = 5\frac{5}{6}$. Agent 2 and 3 are charged nothing in the first round, and in the second round they are charged $4\frac{2}{3}$ and $4\frac{1}{6}$ respectively. The cost allocation after the second round is,

$$\hat{y}(V_0^1, V_{0^1}^2, C) = (4\frac{1}{6}, 4\frac{2}{3}, 4\frac{1}{6}).$$

4.3 Properties

In Chapter 3 we listed several properties and discussed why one would like the charge rule to satisfy those properties in order to call the rule a fair rule. In this section, we first prove that the basic properties are satisfied by the charge rules (a), (b), (c) and (d). We then consider the classical properties and state which of them are satisfied by the different charge rules defined in the previous section. Finally, we discuss the iterative properties. When one of them is not satisfied by the charge rule we check whether the property is satisfied when we restrict ourselves to a particular network structure.

4.3.1 Basic properties

Proposition 1. *Given an iterative mcst problem* $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$*, the charge rules (a), (b), (c) and (d) satisfy the basic properties NPT and BB for all* $k \leq K$*, if* R *satisfies NPT and BB.*

Proof. See Appendix I. □

4.3.2 Classical properties

The following theorem tells us which classical properties, given that R satisfies the considered property, are satisfied by charge rules (a), (b), (c) and (d) for the iterative mcst problem. Most of the proofs are straightforward and rely heavily on R satisfying the classical property.

Theorem 6. *Given an iterative mcst problem* $(V_0^1, \dots, V_{0^{k-1}}^k, C)$*, for* $k \leq K$*, the charge rules (a), (b), (c) and (d) satisfy, respectively,*

(a) CM, ESEC, SYM, RNK, ET, A, IIT, IIE, CON, POL, ESCR and FSCR,

(b) CM, SYM, RNK, A, IIT, IIE, CON and POL,

(c) ESEC, SYM, A, IIT, CON and POL,

(d) CM, CM_0*, SCM, ESEC, SYM, RNK, ET, A, IIT, PS, IIE, RA, CON, POL and ESCR,*

if charge rule R satisfies the considered property.

Proof. See Appendix I. □

Remark 10. The properties SYM, RNK and ET do not only hold for two agents from the same round, but also for agents who join in different rounds, i.e., $i \in V^q$ and $j \in V^{q'}$ such that $q \neq q' \leq K$.

4.3.3 Iterative properties

The following theorem states which of the iterative properties defined in Section 3.3.3 are satisfied by the charge rules (a), (b), (c) and (d).

Theorem 7. *Given an iterative mcst problem* $(V_0^1, \dots, V_{0^{K-1}}^K, C)$*, for all* $k \leq K$*, charge rules (a), (b) and (c) satisfy R-OS, charge rules (a), (b), (c) and (d) satisfy W-JIT. The charge rules (a), (b), (c) and (d) do not satisfy OP, JIT and G-JIT.*

Proof.

R-OS (a) Since, $I(G^k) = 0$,

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} = y_i^R(\mathbf{V}_0^{k+1}, C).$$

(b) Since $\beta = 1$,

$$\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) = \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C) = y_i^R(\mathbf{V}_0^{k+1}, C).$$

(c) Since, $I(G^k) = 0$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|}\delta_i = y_i^R(\mathbf{V}_0^{k+1}, C).$$

OP Given the following iterative mcst problem: $V^1 = \{1\}$, $V^2 = \{3\}$, $V^3 = \{2\}$, and $C = \begin{pmatrix} 0 & 5 & 8 & 10 \\ 5 & 0 & 6 & 100 \\ 8 & 6 & 0 & 2 \\ 10 & 100 & 2 & 0 \end{pmatrix}$. Selecting the Bird rule for R gives for charge rule (a), $\hat{y}_1(V_0^1, V_{0^1}^2, C) = 5 < 6\frac{1}{3} = \hat{y}_1(V_0^1, V_{0^1}^2, V_{0^2}^3, C)$. Which means that agent 1 is charged more in the third round than in the second round. A counterexample for the other charge rules is similar.

JIT For all charge rules, joining in round K can reduce the inefficiency of the constructed network and therefore also lower your cost share. Consider the example presented in Figure 3.7 with R being the folk solution and charge rule (a). Then, in situation 1, when $V^1 = \{2\}$, $V^2 = \{1\}$ and $I(G^2) = 1$, the cost share for agent 2 is $\hat{y}_2(V_0^1, V_{0^1}^2, C) = 2\frac{1}{2}$. In situation 2, when $V^1 = \{1, 2\}$ and $I(G^2) = 0$, the cost share for agent 2 equals $\hat{y}_2(V_0^1, C) = 2 < 2\frac{1}{2}$. Thus, when agent 2 joins in the last round her cost share decreases. For charge rules (b), (c) and (d) a counterexample is similar.

G-JIT G-JIT is not satisfied by charge rule (a), (b), (c) and (d) since JIT is not satisfied by these charge rules and G-JIT implies JIT.

W-JIT Given condition 3.1 in the definition of W-JIT, the inefficiency of the global tree in the last round K can not decrease when agent $i \in V^k$ decides to join in round K. Therefore for charge rules (a), (b), (c) and (d),

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) \leq$$
$$\hat{y}_i(V_0^1, \ldots, (V^k \setminus \{i\})_{0^{k-1}}, \ldots, (V^K \cup \{i\})_{0^{K-1}}, C),$$

which proves that the charge rules (a), (b), (c) and (d) satisfy W-JIT. □

Notice that, with respect to the property W-JIT, in case an agent joins later and the inefficiency can thus only increase (or stay the same), the cost share is not only higher for this agent, but for all agents who were connected until that round.

If $I(G^k) = 0$ for all $k \leq K$, then charge rules (a), (b) and (c) coincide with charge rule R. In that case, independent of the round in which an agent enters the network, all agents who entered until round k will be charged their cost share according to R in round k. This is independent of the selected charge rule R. For OP to be satisfied we require R to satisfy PM since then an agent's cost share will be lower when more agents desire to be connected to the source.

Proposition 2. *Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, the charge rules (a), (b) and (c) satisfy OP (if R satisfies PM), JIT and G-JIT for all $k \leq K$ in network structure II.*[5]

The proof is immediate since in network structure II the inefficiency is equal to 0. In network structures I and III,[6] the inefficiencies will be low, but there might always be a situation, similar to the situation in the counterexample for OP and JIT in Theorem 7, in which OP or JIT are not satisfied.

4.4 Discussion

Considering the classical properties, CS and PM are not satisfied by the charge rules (a), (b), (c) and (d). This can be seen as the main disadvantage of these rules. Agents can be better off by constructing their own network than by using the global network. Furthermore, when more agents join the network in a certain round the cost shares might increase. An advantage is that charge rules (a), (b) and (d) satisfy CM, which says that agents cannot take advantage when they report false connection costs (Bergantiños and Vidal-Puga, 2008).

JIT is not satisfied by any of the charge rules. This is not surprising since by joining later the total cost of the network might decrease which means that in that case every agent is charged at most as much as before. However, W-JIT is satisfied by all discussed charge rules. Agents will thus join the network in the round in which they truly desire to be connected to the source, if the total cost of the network does not decrease.

In addition, OP is not satisfied by any of the charge rules. This means that agents might have to pay extra in later rounds. Since we cannot foresee the inefficiency of the network in round k we cannot charge the agents in the first round in such a way that we do not have to charge them extra in later rounds. Moreover, we want the charge rule to charge the agents in such a way that the parts of the inefficiency that they have to pay are equal after each round. It would thus not make sense to charge the agents who are connected to the source in the first round more in order to let them make only one payment.

One could consider a charge rule which in round k divides the inefficiency of the round, i.e., $I(G_{k-1}^k)$, instead of the inefficiency of the global tree, over

[5]In these network structures, some links are impossible, all other links have cost 1.

[6]Network structure I corresponds to high costs between agents and the source and low cost between agents. Network structure III corresponds to high cost to the source and to other agents not in the same round.

the agents joining in round k. This would mean that the agents who desire to be connected in round k are charged for the inefficiency they cause because of entering in round k instead of entering in round $k-1$. However, with respect to the definition of the inefficiency of a round, considering the inefficiency of a round instead of considering the inefficiency of the global tree, is not so straightforward since $I(G^k) \neq \sum_{j=1}^{k} I(G^j_{j-1})$.[7] Therefore, if we use $I(G^k_{k-1})$ instead of $I(G^k)$ we will not have budget balance. It would be interesting to consider a charge rule that takes into account the inefficiency of the round and satisfies BB. We propose this as a direction for further research.

[7]See Definition 7 in Chapter 3.

Chapter 5

Reimbursement of previously connected agents

In this chapter, a second approach, called *reimbursement of previously connected agents*, solving the second sub-problem of the iterative mcst problem is introduced. It leads to a charge rule for the iterative mcst problem which divides the cost of the constructed network among the agents. In Section 5.1 the general idea behind and motivation for reimbursement of previously connected agents is given. In Section 5.2 we define upper and lower bounds on agents' possible payments and repayments in order to decide how much the previously connected agents can be reimbursed. These bounds are necessary for defining the charge rule in Section 5.3. In Section 5.4, the properties discussed in Chapter 3 are evaluated, i.e., the proofs of the basic, classical and iterative properties which are satisfied by the charge rule are given. We then consider the network structures presented in Chapter 3 for the iterative properties which are not satisfied. Moreover, we prove that some of the iterative properties which are not satisfied in general are satisfied in a particular network structure.

5.1 Motivation

We start by first giving the motivation for *reimbursement of previously connected agents* and then introduce the procedure for allocating the cost.

The main motivation for the approach consists in the fact that we desire every agent to be charged an equal proportion of the total cost of the network after each round. The agents connecting in the first rounds might have high connection costs since they do not have many alternatives to which to connect. The chance of having lower connection costs is greater when an agent joins later, since she simply has more options to connect to available. We therefore want to charge the agents who connected to the network in later rounds something extra in order to reimburse the agents that connected earlier. Thus, we want to charge agents only in the round in which they enter the network since this automatically means that in later rounds they can only get reimbursed. We can even charge them a bit more since we know that they will be reimbursed in later rounds. However, we do not want to charge agents more than what they would have paid if they connected to the source by themselves, i.e., without using the existing network. Considering the properties of the charge rule, we thus require core stability (CS) to be satisfied. Summarizing, we want to charge the new joining agents as much as possible without violating CS.

The procedure for allocating the cost over the agents in the iterative case is as follows: we select a charge rule R for the classical mcst problem.[1] Then, we compute the cost share of the agents of round k according to the selected charge rule R. After allocating the cost of the network constructed in round k we check whether it is possible to charge the agents in round k something extra so that the agents who entered in previous rounds will be reimbursed. Recall that, on the one hand, we want to charge agents in round k as much as possible, on the other hand, we do not want them to deviate from the global tree and construct their own network. In Section 5.2, an upper and a lower bound will be defined in order to decide the range of extra charges. In addition, an upper and a lower bound will be defined for the amount of repayments of agents connected in previous rounds. Depending on the policy, the exact payments for agents in round k, i.e., $i \in V^k$, and reimbursements for previously connected agents, i.e., $i \in \mathbf{V}^k$, are decided. In Section 5.2.3 several policies are introduced.

5.2 Redistributing the cost

In this section, the redistribution of the cost of agents connecting to the source in a certain round via previously connected agents is defined formally. We define upper and lower bounds on the agents' extra payments in Section 5.2.1. Then, in Section 5.2.2, we introduce the upper and lower bounds on agents' reimbursements. Furthermore, several policies are presented in Section 5.2.3 in order to decide the exact payments and reimbursements. This preliminary work is necessary in order to present the charge rule for the iterative mcst problem in Section 5.3.

5.2.1 The amount agents are charged extra

Although the total cost of the network constructed until round k is covered by using the charge rule R for each mcst problem $(V^k_{0^{k-1}}, C)$ for $k \leq K$, we consider the possibility of charging the agents joining in round k extra. As a result, the previously connected agents can be reimbursed. The question

[1] We refer back to Chapter 2 for an overview of the possible charge rules R.

then is, how much of the cost can be transferred from agents in V^k to agents in $\mathbf{V}^k$? We will consider the advantage an agent has from using the existing network, compared to the situation in which she constructs a network herself, i.e., in which she pays the cost of directly connecting to the source.

However, an agent can also form a coalition with agents from the same round and connect to the source together with them without using any edges from the existing network. The following definition expresses the advantage a coalition has from using the existing network instead of constructing a network by itself.

Definition 10 (Benefit of a coalition of agents)**.**
Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, the benefit b of a coalition of agents $S \subseteq V^k$, for $k \leq K$, is defined to be the difference between the sum of their cost shares, given charge rule R, when they connect to the source together with the agents in $V^k \setminus S$, via agents in $\mathbf{V}^k$, and the total cost of the network that they can construct among themselves, i.e.,

$$b(S) = m(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C).$$

The fact that the charge rule in the definition of $b(S)$ takes the mcst problem $(V_{0^{k-1}}^k, C)$ as its input, instead of the mcst problem $(S_{0^{k-1}}, C)$, can be motivated by the following: if agents use the existing network in which case the source is 0^{k-1}, they may also connect via the other agents joining in round k who are not in S. We do not consider arbitrary $S \subseteq \mathbf{V}^{k+1}$ since agents who are already connected to the source do not want to incur more costs by reconnecting.

Remark 11*.* If S is a single agent, $i \in V^k$, Definition 10 simplifies to,

$$b(i) = c_{0i} - y_i^R(V_{0^{k-1}}^k, C).$$

If S equals V^k, the benefit of V^k is

$$b(V^k) = m(V_0^k, C) - m(V_{0^{k-1}}^k, C).$$

It should be noted that for $S = V^k$, the definition does not depend on charge rule R because of the standard assumption that R should be budget balanced, i.e., $\sum_{i \in V^k} y_i^R(V_{0^{k-1}}^k, C) = m(V_{0^{k-1}}^k, C)$. Furthermore, it is easy to observe that $b(V^1) = 0$.

The following two lemmas provide an upper and a lower bound on the benefit of a set of agents S by using CS of the charge rule R.

Lemma 3. *If R satisfies CS, then for $S \subseteq V^k$, $k \leq K$, with $S \neq \emptyset$,*

$$b(S) \geq 0.$$

Proof.

$$
\begin{aligned}
b(S) &= m(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) \\
&\geq m(S_{0^{k-1}}, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) && \text{(Lemma 2)} \\
&\geq 0. && \text{(CS of } R\text{)}
\end{aligned}
$$

□

Lemma 4. *If R satisfies BB and CS, then for all $S \subseteq V^k$, $k \leq K$, with $S \neq \emptyset$,*

$$b(S) \leq \sum_{i \in S} b(i).$$

Proof.

$$
\begin{aligned}
b(S) &= m(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) \\
&= \sum_{i \in S} y_i^R(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) && \text{(BB of } R\text{)} \\
&\leq \sum_{i \in S} c_{0i} - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) && \text{(CS of } R\text{)} \\
&= \sum_{i \in S} \Big(c_{0i} - y_i^R(V_{0^{k-1}}^k, C) \Big) \\
&= \sum_{i \in S} b(i).
\end{aligned}
$$

□

Notice that $b(V^k)$ is the maximum cost share that can be charged extra from the agents in round k without violating CS. Otherwise, they could set up their own network without using the existing network and have lower cost shares.

We now present a lower and an upper bound in order to determine the extra cost share for the agents in round k.

Definition 11 (Lower bound on extra cost share)**.**
Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, in round $k \leq K$, the lower bound on the extra cost share of agent $i \in V^k$ is equal to the maximum of 0 and the difference between agent i's cost share in the optimal situation and i's cost share in round k according to the charge rule R, i.e.,

$$\tilde{\alpha}_i^k = \max\{0, y_i^R(\mathbf{V}_0^{k+1}, C) - y_i^R(V_{0^{k-1}}^k, C)\}.$$

Remark 12. If the charge rule R satisfies SCM, the upper bound is defined as

$$\tilde{\alpha}_i^k = y_i^R(\mathbf{V}_0^{k+1}, C) - y_i^R(V_{0^{k-1}}^k, C).$$

Since formally we have,

$$y_i^R(V_{0^{k-1}}^k, C) = y_i^R(V_{0^{k-1}}^k, C|_{V_{0^{k-1}}^k}) = y_i^R(\mathbf{V}_0^{k+1}, C^k),$$

where C^k is a cost matrix such that $c_{ij}^k = \begin{cases} c_{ij} & \text{if } i,j \in V^k \\ \min\{c_{ij}\} & \text{if } i \in V^k,\ j \in \mathbf{V}_0^k \\ 0 & \text{if } i,j \in \mathbf{V}_0^k. \end{cases}$

Hence, $C^k \leq C$ and thus by SCM, for all $i \in \mathbf{V}^{k+1}$,

$$y_i^R(\mathbf{V}_0^{k+1}, C^k) \leq y_i^R(\mathbf{V}_0^{k+1}, C)$$

and thus $\tilde{\alpha}_i^k \geq 0$.

We set a lower bound as we do not want agents to pay less than they would in the optimal situation. Since we consider R to be a fair rule in the optimal case, we have no reason to let people pay less than what this charge rule allocates to them. Moreover, the lower bound should be at least 0 since we do not want any agent to receive payments from any of the agents of her own round.

We define an upper bound on the extra cost share of an agent in round k to ensure that agents will not deviate from the global tree.

Definition 12 (Upper bound on extra cost share)**.**
Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, in round $k \leq K$, the upper bound on the extra cost share of agent $i \in V^k$ is equal to the maximum of 0 and the minimum of the difference between agent i's cost share in the situation with source 0^{k-1} and agents from V^k, and the situation with source 0 and agents from S, according to charge rule R, i.e.,

$$\mu_i^k = \max\left\{0, \min\{y_i^R(S_0, C) - y_i^R(V_{0^{k-1}}^k, C) \mid S \subseteq V^k \text{ and } i \in S\}\right\}.$$

Remark 13. If the charge rule R satisfies PM, the upper bound is defined as

$$\mu_i^k = \max\left\{0, y_i^R(V_0^k, C) - y_i^R(V_{0^{k-1}}^k, C)\right\}.$$

Since we have, $y_i^R(V_0^k, C) \leq y_i^R(S_0, C)$ for all $S \subseteq V^k$ by PM. If in addition, R satisfies CM_0, then

$$\mu_i^k = y_i^R(V_0^k, C) - y_i^R(V_{0^{k-1}}^k, C).$$

The range in which agent i can be charged extra cost p_i^k in round $k \leq K$, on top of what the charge rule R allocates to him, is the difference between the lower and upper bound, we therefore have

$$\tilde{\alpha}_i^k \leq p_i^k \leq \mu_i^k.$$

The exact payment p_i^k of agent $i \in V^k$ depends on the policy we follow.

The following lemmas establish bounds on $\tilde{\alpha}_i^k$ and μ_i^k in terms of the benefit of a coalition $S \subseteq V^k$. Whether the bounds hold depends on the properties satisfied by the charge rule R selected in the beginning of the procedure. Moreover, a comparison of the upper and lower bound tells us which properties charge rule R should satisfy in order to have the lower bound smaller than or equal to the upper bound.

Lemma 5. *If R satisfies BB, PM and CM$_0$, then for all $S \subseteq V^k$ with $S \neq \emptyset$, $k \leq K$, we have*

$$\sum_{i \in S} \mu_i^k \leq b(S).$$

Proof.

$$\begin{aligned}
\sum_{i \in S} \mu_i^k &= \sum_{i \in S} \Big(y_i^R(V_0^k, C) - y_i^R(V_{0^{k-1}}^k, C) \Big) && (\text{CM}_0 \text{ \& PM of } R) \\
&\leq \sum_{i \in S} y_i^R(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) && (\text{PM of } R) \\
&= m(S_0, C) - \sum_{i \in S} y_i^R(V_{0^{k-1}}^k, C) && (\text{BB of } R) \\
&= b(S).
\end{aligned}$$

□

Remark 14. If $S = V^k$ and R satisfies PM and CM$_0$, then $\sum_{i \in V^k} \mu_i^k = b(V^k)$.

Lemma 6. *If R satisfies PM, then for $i \in V^k$ for all $k \leq K$, we have*

$$\tilde{\alpha}_i^k \leq \mu_i^k.$$

Proof. We distinguish two cases:
Case 1: $\tilde{\alpha}_i^k = 0 \leq \mu_i^k$ since $\mu_i^k \geq 0$.
Case 2: $\tilde{\alpha}_i^k \neq 0$

$$\begin{aligned}
\tilde{\alpha}_i^k &= y_i^R(\mathbf{V}_0^{k+1}, C) - y_i^R(V_{0^{k-1}}^k, C) \\
&\leq y_i^R(V_0^k, C) - y_i^R(V_{0^{k-1}}^k, C) && (\text{PM of } R) \\
&\leq \mu_i^k.
\end{aligned}$$

□

The next corollary follows from Lemma 5 and Lemma 6.

Corollary 2. *If R satisfies BB, PM and CM$_0$, then for $S \subseteq V^k$, we have*

$$\sum_{i \in S} \tilde{\alpha}_i^k \leq b(S).$$

By Lemma 6 we can conclude that the charge rule R we select in the beginning of the procedure should satisfy PM in order to satisfy the condition that the lower bound on the extra cost share of agent i is less than or equal to the upper bound on the extra cost share of agent i. This is a strong restriction on the selection of the charge rule R since there are only a few rules which satisfy this property of which the folk solution is one (Bergantiños and Vidal-Puga, 2007a).

5.2.2 Reimbursement of agents

Now that we know the upper and lower bound on the amount that can be charged extra to the agents joining in round k, the question is how to divide these costs over the agents who entered the network in previous rounds, i.e., in rounds $1, \ldots, k-1$. Similarly as for the agents in round k, the agents of the previous rounds should not be charged less than their cost share in the optimal problem under the given charge rule R. We therefore define an upper bound on what the previously connected agents can receive in round k.

Definition 13 (Upper bound on reimbursement)**.**
Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, for $k \leq K$, let α_i^k be the difference between agent i's cost share in round $k-1$, and her cost share in the optimal problem under charge rule R. For $i \in \mathbf{V}^k$,

$$\alpha_i^k = \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - y_i^R(\mathbf{V}_0^{k+1}, C).$$

The lower bound on what agent i receives is equal to 0. Hence, the range in which an agent i can receive payment r_i^k is the difference between the lower and upper bound, i.e.,

$$0 \leq r_i^k \leq \alpha_i^k.$$

Now that we defined the lower bound on what agents in V^k should be charged extra and the upper bound on what agents in $\mathbf{V}^k$ can receive in round $k \leq K$, the following lemma shows that the amount which can be received by the previously connected agents is greater than or equal to the amount which has to be charged extra to the new agents, i.e., their lower bound.

Lemma 7. *If R satisfies BB and SCM, then for all $k \leq K$ we have*

$$\sum_{i \in \mathbf{V}^k} \alpha_i^k \geq \sum_{i \in V^k} \tilde{\alpha}_i^k.$$

Proof. By definition,

$$\begin{aligned}
\sum_{i\in\mathbf{V}^k} \alpha_i^k &= \sum_{i\in\mathbf{V}^k} \hat{y}_i(V_0^1,\dots,V_{0^{k-2}}^{k-1},C) - \sum_{i\in\mathbf{V}^k} y_i^R(\mathbf{V}_0^{k+1},C) \\
&= \sum_{j=1}^{k-1} m(V_{0^{j-1}}^j,C) - \sum_{i\in\mathbf{V}^k} y_i^R(\mathbf{V}_0^{k+1},C), && \text{(BB of } \hat{y}) \\
\sum_{i\in V^k} \tilde{\alpha}_i^k &= \sum_{i\in V^k} y_i^R(\mathbf{V}_0^{k+1},C) - \sum_{i\in V^k} y_i^R(V_{0^{k-1}}^k,C) && \text{(SCM of } R) \\
&= \sum_{i\in V^k} y_i^R(\mathbf{V}_0^{k+1},C) - m(V_{0^{k-1}}^k,C). && \text{(BB of } R)
\end{aligned}$$

Hence, it suffices to prove that,

$$\begin{aligned}
\sum_{j=1}^{k-1} m(V_{0^{j-1}}^j,C) + m(V_{0^{k-1}}^k,C) &\geq \sum_{i\in V^k} y_i^R(\mathbf{V}_0^{k+1},C) + \sum_{i\in\mathbf{V}^k} y_i^R(\mathbf{V}_0^{k+1},C) \\
&= \sum_{i\in\mathbf{V}^{k+1}} y_i^R(\mathbf{V}_0^{k+1},C).
\end{aligned}$$

Thus, by BB of R, it suffices to prove

$$\sum_{j=1}^{k} m(V_{0^{j-1}}^j,C) \geq m(\mathbf{V}_0^{k+1},C),$$

which is the case by definition. □

Furthermore, the next lemma tells us that the difference between the summations of the previous lemma is less than or equal to the inefficiency of the global tree in round k.

Lemma 8. *If R and $\hat{y}$ satisfy BB, then for all $k \leq K$, we have*

$$\sum_{i\in\mathbf{V}^k} \alpha_i^k - \sum_{i\in V^k} \tilde{\alpha}_i^k \leq I(G^k).$$

Proof. From the definitions of α_i^k and $\tilde{\alpha}_i^k$ and the proof of Lemma 7 we have,

$$\begin{aligned}\sum_{i\in\mathbf{V}^k}\alpha_i^k-\sum_{i\in V^k}\tilde{\alpha}_i^k &\leq \sum_{i\in\mathbf{V}^k}\hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C)+\sum_{i\in V^k}y_i^R(V_{0^{k-1}}^k,C)\\ &\quad-\sum_{i\in\mathbf{V}^{k+1}}y_i^R(\mathbf{V}_0^{k+1},C)\\ &=\sum_{j=1}^{k-1}m(V_{0^{j-1}}^j,C)+\sum_{i\in V^k}y_i^R(V_{0^{k-1}}^k,C) &&\text{(BB of }\hat{y})\\ &\quad-\sum_{i\in\mathbf{V}^{k+1}}y_i^R(\mathbf{V}_0^{k+1},C)\\ &=\sum_{j=1}^{k}m(V_{0^{j-1}}^j,C)-m(\mathbf{V}_0^{k+1},C)=I(G^k). &&\text{(BB of }R)\end{aligned}$$

□

We can now define the *total amount of payments* P. Given that P is the total amount of payments, it is also the total amount of reimbursements. Thus,

$$P=\sum_{i\in V^k}p_i^k=\sum_{i\in\mathbf{V}^k}r_i^k.$$

Since we want to reimburse the previously connected agents as much as possible we define

$$P=\begin{cases}b(V^k) & \text{if } b(V^k)\leq\sum_{i\in\mathbf{V}^k}\alpha_i^k\\ \sum_{i\in\mathbf{V}^k}\alpha_i^k & \text{if } b(V^k)>\sum_{i\in\mathbf{V}^k}\alpha_i^k\end{cases}.$$

We want to distribute P equally among the agents in $\mathbf{V}^k$. This way we make sure that, after the reimbursements are done, all agents in $\mathbf{V}^k$ can be treated equally in the next round. However, it is not always possible to distribute P equally over the agents in $\mathbf{V}^k$ by reimbursing each agent with $r_i^k=\frac{P}{|\mathbf{V}^k|}$. It is possible that $\frac{P}{|\mathbf{V}^k|}>\alpha_i^k$ for some agent $i\in\mathbf{V}^k$, for example, when for agent i, $\alpha_i^k=0$.

In the next section we define several policies which describe the reimbursement for each previously connected agent in round k. Recall that we want to reimburse the previously connected agents as much as possible, but that they should not be charged less than their cost share in the optimal situation.

5.2.3 Different policies

In this section we propose different policies in order to decide the exact extra payments and reimbursements. First, we propose some policies according to which the extra cost share p_i^k of agents $i \in V^k$ is determined. Second, we propose some policies according to which the reimbursement r_i^k of agents $i \in \mathbf{V}^k$ is decided.

For agents joining in round k, we assume that every agent is charged at least the lower bound $\tilde{\alpha}_i^k$. The policy then decides the amount which will be charged on top of $\tilde{\alpha}_i^k$ not exceeding the upper bound μ_i^k. Two possible ways of charging the agents who enter in round k extra are listed below. The first policy charges everyone the same value as far as this is possible considering the upper bound. The second charges the agents who have low cost in the optimal situation, according to charge rule R, more than the ones who have high costs in the optimal situation.

1. (Same value paying)
   $p_i^k = \tilde{\alpha}_i^k + \min\{p^*, \mu_i^k - \tilde{\alpha}_i^k\}$, where $\sum_{i \in V^k} \left(\tilde{\alpha}_i^k + \min\{p^*, \mu_i^k - \tilde{\alpha}_i^k\}\right) = P$. This is well-defined since $P > \sum_{i \in V^k} \mu_i^k$ is impossible.

2. (Egalitarian paying)
   $p_i^k = \tilde{\alpha}_i^k + \gamma_i^k$ where γ_i^k depends on $y_i^R(\mathbf{V}_0^{k+1}, C)$. First arrange the agents in increasing order according to $y_i^R(\mathbf{V}_0^{k+1}, C)$. Agent i with $y_i^R(\mathbf{V}_0^{k+1}, C)$ minimal will have $\gamma_i^k = \mu_i^k - \tilde{\alpha}_i^k$. Then, the second agent j in the order will have $\gamma_j^k = \mu_j^k - \tilde{\alpha}_j^k$. This continues if $\sum_{i \in V^k}(\tilde{\alpha}_i^k + \gamma_i^k) < P$. At some point $\sum_{i \in V^k}(\tilde{\alpha}_i^k + \gamma_i^k) \geq P$, then the agent ℓ who is considered at that moment will be charged γ_ℓ^k such that $\sum_{i \in V^k}(\tilde{\alpha}_i^k + \gamma_i^k) = P$. The agents who come later in the ordering than agent ℓ have $\gamma_i^k = 0$.[2]

It is easy to see that $\sum_{i \in V^k} p_i^k$ is equal to P and that p_i^k does not exceed the upper bound μ_i^k in both cases.

An equal distribution of the extra cost shares among the agents in $\mathbf{V}^k$ can be defined in different ways. We specify three possible distributions of P. Following the first policy, the agents are reimbursed proportional to their upper bound. Considering the second, everyone receives the same value as far as this is possible, considering the upper bound. According to the last policy, agents who can be reimbursed a lot, i.e., these that have high upper bounds α_i^k, will be reimbursed first and as much as possible.

1. (Proportional distribution)
   $r_i^k = \rho \cdot \alpha_i^k$ where $\rho = \frac{P}{\sum_{i \in \mathbf{V}^k} \alpha_i^k} \leq 1$.

2. (Same value distribution)
   $r_i^k = \min\{\alpha_i^k, r^*\}$, where $\sum_{i \in \mathbf{V}^k} \min\{\alpha_i^k, r^*\} = P$.

[2] Instead of the payment in the optimal case given by charge rule R, we can also consider μ_i^k or $b(i)$, meaning that if someone's upper bound on the extra cost share (or individual benefit) is maximal, then the extra cost share p_i^k should be maximal too.

3. (Egalitarian distribution)
   $r_i^k = \beta_i^k$ where β_i^k depends on α_i^k. Arrange the agents in decreasing order according to α_i^k. The agents with α_i^k maximal will be reimbursed the most, thus set β_i^k as large as possible, i.e., $\beta_i^k = \alpha_i^k$. Continue with the agent who is second in the ordering, etc. At some point $\sum_{i \in \mathbf{V}^k} \beta_i^k \geq P$, for the last considered agent set β_i^k such that $\sum_{i \in \mathbf{V}^k} \beta_i^k = P$ and for the agents who come later in the ordering set $\beta_i^k = 0$.

It is easy to see that in all cases $\sum_{i \in \mathbf{V}^k} r_i^k$ is equal to P and that r_i^k does not exceed the upper bound α_i^k.

5.2.4 Procedure

We present a step-by-step procedure to summarize the cost allocation proposed by the approach *reimbursement of previously connected agents*. The corresponding charge rule will be introduced in the next section.

Before the agents of round 1 enter the network we select a charge rule R. Then, in round $k \leq K$, we continue with the following steps:

- Allocate the cost for agents of V^k according to charge rule R.
- Calculate α_i^k for all $i \in \mathbf{V}^k$.
- Determine $b(V^k)$ and let $P = \begin{cases} b(V^k) & \text{if } b(V^k) \leq \sum_{i \in \mathbf{V}^k} \alpha_i^k \\ \sum_{i \in \mathbf{V}^k} \alpha_i^k & \text{if } b(V^k) > \sum_{i \in \mathbf{V}^k} \alpha_i^k \end{cases}$.
- Choose a reimbursement policy from Section 5.2.3 and determine r_i^k for all $i \in \mathbf{V}^k$.
- Calculate $\tilde{\alpha}_i^k$ and μ_i^k for all $i \in V^k$.
- Choose a payment policy from Section 5.2.3 and determine p_i^k for $i \in V^k$.

This gives the new cost shares for all agents in $\mathbf{V}^{k+1}$ in round k.

5.3 Charge rule

The charge rule will be applied in each round. It provides, after new agents enter the network, the cost share for each new agent and gives the possible reduced cost share of agents who connected in previous rounds. The agents who will connect in later rounds are not charged anything yet. The charge rule for the iterative problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$ is defined below. For $k \leq K$, r_i^k is defined for agents $i \in \mathbf{V}^k$ and p_i^k is defined for agents $i \in V^k$ by one of the policies of Section 5.2.3.

Definition 14 (Charge rule for Reimbursement of previously connected agents)**.**
Given any iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, a charge rule R for the classical mcst problem and variables p_i^k and r_i^k as defined before, the charge rule is defined as follows, for $k = 1$,

$$\hat{y}_i(V_0^1, C) = y_i^R(V_0^1, C) \text{ if } i \in V^1$$

and for $2 \leq k \leq K$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \begin{cases} \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k & \text{if } i \in \mathbf{V}^k \\ y_i^R(V_{0^{k-1}}^k, C) + p_i^k & \text{if } i \in V^k \end{cases}.$$

Example

In this section we consider an example, similar to the one presented in Chapter 2 in Figure 2.1, of the charge rule of Definition 14. For charge rule R we select the folk solution. The iterative mcst problem is given by the following partition of the agents: $V^1 = \{1\}$ and $V^2 = \{2, 3\}$ with cost matrix
$$C = \begin{pmatrix} 0 & 10 & 5 & 11 \\ 10 & 0 & 2 & 1 \\ 5 & 2 & 0 & 4 \\ 11 & 1 & 4 & 0 \end{pmatrix} \text{ and thus } C^* = \begin{pmatrix} 0 & 5 & 5 & 5 \\ 5 & 0 & 2 & 1 \\ 5 & 2 & 0 & 2 \\ 5 & 1 & 2 & 0 \end{pmatrix}.$$
Then,

$$\begin{aligned} &\hat{y}_1(V_0^1, C) = y_1^R(V_0^1, C) = 10 \\ &\hat{y}_1(V_0^1, V_{0^1}^2, C) = 10 - r_1^2 \\ &\hat{y}_2(V_0^1, V_{0^1}^2, C) = y_2^R(V_0^1, V_{0^1}^2, C) + p_2^2 = 2 + p_2^2 \\ &\hat{y}_3(V_0^1, V_{0^1}^2, C) = y_3^R(V_0^1, V_{0^1}^2, C) + p_3^2 = 1 + p_3^2.. \end{aligned}$$

Since $y^{\varphi}(\mathbf{V}_0^3, C) = (\frac{5}{2}, 3, \frac{5}{2})$, we have

$$\begin{aligned} &\alpha_1^2 = 10 - \frac{5}{2} = \frac{15}{2}, \text{ so } 0 \leq r_1^2 \leq \frac{15}{2}, \\ &\tilde{\alpha}_2^2 = 3 - 2 = 1 \text{ and } \mu_2^2 = \frac{9}{2} - 2 = \frac{5}{2}, \text{ so } 1 \leq p_2^2 \leq \frac{5}{2}, \\ &\tilde{\alpha}_3^2 = \frac{5}{2} - 1 = \frac{3}{2} \text{ and } \mu_3^2 = \frac{9}{2} - 1 = \frac{7}{2}, \text{ so } \frac{3}{2} \leq p_3^2 \leq \frac{7}{2}. \end{aligned}$$

Since the previously connected agents will be reimbursed as much as possible and $\tilde{\alpha}_2^2 + \tilde{\alpha}_3^2 < \alpha_1^2$, we get

$$p_2^2 = \frac{5}{2}, \quad p_3^2 = \frac{7}{2}, \quad r_1^2 = \frac{5}{2} + \frac{7}{2} = 6.$$

Thus, agents 2 and 3 are both charged the upper bounds μ_2^2 and μ_3^2 on what they can be charged extra and agent 1 is reimbursed as much as possible. The cost allocation after the second round is,

$$\hat{y}(V_0^1, V_{0^1}^2, C) = (4, 4\frac{1}{2}, 4\frac{1}{2}).$$

5.4 Properties

In Chapter 2 several classical properties were stated and in Chapter 3 we defined iterative properties. Furthermore, we discussed why one would like the charge rule to satisfy those properties in order to call the charge rule fair. In this section, we first prove that the basic properties are satisfied by the charge rule $\hat{y}$. Moreover, we consider the classical properties and check whether they are satisfied by the charge rule $\hat{y}$. Finally, we discuss the iterative properties. When one of them is not satisfied by the charge rule in general, it is checked whether the property is satisfied in one of the network structures presented in Chapter 3.

5.4.1 Basic properties

The following proposition states that the basic properties are satisfied by charge rule $\hat{y}$ defined in Section 5.3.

Proposition 3. *Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, the charge rule $\hat{y}$ satisfies the basic properties NPT and BB for all $k \leq K$, if R satisfies NPT and BB.*

Proof. See Appendix II. □

5.4.2 Classical properties

Among the classical properties we consider CS to be one of the most important ones a charge rule should satisfy. A charge rule satisfying CS prevents agents from constructing their own network and thus deviating from the global tree. We require R to satisfy CS, PM and CM_0 in order for the charge rule $\hat{y}$ to satisfy CS. The proof is by induction on the number of rounds.

Proposition 4. *Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, the charge rule $\hat{y}$ satisfies CS for all $k \leq K$, if R satisfies CS, PM and CM_0.*

Proof. By induction on k.
For $k = 1$, let $S \subseteq V^1$, by CS of R,

$$\sum_{i \in S} \hat{y}_i(V_0^1, C) = \sum_{i \in S} y_i^R(V_0^1, C) \leq m(S_0, C).$$

Assume CS is satisfied for $k < n \leq K$. For $k = n$, let $S \subseteq V^n$,

$$\begin{aligned}
\sum_{i\in S} \hat{y}_i(V_0^1, \dots V_{0^{n-1}}^n, C) &= \sum_{i\in S} \left(y_i^R(V_{0^{n-1}}^n, C) + p_i^n\right) \\
&\leq \sum_{i\in S} y_i^R(V_{0^{n-1}}^n, C) + \sum_{i\in S} \mu_i^n \\
&\leq \sum_{i\in S} y_i^R(V_{0^{n-1}}^n, C) + b(S) \qquad \text{(Lemma 5)} \\
&= \sum_{i\in S} y_i^R(V_{0^{n-1}}^n, C) + m(S_0, C) - \sum_{i\in S} y_i^R(V_{0^{n-1}}^n, C) \\
&= m(S_0, C).
\end{aligned}$$

This proves that $\hat{y}$ satisfies CS. □

Remark 15. If we drop the assumption that $S \subseteq V^k$ for some $k \leq K$ and assume $S \subseteq \mathbf{V}^{k+1}$, then we cannot ensure that the charge rule $\hat{y}$ satisfies CS. A counterexample is given below.

Example 12. Let the complete graph be given as in Figure 5.1.
Let $V^1 = \{2\}$ and $V^2 = \{1\}$, then $m(V_0^1, V_{0^1}^2, C) = 3 + 1 = 4$.
Let $S = \{1, 2\}$, which is a subset of neither V^1 nor V^2. Then, $m(S_0, C) = 2$.
But, since $r_2^2 = p_1^2$ we have,

$$\sum_{i\in S} \hat{y}_i(V_0^1, V_{0^1}^2, C) = 3 - r_2^2 + 1 + p_1^2 = 4 > 2 = m(S_0, C).$$

This shows that CS is not satisfied.

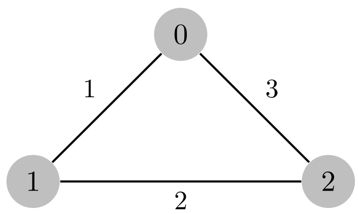

Figure 5.1: Counterexample for CS if $S \subseteq \mathbf{V}^{k+1}$.

The following theorem tells us which classical properties are satisfied by charge rule $\hat{y}$ for the iterative mcst problem, given that the charge rule R satisfies the corresponding property.

Theorem 8. *Given an iterative mcst problem $(V_0^1, \dots, V_{0^{K-1}}^K, C)$, the charge rule $\hat{y}$ satisfies CS, ESEC, IOC, SYM, ET, A, IIT, PS, IIE, RA, CON, POL, ESCR and FSCR for all $k \leq K$, if the charge rule R satisfies the corresponding property.*

Proof. See Appendix II. □

For the classical property PM, i.e., no agent is worse off with the entrance of new agents, it is an open problem whether it is satisfied by the charge rule $\hat{y}$. Since PM is considered to be an important property in the literature, which is difficult to obtain, we briefly discuss some situations in which this property is satisfied. In order for PM to be satisfied we need to prove the following inequality, assuming that R satisfies PM. Let $S \subseteq V^k$ and $i \in S$,

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = y_i^R(V_{0^{k-1}}^k, C) + p_i^k$$
$$\leq y_i^R(S_{0^{k-1}}, C) + p_i^S = \hat{y}_i(V_0^1, \ldots, S_{0^{k-1}}, C).$$

If $p_i^k = \tilde{\alpha}_i^k$, $p_i^k = \mu_i^k$ and $b(S) \leq b(V^k)$ or $p_i^S = \mu_i^S \neq 0$, then PM is satisfied by $\hat{y}$. For the proofs of these special cases we refer to Appendix II. An interesting direction for future work would be to adapt the charge rule $\hat{y}$ in such a way that PM is satisfied.

5.4.3 Iterative properties

The following theorem states which of the iterative properties defined in Section 3.3.3 are satisfied by the charge rule $\hat{y}$. The proposition thereafter shows that the iterative property W-JIT, which is not satisfied by the charge rule in general, is satisfied in a particular network structure.

Theorem 9. *Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, the charge rule $\hat{y}$ satisfies OP and R-OS (if R satisfies BB and SCM) for all $k \leq K$. It does not satisfy JIT, G-JIT and W-JIT.*

Proof. OP By definition of $\hat{y}$ and since $r_i^k \geq 0$.

R-OS Let $i \in \mathbf{V}^k$. If $r_i^k = \alpha_i^k$, then

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - \alpha_i^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - \Big(\hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) \\
&\quad - y_i^R(\mathbf{V}_0^{k+1}, C)\Big) \\
&= y_i^R(\mathbf{V}_0^{k+1}, C).
\end{aligned}$$

Let $i \in V^k$. If $p_i^k = \tilde{\alpha}_i^k$, then

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + p_i^k \\
&= y_i^R(V_{0^{k-1}}^k, C) + \tilde{\alpha}_i^k \\
&= y_i^R(V_{0^{k-1}}^k, C) + \Big(y_i^R(\mathbf{V}_0^{k+1}, C) - y_i^R(V_{0^{k-1}}^k, C)\Big) \\
&= y_i^R(\mathbf{V}_0^{k+1}, C).
\end{aligned}$$

Thus, we need to show that $r_i^k = \alpha_i^k$ for all $i \in \mathbf{V}^k$ and that $p_i^k = \tilde{\alpha}_i^k$ for all $i \in V^k$. Since $\sum_{i\in\mathbf{V}^k} r_i^k = \sum_{i\in V^k} p_i^k$, it suffices to prove the following claim.

Claim 1. If $I(G^k) = 0$, then $\sum_{i\in\mathbf{V}^k} \alpha_i^k = \sum_{i\in V^k} \tilde{\alpha}_i^k$.

Proof of Claim 1. By Lemma 7 and Lemma 8.
Thus, following the policy that every agent $i \in V^k$ pays $p_i^k = \tilde{\alpha}_i^k$ and every agent $i \in \mathbf{V}^{k+1}$ receives $r_i^k = \alpha_i^k$, we can conclude that $\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = y_i^R(\mathbf{V}_0^{k+1}, C)$ for all $i \in \mathbf{V}^{k+1}$ and $k \leq K$. □

JIT A counterexample is given in Section 3.3.3, Theorem 5 since $\hat{y}$ satisfies NPT, BB and CS.

G-JIT Since G-JIT implies JIT and JIT is not satisfied by $\hat{y}$.

W-JIT Consider Example 1 in Chapter 2 presented in Figure 2.1. Select for charge rule R the folk solution. Let the partition of the agents be $V^1 = \{3\}$, $V^2 = \{1\}$ and $V^3 = \{2\}$ and the cost matrix be as given in Figure 2.1. Then, $m(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = 14$. Following the assumption that the previously connected agents will be reimbursed as much as possible, gives rise to the following cost allocation: $\hat{y}(V_0^1, V_{0^1}^2, V_{0^2}^3, C) = (6\frac{1}{2}, 5, 2\frac{1}{2})$. If agent 1 decides to join in a later round, together with agent 2, i.e., $V'^1 = \{3\}$ and $V'^2 = \{1, 2\}$, then the total cost of the network remains the same, i.e., $m(V_0'^1, V_{0^1}'^2, C) = 14$. However, we have the following cost allocation: $\hat{y}(V_0'^1, V_{0^1}'^2, V_{0^2}'^3, C) = (3\frac{1}{2}, 3\frac{1}{2}, 7)$. This shows that agents 1 is charged $6\frac{1}{2}$ in the first situation and $3\frac{1}{2}$ in the second situation. Hence, agent 1 is better off by joining later and thus W-JIT is not satisfied.

□

The following proposition shows that if the direct connection cost to the source is high for all agents and the connection cost to other agents is low, then W-JIT is satisfied by the charge rule. To be more precise, the inefficiency of the network is charged to the agents who enter the network in the last round, given an lower bound on everyone's direct connection cost to the source. It is therefore not beneficial to join later if the total cost of the global tree increase.

Proposition 5. *Given an iterative mcst problem* $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$*, the charge rule* $\hat{y}$ *satisfies W-JIT in network structure I for all* $k \leq K$*, if R satisfies BB, PM,* CM_0 *and ESEC.*

Proof. See Appendix II. □

5.5 Discussion

The main purpose of defining the charge rule was, on the one hand, to satisfy CS and, on the other hand, to force the agents to pay as much extra as possible. The fact that CS is satisfied comes at the price of JIT and even W-JIT being dissatisfied. However, we proved that W-JIT is satisfied if, for all agents, the cost of connecting directly to the source is high and the connection costs between agents are low. An interesting question is how to change the upper bound μ_i^k in order to satisfy JIT (or W-JIT) and at the same time aiming for CS and BB being satisfied.

The cost monotonicity properties CM, CM_0 and SCM are not satisfied by $\hat{y}$. It would be interesting to consider how the definition of α_i^k should be changed in order to have these properties satisfied by $\hat{y}$. An interesting possibility for future research would be to define a charge rule which satisfies these properties, as well as PM.

By making the assumption that everyone is charged as much as possible in the round that new agents enter the network, we implicitly leave the inefficiency of the network for the agents who enter in the last round.

Considering the bounds on the extra cost shares and possible reimbursements, the classical charge rule should satisfy PM, CM_0 and SCM in order for these bounds to behave in a nice way. This is quite a strong assumption since the folk solution is the only charge rule of the ones discussed in Chapter 2 which satisfies these properties.

Chapter 6

Conclusion

In this chapter we briefly summarize the main achievements of the thesis, focusing on the comparison of the charge rules defined in Chapter 4 and Chapter 5. Furthermore, we provide some directions for further research.

The classical minimum cost spanning tree problem, especially the subproblem of distributing the cost among the agents, has been studied extensively. However, within the cooperative game theory approach there are to our knowledge no results regarding the construction of the network and distribution of the cost among the agents in an iterative setting. Chapters 3, 4 and 5 contain the original part of this thesis. In Chapter 3 we provided a novel framework for the iterative mcst problem. We presented an algorithm which constructs a network connecting the agents to the source in different rounds. In addition, the complexity of the algorithm for the iterative mcst problem has been discussed. We proved that the upper bound on the complexity of the algorithm for the iterative mcst problem is equal to the upper bound on the complexity of Prim's algorithm for the classical case, i.e., quadratic in the number of agents.

Chapters 4 and 5 constitute the main contribution of this thesis. Both chapters describe solutions to the second part of the iterative mcst problem, i.e., dividing the cost of the constructed network among the agents in a fair way. In Chapter 4, the approach *fair sharing of inefficiencies* was explained and four charge rules were defined: the inefficiency of the network constructed in round k, caused by the fact that agents join in different rounds, should be equally divided over the agents who use the network in round k. In Chapter 5, another approach, *reimbursement of previously connected agents*, was motivated: on the one hand, the agents who join in round k should reimburse the previously connected agents as much as possible. On the other hand, we want to avoid that the agents of round k deviate from the global tree because they can construct their own network with lower connection costs. A charge rule was defined with upper and lower bounds for the extra cost shares of the agents in round k and for the reimbursements of the agents who connected in previous rounds.

In order to assess the fairness of the charge rules we designed iterative properties which are particularly interesting in the iterative setting. Ideally, we would like a charge rule for the iterative mcst problem to satisfy all the basic, classical and iterative properties. However, we proved that it is impossible that BB, NPT, CS and JIT are satisfied at the same time. The next section compares the charge rules of Chapters 4 and 5 according to the properties they satisfy. Afterwards we discuss their fairness.

6.1 Comparison of the charge rules

The properties satisfied by the charge rules defined in Chapters 4 and 5 are presented in the following table:

Property	Chapter 4 (a)	(b)	(c)	(d)	Chapter 5
NPT	✓	✓	✓	✓	✓
BB	✓	✓	✓	✓	✓
CS	–	–	–	–	✓
PM	–	–	–	–	–
CM	✓	✓	$-^{\oplus}$	✓	–
CM_0	–	–	–	✓	–
SCM	$-^{*}$	$-^{*}$	$-^{*}$	✓	–
ESEC	✓	$-^{\star}$	✓	✓	✓
IOC	$-^{\dagger}$	$-^{\dagger}$	$-^{\dagger}$	$-^{\dagger}$	✓
SYM	✓	✓	✓	✓	✓
RNK	✓	✓	$-^{\oplus}$	✓	–
ET	✓	$-^{\dagger}$	–	✓	✓
A	✓	✓	✓	✓	✓
IIT	✓	✓	✓	✓	✓
SEP	–	–	–	–	–
GI	–	–	–	–	–
PS	–	$-^{\circ}$	–	✓	✓
IIE	✓	✓	$-^{\oplus}$	✓	✓
RA	$-^{\triangleright}$	$-^{\diamond}$	$-^{\triangleright}$	✓	✓
CON	✓	✓	✓	✓	✓
POL	✓	✓	✓	✓	✓
ESCR	✓	$-^{\star}$	$-^{\oplus}$	✓	✓
FSCR	✓	$-^{\star}$	$-^{\oplus}$	–	✓

Property	Chapter 4 (a)	(b)	(c)	(d)	Chapter 5
OP	–	–	–	–	✓
R-OS	✓	✓	✓	–	✓
JIT	–	–	–	–	–
G-JIT	–	–	–	–	–
W-JIT	✓	✓	✓	✓	–

The symbols indicate exceptional cases in which the property does hold:
$*$ holds only if $I(G^k) \leq I(G'^k)$ and $\beta \leq \beta'$.
$\star$ holds only if $\beta' = \beta = 1$.
† holds only if $I(G^k) = I(G'^k)$ (and thus $\beta = \beta'$).
$\diamond$ holds only if $\beta^{C+C'} = \beta^C = \beta^{C'}$.
$\circ$ holds only if $\beta = \hat{\beta} = \tilde{\beta} = \dot{\beta}$.
$\triangleright$ holds only if $I(G^{k+'k}) = I(G^k) + I(G'^k)$.
$\oplus$ holds only if $\delta_i \leq \delta_i^{'}$.

We can draw several conclusions from the property table. Simply counting the number of satisfied properties reveals that the charge rule of Chapter 5 and charge rule (d) of Chapter 4 both satisfy 18 properties. Since charge rule (d) can be seen as a somewhat naive way of allocating the cost (it just divides the total cost of the constructed network over the agents), this is a bit surprising. Notice that charge rule (d) is not able to distinguish between two agents since it charges the same to every agent. However, the rule does not satisfy CS, and neither do charge rules (a), (b) and (c). Thus, according to these rules, some agents might be better off constructing their own network without using the previously constructed one. The charge rule of Chapter 5 is defined in such a way that CS is satisfied. Recall that we need charge rule R to satisfy CM_0 and PM in order for CS to be satisfied and the folk solution is one of the few charge rules who satisfies these properties. Considering the classical properties, another difference between the charge rules is that CM, i.e., no agent can take advantage by reporting false connection cost, is satisfied by the charge rules from Chapter 4, but not by the charge rule from Chapter 5. PM is satisfied by none of the discussed charge rules. However, the charge rule of Chapter 5 does satisfy PM in some cases.

From the iterative properties, JIT is satisfied by none of the charge rules. However, W-JIT is satisfied by the charge rules from Chapter 4, but not by the charge rule of Chapter 5. OP, i.e., an agent being only charged in the round in which she joins the network, is satisfied by the charge rule of Chapter 5 and not by the charge rules of Chapter 4.

Summarizing, we can say that the main differences between the introduced charge rules lie in the fact that CS and OP are satisfied by the charge rule of Chapter 5 (and not by the charge rules of Chapter 4) and CM and W-JIT are satisfied by the charge rules of Chapter 4 (and not by the charge rule of Chapter 5).

Based on above discussion of the differences between the charge rules presented in this thesis we would like to assess their fairness. Since the charge rules differ in the properties which they do and which they do not satisfy, depending on the situation in which the charge rule will be applied, a charge rule can be considered more fair than another. Recalling examples such as the one presented in the introduction, CS is a property which should be satisfied and reimbursement of previously connected agents is therefore a suitable charge rule. Moreover, OP makes sure that agents are only charged in the round in which they join the network, which we believe to be an important fairness criterion. If CS is not a very important property, e.g., because the chance of agents constructing their own network is very low, one of the charge rules from Chapter 4 may be more suitable. These satisfy at least W-JIT which prevents agents from entering later if the inefficiency of the network stays the same.

6.2 Future work

For further research we propose different directions. One direction would be to adapt the charge rules for the iterative mcst problem proposed in this thesis, or define new charge rules, that satisfy more properties. The properties JIT and PM contribute most to the fairness of the defined charge rules. A charge rule which satisfies JIT should charge every agent the same in each round, independent of the round in which she joins, or an agent should be charged more when she comes later, for example, by giving a penalty to later coming agents. In order for JIT to be satisfied we can also think of relaxing the basic assumption BB to β-BB. This way, more than the necessary costs of the network are covered, and this would help especially in the later rounds. PM is considered to be an important property in the literature on the mcst problem but it is not fully satisfied by any of the proposed charge rules. It would be interesting to see how the charge rule in Chapter 5 can be adapted in order to satisfy PM.

We can further investigate new properties in order to obtain a better understanding of which charge rule to best use in which situation. An example of a property is provided by Bergantiños and Vidal-Puga (2008). They introduce the property *strategic merging*: no group of agents will benefit from acting as one single node. Moulin (2009) defined a similar property called *routing-proofness*. Trudeau (2009) considers the influence of an agent joining a coalition. If an agent does not bring anything to the coalition, then this agent should not gain from the cooperation. The influence of agent i can be measured in terms of the difference between the benefit of a coalition including and excluding agent i. Taking the influence of agents in a certain coalition into consideration while deciding the cost allocation would constitute an extra criterion for devising a fair charge rule.

In the literature several variants of the classical mcst problem are proposed. Our framework for the iterative mcst problem constitutes an interesting theoretical setting which can be applied to different variants of the classical mcst problem. We propose some alternatives: the mcst problem with groups (Bergantiños and Gómez-Rúa, 2010); the multi-criteria mcst game (Fernández et al., 2004); the mcst problem with multiple sources (Moulin, 2013); the mcst problem with indifferent agents (Trudeau, 2013d); the mcst problem with capacities on the edges or budget restrictions (Bergantiños and Lorenzo, 2008a; Bogomolnaia et al., 2010; Moulin, 2013; Moulin and Laigret, 2011).

Adding utilities to the framework for the iterative mcst problem constitutes a worthwhile direction of future work. This way we can express the differences between a situation in which an agent desires to be connected but thinks that the cost for connecting is currently too high and a situation in which the agent waits for some time and connects later for lower cost. It is possible to compare our framework with the related models discussed in Section 3.1.2 by enlarging the framework with utilities. The work of Angel et al. (2006) is related to this suggestion. They propose two cost sharing's methods minimizing the maximum or average dissatisfaction of the agents for the classical mcst problem. It would be interesting to define the height of dissatisfaction of the agents for the iterative mcst problem.

An important area for future research is the modification of the algorithm for the iterative mcst problem in such a way that constraints on the length of the path or connectivity of the vertices can be taken into account. It might be possible to construct minimal networks satisfying these constraints, as given in Section 3.4 by network structures IV and V, with lower complexity if the number of rounds in the iterative mcst problem is fixed. Future work on the complexity of the algorithm will be required in order to affirm or refute this. We do not allow for existing edges to be rebuilt when a new group of agents joins the network. It is worthwhile to analyze a situation in which this is allowed and investigate the possibilities for charge rules.

Perhaps the most challenging issue that deserves further research is the application of the algorithm and charge rules for the iterative mcst problem to real-life problems. As a first step it would be interesting to implement the algorithm and charge rules and apply it to different empirical situations.

Proofs Chapter 4

Proposition 1

Proof.

NPT By definition of charge rules (a), (b), (c) and (d) since $I(G^k) \geq 0$ and NPT of R.

BB (a)

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \sum_{i \in \mathbf{V}^{k+1}} y_i^R(\mathbf{V}_0^{k+1}, C) + \sum_{i \in \mathbf{V}^{k+1}} \frac{I(G^k)}{|\mathbf{V}^{k+1}|}$$
$$= m(\mathbf{V}_0^{k+1}, C) + I(G^k) = \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

(b)

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \sum_{i \in \mathbf{V}^{k+1}} \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C) = \beta \cdot m(\mathbf{V}_0^{k+1}, C)$$
$$= \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

(c)

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \sum_{i \in \mathbf{V}^{k+1}} y_i^R(\mathbf{V}_0^{k+1}, C) + \sum_{i \in \mathbf{V}^{k+1}} \frac{I(G^k)}{|\mathbf{V}^{k+1}|}\delta_i$$
$$= m(\mathbf{V}_0^{k+1}, C) + I(G^k) = \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

(d)

$$\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) = \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} = \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C).$$

□

Theorem 6

The proofs are by induction on k, base cases are trivial and therefore not stated here. The proofs rely on R satisfying the considered property and the definition of inefficiency. Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, let $\hat{y}$ be defined as in Definition 9 and let R satisfy the corresponding property. Then, for all $k \leq K$,

$\hat{y}$ does not satisfy CS.

Proof. Let $V^1 = \{2\}$ and $V^2 = \{1\}$, select R to be the folk solution and let the complete graph be given in Figure 6.1.

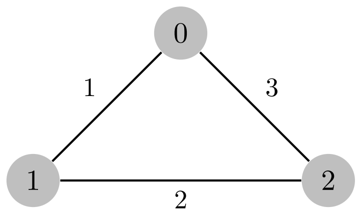

Figure 6.1: CS not satisfied by charge rules (a)-(d).

This gives $I(G^k) = m(V_0^1, C) + m(V_{0^1}^2, C) - m(\mathbf{V}_0^3, C) = 3 + 1 - 3 = 1$ and $\beta = \frac{4}{3}$. Let $S = V^2$, then we have

(a) $\hat{y}_1(V_0^1, V_{0^1}^2, C) = \frac{3}{2} + \frac{1}{2} = 2 > 1 = m(V_0^2, C)$.

(b) $\hat{y}_1(V_0^1, V_{0^1}^2, C) = \frac{4}{3} \cdot \frac{3}{2} = 2 > 1 = m(V_0^2, C)$.

(c) $\hat{y}_1(V_0^1, V_{0^1}^2, C) = \frac{3}{2} + \frac{1}{2} \cdot \delta_1 > \frac{3}{2} > 1 = m(V_0^2, C)$.

(d) $\hat{y}_1(V_0^1, V_{0^1}^2, C) = \frac{4}{2} = 2 > 1 = m(V_{0^1}^2, C)$.

Thus, for charge rules (a)-(d) CS is not satisfied. □

$\hat{y}$ satisfies CM, except for rule (c).

Proof. Let $i \in V^k$, $j \in V_{0^{k-1}}^k$.

(a)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \\
&\leq y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} && \text{(CM of } R\text{)} \\
&\leq y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && \text{(Lemma 9)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

(b)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C) \\
&\leq \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && \text{(CM of } R\text{)} \\
&\leq \beta' \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && \text{(Corollary 3)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

(d)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} \\
&\leq \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} && (c_{ij} < c'_{ij}) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

Lemma 9. *If $i \in V^k$ and $j \in V_{0^{k-1}}^k$ such that $c_{ij} < c'_{ij}$ and $c_{m\ell} = c'_{m\ell}$ otherwise, then $I(G^k) \leq I(G'^k)$.*

Proof. By definition $I(G^k) = \sum_{j=1}^k m(V_{0^{j-1}}^j, C) - m(\mathbf{V}_0^{k+1}, C)$. We consider two cases:

(i) If $m(V_{0^{k-1}}^k, C) = m(V_{0^{k-1}}^k, C')$, then the edge (i, j) with cost c'_{ij} is not used in the tree constructed by the algorithm in round k with cost matrix C'. Hence, also in the optimal problem the edge (i, j) is not used, thus $m(\mathbf{V}_0^{k+1}, C) = m(\mathbf{V}_0^{k+1}, C')$. Therefore, $I(G^k) = I(G'^k)$.

(ii) If $m(V_{0^{k-1}}^k, C) \neq m(V_{0^{k-1}}^k, C')$, then an edge with cost $c^* \leq c'_{ij}$ is used in the tree constructed by the algorithm in round k with cost matrix C'. Hence, $m(V_{0^{k-1}}^k, C) = m(V_{0^{k-1}}^k, C') - c^* + c_{ij}$ and since $c_{ij} \leq c^*$ we have $m(V_{0^{k-1}}^k, C) \leq m(V_{0^{k-1}}^k, C')$. We consider two subcases:

(a) If in the optimal problem with cost matrix C the edge (i, j) is not used. Then, $m(\mathbf{V}_0^{k+1}, C) = m(\mathbf{V}_0^{k+1}, C')$ and thus

$$\begin{aligned}
I(G^k) &= \sum_{j=1}^k m(V_{0^{j-1}}^j, C) - m(\mathbf{V}_0^{k+1}, C) \\
&\leq \sum_{j=1}^k m(V_{0^{j-1}}^j, C') - m(\mathbf{V}_0^{k+1}, C') = I(G'^k).
\end{aligned}$$

(b) If in the optimal problem with cost matrix C the edge (i, j) is used. Then $m(\mathbf{V}_0^{k+1}, C) \leq m(\mathbf{V}_0^{k+1}, C') - c^{**} + c_{ij}$ such that $c_{ij} \leq c^{**} \leq c^* \leq c'_{ij}$ and thus

$$\begin{aligned} I(G^k) &= \sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C) - m(\mathbf{V}_0^{k+1}, C) \\ &= \sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C') - c^* + c_{ij} - (m(\mathbf{V}_0^{k+1}, C') - c^{**} + c_{ij}) \\ &\leq \sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C') - c^* - m(\mathbf{V}_0^{k+1}, C') + c^* \\ &= I(G'^k). \end{aligned}$$

This shows that in all cases $I(G^k) \leq I(G'^k)$. □

Corollary 3. *If $I(G^k) \leq I(G'^k)$, then $\beta \leq \beta'$.*

$\hat{y}$ satisfies CM_0 for rule (d).

Proof. (d)

$$\begin{aligned} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^{k}, C) &= \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C)}{|\mathbf{V}^{k+1}|} \\ &\leq \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C')}{|\mathbf{V}^{k+1}|} \\ &\leq \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^{k}, C'). \end{aligned}$$

□

$\hat{y}$ satisfies SCM for rule (d).

Proof.

$$\begin{aligned} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^{k}, C) &= \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C)}{|\mathbf{V}^{k+1}|} \\ &\leq \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^{j}, C')}{|\mathbf{V}^{k+1}|} \\ &\leq \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^{k}, C'). \end{aligned}$$

□

$\hat{y}$ satisfies ESEC, except for rule (b).

Proof. Let $i \in \mathbf{V}^{k+1}$.

(a)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} \\
&= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{c_0' - c_0}{|\mathbf{V}^{k+1}|} + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && \text{(ESEC of } R\text{)} \\
&= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{c_0' - c_0}{|\mathbf{V}^{k+1}|} + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} && (I(G^k) \overset{*}{=} I(G'^k)) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \frac{c_0' - c_0}{|\mathbf{V}^{k+1}|}.
\end{aligned}$$

*:$I(G^k) = I(G'^k)$.

$$\begin{aligned}
I(G^k) &= \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C) - m(\mathbf{V}_0^{k+1}, C) \\
&= \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C') - (c_{0i}' - c_{0i}) - m(\mathbf{V}_0^{k+1}, C) + (c_{0i}' - c_{0i}) \\
&\text{(by conditions of ESEC)} \\
&= \sum_{j=1}^{k} m(V_{0^{j-1}}^j, C') - m(\mathbf{V}_0^{k+1}, C') = I(G'^k).
\end{aligned}$$

(c) Similar to the proof of (a).

(d)

$$\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') = \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} = \frac{\sum_{j=1}^{k} m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} + \frac{c_0' - c_0}{|\mathbf{V}^{k+1}|}.$$

□

$\hat{y}$ satisfies SYM.

Proof. Let $i, j \in V^k$.

(a)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \\
&= y_j^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} && \text{(SYM of } R\text{)} \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-1}}^k, C).
\end{aligned}$$

(b)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= \beta\cdot y_i^R(\mathbf{V}_0^{k+1},C)\\ &= \beta\cdot y_j^R(\mathbf{V}_0^{k+1},C) && \text{(SYM of } R\text{)}\\ &= \hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).\end{aligned}$$

(c)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(\mathbf{V}_0^{k+1},C)+\frac{I(G^k)}{|\mathbf{V}^{k+1}|}\cdot\delta_i\\ &= y_j^R(\mathbf{V}_0^{k+1},C)+\frac{I(G^k)}{|\mathbf{V}^{k+1}|}\cdot\delta_j && \text{(SYM of } R \text{ and } \delta_i=\delta_j\text{)}\\ &= \hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).\end{aligned}$$

(d)

$$\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C)=\frac{\sum_{j=1}^k m(V_{0^{j-1}}^j,C)}{|\mathbf{V}^{k+1}|}=\hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).$$

Let $i\in V^q\subseteq\mathbf{V}^{k+1}, j\in V^{q'}\subseteq V^{k+1}$ for $q\neq q'$.
The proof is similar to the previous one since $I(G^k),\beta,\delta_i,\delta_j$ are not different for agents from different rounds. □

$\hat{y}$ satisfies RNK, except for rule (c).

Proof. Let $i,j\in V^k$.

(a)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(\mathbf{V}_0^{k+1},C)+\frac{I(G^k)}{|\mathbf{V}^{k+1}|}\\ &\leq y_j^R(\mathbf{V}_0^{k+1},C)+\frac{I(G^k)}{|\mathbf{V}^{k+1}|} && \text{(RNK of } R\text{)}\\ &= \hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).\end{aligned}$$

(b)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= \beta\cdot y_i^R(\mathbf{V}_0^{k+1},C)\\ &\leq \beta\cdot y_j^R(\mathbf{V}_0^{k+1},C) && \text{(RNK of } R\text{)}\\ &= \hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).\end{aligned}$$

(d)

$$\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C)=\frac{\sum_{j=1}^k m(V_{0^{j-1}}^j,C)}{|\mathbf{V}^{k+1}|}=\hat{y}_j(V_0^1,\ldots,V_{0^{k-1}}^k,C).$$

Let $i \in V^q \subseteq \mathbf{V}^{k+1}, j \in V^{q'} \subseteq \mathbf{V}^{k+1}$ such that $q \neq q'$.
The proof is similar to the previous one since $I(G^k), \beta$ are not different for agents from different rounds. □

$\hat{y}$ satisfies ET for rule (a) and (d).

Proof. Let $i, j \in V^k$.

(a)

$$\begin{aligned}
\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C') &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \\
&- \left(y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} \right) \\
&= y_j^R(\mathbf{V}_0^{k+1}, C) - y_j^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G^k) - I(G'^k)}{|\mathbf{V}^{k+1}|} \quad \text{(ET of } R) \\
&= \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C').
\end{aligned}$$

(d) Trivial.

Let $i \in V^q \subseteq \mathbf{V}^{k+1}$ and $j \in V^{q'} \subseteq V^{k+1}$ such that $q \neq q'$.
The proof is similar to the previous one since $I(G^k)$ is not different for agents from different rounds. □

$\hat{y}$ satisfies A.

Proof. (a)

$$\begin{aligned}
\pi(\hat{y}(V_0^1, \dots, V_{0^{k-1}}^k, C)) &= \pi \left(y^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \right) \\
&= y^R(\mathbf{V}_0^{k+1}, \pi C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \qquad \text{(A of } R) \\
&= \hat{y}(V_0^1, \dots, V_{0^{k-1}}^k, \pi C).
\end{aligned}$$

(b)

$$\begin{aligned}
\pi(\hat{y}(V_0^1, \dots, V_{0^{k-1}}^k, C)) &= \pi \left(\beta \cdot y^R(\mathbf{V}_0^{k+1}, C) \right) \\
&= \beta \cdot y^R(\mathbf{V}_0^{k+1}, \pi C) \qquad \text{(A of } R) \\
&= \hat{y}(V_0^1, \dots, V_{0^{k-1}}^k, \pi C).
\end{aligned}$$

(c) Similar to the proof of (a) since $\pi \delta_{\pi_i} = \delta_i$.

(d) Trivial.

□

$\hat{y}$ satisfies IIT.

Proof. Let $i \in \mathbf{V}^{k+1}$.

(a)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \\
&= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} && \text{(IIT of } R\text{)} \\
&= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && (I(G^k) \overset{*}{=} I(G'^k)) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds because of coniditions of IIT/tree-equivalence.

(b)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C) \\
&= \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && \text{(IIT of } R\text{)} \\
&= \beta' \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && (\beta = \beta') \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

(c)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \cdot \delta_i \\
&= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \cdot \delta_i && \text{(IIT of } R\text{)} \\
&= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} \cdot \delta_i' && (\delta_i \overset{*}{=} \delta_i') \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds since $y_i^R(\mathbf{V}_0^{k+1}, C) = y_i^R(\mathbf{V}_0^{k+1}, C')$.

(d)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} \\
&= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} = \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

$\hat{y}$ satisfies PS for rule (d).

Proof. Let $i \in \mathbf{V}^{k+1}$.

(d)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} \\ &\overset{*}{=} \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, \hat{C}) + \sum_{j=1}^k m(V_{0^{j-1}}^j, \tilde{C}) + \sum_{j=1}^k m(V_{0^{j-1}}^j, \dot{C})}{|\mathbf{V}^{k+1}|} \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').\end{aligned}$$

* By definition of $\hat{C}, \tilde{C}$ and $\dot{C}$. □

$\hat{y}$ satisfies IIE, except for rule (c).

Proof. Let $i \in \mathbf{V}^{k+1}$.

(a)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} \\ &= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G^k)}{|\mathbf{V}^{k+1}|} && \text{(IIE of } R\text{)} \\ &= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && (I(G^k) \overset{*}{=} I(G'^k)) \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').\end{aligned}$$

*: $I(G^k) = I(G'^k)$.
Given that $c'_{ij} > c_{ij} \geq \max\{c_{0i}, c_{0j}\}$ and $c_e = c'_e$ otherwise the edge (i, j) will never be used by the algorithm for constructing the tree. Therefore, $I(G^k) = I(G'^k)$.

(b)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C) \\ &= \beta \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && \text{(IIE of } R\text{)} \\ &= \beta' \cdot y_i^R(\mathbf{V}_0^{k+1}, C') && (\beta = \beta') \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').\end{aligned}$$

(d)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} \\ &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} = \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').\end{aligned}$$

□

$\hat{y}$ satisfies RA for rule (d).

Proof. Let $i \in \mathbf{V}^{k+1}$.

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C + C') &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C + C')}{|\mathbf{V}^{k+1}|} \\ &\overset{*}{=} \frac{\sum_{j=1}^k (m(V_{0^{j-1}}^j, C) + m(V_{0^{j-1}}^j, C'))}{|\mathbf{V}^{k+1}|} \\ &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} + \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').\end{aligned}$$

* holds by conditions of RA. □

$\hat{y}$ satisfies CON.

Proof. By CON of R. □

$\hat{y}$ satisfies POL.

Proof. By POL of R. □

$\hat{y}$ satisfies ESCR, for rule (a) and (d).

Proof. Let $i \in \mathbf{V}^{k+1}$.

(a)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} \\ &= y_i^R(\mathbf{V}_0^{k+1}, C) + \frac{x}{|\mathbf{V}^{k+1}|} + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && \text{(ESCR of } R) \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \frac{x}{|\mathbf{V}^{k+1}|} && (I(G^k) \overset{*}{=} I(G'^k))\end{aligned}$$

* holds by conditions of ESCR.

(d)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C')}{|\mathbf{V}^{k+1}|} \\ &= \frac{\sum_{j=1}^k m(V_{0^{j-1}}^j, C)}{|\mathbf{V}^{k+1}|} + \frac{x}{|\mathbf{V}^{k+1}|}.\end{aligned}$$

□

$\hat{y}$ satisfies FSCR for rule (a).

Proof. Let $i \in \mathbf{V}^{k+1}$.

(a)

$$\begin{aligned}\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} \\ &= y_i^R(\mathbf{V}_0^{k+1}, C) - x + \frac{I(G'^k)}{|\mathbf{V}^{k+1}|} && \text{(FSCR of } R) \\ &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - x && (I(G^k) \stackrel{*}{=} I(G'^k))\end{aligned}$$

* holds by conditions of FSCR.

□

Proofs Chapter 5

Proposition 3

Proof. NPT: Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k \\
&\geq \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - \alpha_i^k && (r_i^k \leq \alpha_i^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) \\
&- (\hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - y_i^R(\mathbf{V}_0^{k+1}, C)) \\
&= y_i^R(\mathbf{V}_0^{k+1}, C) \geq 0 && \text{(POS of } R\text{)}.
\end{aligned}$$

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + p_i^k \\
&\geq p_i^k && \text{(POS of } R\text{)} \\
&\geq 0 && (p_i^k \geq 0).
\end{aligned}$$

BB: By induction on k. The base case is trivial and therefore omitted.

$$\begin{aligned}
\sum_{i \in \mathbf{V}^{k+1}} \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \sum_{i \in \mathbf{V}^k} \left(\hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k\right) \\
&+ \sum_{i \in V^k} \left(y_i^R(V_{0^{k_1}}^k, C) + p_i^k\right) \\
&= \sum_{i \in \mathbf{V}^k} \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) + \sum_{i \in V^k} y_i^R(V_{0^{k_1}}^k, C) && (*) \\
&= \sum_{i \in \mathbf{V}^k} \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) + m(V_{0^{k-1}}^k, C) && \text{(BB of } R\text{)} \\
&= \sum_{j=1}^{k-1} m(V_{0^{k-1}}^j, C) + m(V_{0^{k-1}}^k, C) && \text{(I.H.)} \\
&= \sum_{j=1}^{k} m(V_{0^{k-1}}^j, C).
\end{aligned}$$

*: $\sum_{i \in \mathbf{V}_0^k} r_i^k = \sum_{i \in V^k} p_i^k$. □

Theorem 8

The proofs are by induction on the number of rounds, base cases are trivial and therefore not stated. Most of the proofs consist of two parts, one of them considers the case for agents $i \in \mathbf{V}^k$, the other considers the case for agents $i \in V^k$. In most of the proofs we rely on the definitions of α_i^k, $\tilde{\alpha}_i^k$ and μ_i^k in order to say that r_i^k and p_i^k behave in the right way. This is because the definitions of α_i^k, $\tilde{\alpha}_i^k$ and μ_i^k depend on the charge rule R which satisfies the considered property.

Given an iterative mcst problem $(V_0^1, \ldots, V_{0^{K-1}}^K, C)$, let $\hat{y}$ be defined as in Definition 14 and let R satisfy the considered property. Then, for all $k \leq K$,

$\hat{y}$ satisfies PM if

(i) $p_i^k = \tilde{\alpha}_i^k = 0$

(ii) $p_i^k = \tilde{\alpha}_i^k \neq 0$ and R satisfies SCM

(iii) $p_i^k = \mu_i^k \neq 0$ and $b(S) \leq b(V^k)$

(iv) $p_i^S = \mu_i^S \neq 0$.

Proof. Let $S \subseteq V^k$.

(i)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + \tilde{\alpha}_i^k \\
&= y_i^R(V_{0^{k-1}}^k, C) \\
&\leq y_i^R(S_{0^{k-1}}, C) + p_i^S \qquad \text{(PM of } R \text{ and } p_i^S \geq 0) \\
&= \hat{y}_i(V_0^1, \ldots, S_{0^{k-1}}, C).
\end{aligned}$$

(ii)

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + \tilde{\alpha}_i^k \\
&= y_i^R(\mathbf{V}_0^{k+1}, C) \\
&\leq y_i^R((\mathbf{V}^k \cup S)_0, C) \qquad \text{(PM of } R) \\
&= y_i^R(S_{0^{k-1}}, C) + \tilde{\alpha}_i^S \qquad \text{(SCM of } R) \\
&\leq y_i^R(S_{0^{k-1}}, C) + p_i^S \qquad (p_i^S \geq \tilde{\alpha}_i^S) \\
&= \hat{y}_i(V_0^1, \ldots, S_{0^{k-1}}, C).
\end{aligned}$$

(iii)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(V_{0^{k-1}}^k,C)+\mu_i^k \\ &= y_i^R(V_0^k,C) \\ &\le y_i^R(S_0,C) && \text{(PM of } R) \\ &= y_i^R(S_{0^{k-1}},C)+\mu_i^S && (b(S)\le b(V^k)) \\ &= \hat{y}_i(V_0^1,\ldots,S_{0^{k-1}},C).\end{aligned}$$

(iv)

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(V_{0^{k-1}}^k,C)+p_i^k \\ &\le y_i^R(V_{0^{k-1}}^k,C)+\mu_i^k \\ &= y_i^R(V_0^k,C) \\ &\le y_i^R(S_0,C) && \text{(PM of } R) \\ &= y_i^R(S_{0^{k-1}},C)+\mu_i^S \\ &= \hat{y}_i(V_0^1,\ldots,S_{0^{k-1}},C) && (p_i^S=\mu_i^S).\end{aligned}$$

□

$\hat{y}$ satisfies ESEC.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C') &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C')-r_i'^k \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C)+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|}-r_i'^k && \text{(I.H.)} \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C)-r_i^k+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|} && (\alpha_i^k \overset{*}{=} \alpha_i'^k) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C)+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|}.\end{aligned}$$

* holds by I.H. and ESEC of R.

Let $i \in V^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C') &= y_i^R(V_{0^{k-1}}^k,C')+p_i'^k \\ &= y_i^R(V_{0^{k-1}}^k,C)+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|}+p_i'^k && \text{(ESEC of } R) \\ &= y_i^R(V_{0^{k-1}}^k,C)+p_i^k+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|} && (p_i^k=p_i'^k) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C)+\frac{c_0'-c_0}{|\mathbf{V}^{k+1}|}.\end{aligned}$$

□

$\hat{y}$ satisfies IOC.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') - r_i^k && \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') - r_i'^k && (\alpha_i^k \overset{*}{=} \alpha_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds by I.H. and IOC of R.

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + p_i^k \\
&= y_i^R(V_{0^{k-1}}^k, C') + p_i^k && \text{(IOC of } R) \\
&= y_i^R(V_{0^{k-1}}^k, C') + p_i'^k && (p_i^k = p_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

$\hat{y}$ satisfies SYM.

Proof. Let $i, j \in V^q \subseteq \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_j^k && (\text{I.H. and } \alpha_i^k \overset{*}{=} \alpha_j^k) \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-1}}^k, C).
\end{aligned}$$

Let $i, j \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + p_i^k \\
&\leq y_j^R(V_{0^{k-1}}^k, C) + p_i^k && \text{(SYM of } R) \\
&\leq y_j^R(V_{0^{k-1}}^k, C) + p_j^k && (p_i^k = p_i'^k) \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-1}}^k, C).
\end{aligned}$$

Proof of $*$:

$$\begin{aligned}
\alpha_i^k &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - y_i^R(\mathbf{V}_0^{k+1}, C) \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - y_i^R(\mathbf{V}_0^{k+1}, C) && \text{(I.H.)} \\
&= \hat{y}_j(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - y_j^R(\mathbf{V}_0^{k+1}, C) && \text{(SYM of } R) \\
&= \alpha_j^k.
\end{aligned}$$

□

$\hat{y}$ satisfies ET.

Proof. Let $i, j \in V^q \subseteq \mathbf{V}^k$.

$$\begin{aligned}
&\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C') \\
&\quad = \hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - r_i^k - \hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C') + r_i'^k \\
&\quad = \hat{y}_j(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - \hat{y}_j(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C') - r_i^k + r_i'^k \qquad \text{(I.H.)} \\
&\quad = \hat{y}_j(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - \hat{y}_j(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C') - r_j^k + r_j'^k \quad (\alpha_i'^k - \alpha_i^k \overset{*}{=} \alpha_j'^k - \alpha_j^k) \\
&\quad = \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds by I.H. and ET of R.

Let $i, j \in V^k$.

$$\begin{aligned}
&\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C') \\
&\quad = y_i^R(V_{0^{k-1}}^k, C) + p_i^k - y_i^R(V_{0^{k-1}}^k, C') - p_i'^k \\
&\quad = y_j^R(V_{0^{k-1}}^k, C) - y_j^R(V_{0^{k-1}}^k, C') + p_i^k - p_i'^k \qquad \text{(ET of } R) \\
&\quad = y_j^R(V_{0^{k-1}}^k, C) - y_j^R(V_{0^{k-1}}^k, C') + p_j^k - p_j'^k \qquad (p_i^k - p_i'^k = p_j^k - p_j'^k) \\
&\quad = \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C) - \hat{y}_j(V_0^1, \dots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

$\hat{y}$ satisfies A.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\pi\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) &= \pi(\hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - r_i^k) \\
&= \pi\hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - \pi r_i^k \\
&= \hat{y}_{\pi_i}(V_0^1, \dots, V_{0^{k-2}}^{k-1}, C) - r_{\pi_i}^k \qquad (r_{\pi_i}^k = \pi r_i^k) \\
&= \hat{y}_i(V_0^1, \dots, V_{0^{k-2}}^{k-1}, \pi C) - r_{\pi_i}^k \qquad \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, \pi C).
\end{aligned}$$

Let $i \in V^k$.

$$\begin{aligned}
\pi\hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, C) &= \pi(y_i^R(V_{0^{k-1}}^k, C) + p_i^k) \\
&= \pi y_i^R(V_{0^{k-1}}^k, C) + \pi p_i^k \\
&= y_{\pi_i}^R(V_{0^{k-1}}^k, C) + p_{\pi_i}^k \qquad (p_{\pi_i}^k = \pi p_i^k) \\
&= y_i^R(V_{0^{k-1}}^k, \pi C) + p_{\pi_i}^k \qquad \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \dots, V_{0^{k-1}}^k, \pi C).
\end{aligned}$$

□

$\hat{y}$ satisfies IIT.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C) - r_i^k \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C') - r_i^k && \text{(I.H.)} \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C') - {r'}_i^k && (\alpha_i^k \overset{*}{=} {\alpha'}_i^k) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C').\end{aligned}$$

* holds by I.H. and IIT of R.

Let $i \in V^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(V_{0^{k-1}}^k,C) + p_i^k \\ &= y_i^R(V_{0^{k-1}}^k,C') + p_i^k && \text{(IIT of } R) \\ &= y_i^R(V_{0^{k-1}}^k,C') + {p'}_i^k && (p_i^k = {p'}_i^k) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C').\end{aligned}$$

□

$\hat{y}$ satisfies PS.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},C) - r_i^k \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\hat{C}) + \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\tilde{C}) \\ &\quad - \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\dot{C}) - r_i^k && \text{(I.H.)} \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\hat{C}) + \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\tilde{C}) \\ &\quad - \hat{y}_i(V_0^1,\ldots,V_{0^{k-2}}^{k-1},\dot{C}) - r_i^{\hat{k}} - r_i^{\tilde{k}} + r_i^{\dot{k}} && (\alpha_i^k \overset{*}{=} \alpha_i^{\hat{k}} + \alpha_i^{\tilde{k}} - \alpha_i^{\dot{k}}) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\hat{C}) + \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\tilde{C}) - \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\dot{C}).\end{aligned}$$

* holds by I.H. and PS of R.

Let $i \in V^k$.

$$\begin{aligned}\hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,C) &= y_i^R(V_{0^{k-1}}^k,C) + p_i^k \\ &= y_i^R(V_{0^{k-1}}^k,\hat{C}) + y_i^R(V_{0^{k-1}}^k,\tilde{C}) \\ &\quad - y_i^R(V_{0^{k-1}}^k,\dot{C}) + p_i^k && \text{(PS of } R) \\ &= y_i^R(V_{0^{k-1}}^k,\hat{C}) + y_i^R(V_{0^{k-1}}^k,\tilde{C}) \\ &\quad - y_i^R(V_{0^{k-1}}^k,\dot{C}) + p_i^{\hat{k}} + p_i^{\tilde{k}} - p_i^{\dot{k}} && (p_i^k = p_i^{\hat{k}} + p_i^{\tilde{k}} - p_i^{\dot{k}}) \\ &= \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\hat{C}) + \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\tilde{C}) \\ &\quad - \hat{y}_i(V_0^1,\ldots,V_{0^{k-1}}^k,\dot{C}).\end{aligned}$$

□

$\hat{y}$ satisfies IIE.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') - r_i^k && \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') - r_i'^k && (\alpha_i^k \stackrel{*}{=} \alpha_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds by I.H. and IIE of R.

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) &= y_i^R(V_{0^{k-1}}^k, C) + p_i^k \\
&= y_i^R(V_{0^{k-1}}^k, C') + p_i^k && \text{(IIE of } R\text{)} \\
&= y_i^R(V_{0^{k-1}}^k, C') + p_i'^k && (p_i^k = p_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

$\hat{y}$ satisfies RA.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C + C') &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C + C') \\
&\quad - r_i^{k+'k} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) + \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') \\
&\quad - r_i^{k+'k} && \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) + \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') \\
&\quad - r_i^k + r_i'^k && (\alpha_i^{k+'k} \stackrel{*}{=} \alpha_i^k + \alpha_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

* holds by I.H. and RA of R.

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C + C') &= y_i^R(V_{0^{k-1}}^k, C + C') + p_i^{k+'k} \\
&= y_i^R(V_{0^{k-1}}^k, C) + y_i^R(V_{0^{k-1}}^k, C') + p_i^{k+'k} && \text{(RA of } R\text{)} \\
&= y_i^R(V_{0^{k-1}}^k, C) + y_i^R(V_{0^{k-1}}^k, C') + p_i^k + p_i'^k && (p_i^{k+'k} = p_i^k + p_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) + \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C').
\end{aligned}$$

□

$\hat{y}$ satisfies CON.

Proof. By CON of R. □

$\hat{y}$ satisfies POL.

Proof. r_i^k depends on α_i^k which is computable in polynomial time because we assume that charge rule R is computable in polynomial time. Taking the minimum, multiplication and substraction can be done in polynomial time. p_i^k depends on μ_i^k and $\tilde{\alpha}_i^k$, both computable in polynomial time since charge rule R is computable in polyomial time. □

$\hat{y}$ satisfies ESCR.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') - r_i'^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - \frac{x}{|\mathbf{V}^{k+1}|} - r_i'^k && \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k - \frac{x}{|\mathbf{V}^{k+1}|} && (\alpha_i^k \overset{*}{=} \alpha_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - \frac{x}{|\mathbf{V}^{k+1}|}.
\end{aligned}$$

* holds by I.H. and ESCR of R.

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= y_i^R(V_{0^{k-1}}^k, C') + p_i'^k \\
&= y_i^R(V_{0^{k-1}}^k, C) - \frac{x}{|\mathbf{V}^{k+1}|} + p_i'^k && \text{(ESCR of } R) \\
&= y_i^R(V_{0^{k-1}}^k, C) + p_i^k - \frac{x}{|\mathbf{V}^{k+1}|} && (p_i^k = p_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - \frac{x}{|\mathbf{V}^{k+1}|}.
\end{aligned}$$

□

$\hat{y}$ satisfies FSCR.

Proof. Let $i \in \mathbf{V}^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C') - r_i'^k \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - x - r_i'^k && \text{(I.H.)} \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - r_i^k - x && (\alpha_i^k \overset{*}{=} \alpha_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - x.
\end{aligned}$$

* holds by I.H. and FSCR of R.

Let $i \in V^k$.

$$\begin{aligned}
\hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C') &= y_i^R(V_{0^{k-1}}^k, C') + p_i'^k \\
&= y_i^R(V_{0^{k-1}}^k, C) - x + p_i'^k && \text{(FSCR of } R\text{)} \\
&= y_i^R(V_{0^{k-1}}^k, C) + p_i^k - x && (p_i^k = p_i'^k) \\
&= \hat{y}_i(V_0^1, \ldots, V_{0^{k-1}}^k, C) - x.
\end{aligned}$$

□

Proposition 5

Proof. Assume $0 < I(G^K) \leq I(G'^K)$, where G'^K is the global tree constructed after agent $i \in V^k$ decided to join in the lasy round K. Let, for $k \leq K$, for all $i \in \mathbf{V}^{k+1}$, $c_{0i} = \gamma \geq |\mathbf{V}^{k+1}| \left(2 \max_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\} - \min_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\}\right) + c_0'$, where $c_0' = c_{0i}'$ and $\gamma > c_0' \geq c_{ij} = c_{ij}'$ for all $i, j \in \mathbf{V}^{k+1}$.

Claim 2. For $k \leq K$, $I(G^k)$ is charged to the agents in V^k, if $k \neq 1$.

Proof of Claim 2.
By PM and CM_0 of R (see Remark 14) and the definition of benefit, if $k \neq 1$,

$$\sum_{i \in V^k} \mu_i^k = b(V^k) = m(V_0^k, C) - m(V_{0^{k-1}}^k, C) \geq \gamma - \max_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\}.$$

We need to show that $\sum_{i \in \mathbf{V}^k} \alpha_i^k \leq \gamma - \max_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\}$.

$$\begin{aligned}
\sum_{i \in \mathbf{V}^k} \alpha_i^k &= \sum_{i \in \mathbf{V}^k} \hat{y}_i(V_0^1, \ldots, V_{0^{k-2}}^{k-1}, C) - \sum_{i \in \mathbf{V}^k} y_i^R(\mathbf{V}_0^{k+1}, C) \\
&= \sum_{j-1}^{k-1} m(V_{0^{j-1}}^j, C) - \sum_{i \in \mathbf{V}^k} y_i^R(\mathbf{V}_0^{k+1}, C) && \text{(BB)} \\
&= \sum_{j-1}^{k-1} m(V_{0^{j-1}}^j, C) - \left(\sum_{i \in \mathbf{V}^k} \left(y_i^R(\mathbf{V}_0^{k+1}, C') + \frac{\gamma - c_0'}{|\mathbf{V}^{k+1}|} \right) \right) && \text{(ESEC)} \\
&\leq \gamma + |\mathbf{V}^k| \max\{c_{ij}\} - |\mathbf{V}^k| \min\{c_{ij}'\} - \frac{|\mathbf{V}^k|(\gamma - c_0')}{|\mathbf{V}^{k+1}|} \\
&= \gamma + |\mathbf{V}^k| \left(\max\{c_{ij}\} - \min\{c_{ij}\} - \frac{(\gamma - c_0')}{|\mathbf{V}^{k+1}|} \right).
\end{aligned}$$

It suffices to show now that

$$|\mathbf{V}^k| \left(\max\{c_{ij}\} - \min\{c_{ij}\} - \frac{(\gamma - c_0')}{|\mathbf{V}^{k+1}|} \right) \leq - \max\{c_{ij}\}.$$

Filling in γ gives the desired inequality, hence $\sum_{i \in \mathbf{V}^k} \alpha_i^k \leq \gamma - \max_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\}$. □

$\sum_{i \in \mathbf{V}^k} \alpha_i^k \leq \gamma - \max_{i,j \in \mathbf{V}^{k+1}}\{c_{ij}\}$ means that all agents $i \in \mathbf{V}^k$ are charged $y_i^R(\mathbf{V}_0^{k+1}, C)$ in round k, i.e., they are charged the amount they would be charged in the optimal situation. The inefficiency of the global tree in round k is thus charged to the agents in V^k. □

Notation

$\mathcal{N} = \{1, 2, \dots\}$ is a set of agents.
$N = \{1, 2, \dots, n\} \subset \mathcal{N}$ is a finite set of agents.
0 denotes the source.
$C = (c_{ij})_{i,j \in N_0}$ is the cost matrix.
C^* is the irreducible cost matrix.
$\overline{C}$ is the cycle-complete cost matrix.
$\mathcal{C}^N$ is the class of all cost matrices over N.
$c(N_0, C, G)$ is the cost of graph G, with vertices in N_0 and cost matrix C.
$\mathcal{G}_0^N$ is the class of graphs with vertices N that are all connected to 0.
$m(N_0, C)$ is the minimum cost of the network.
$v_C(S) = m(S_0, C)$.
$V^k \subseteq N$ is the set of agents joining in round k.
K denotes the total number of rounds.
$\mathbf{V}^k$ is the set of agents which joined in round $1, \dots, k-1$.
0^{k-1} denotes the source of round k.
G^k is the global tree constructed in round k.
y is the charge rule for the classical mcst problem.
y^R is the charge rule for the classical mcst problem, given a charge rule R.
$\hat{y}$ is the charge rule for the iterative mcst problem.
$I(G^k)$ is the inefficiency of the global tree G^k.
$I(G^k_{k-1})$ is the inefficiency of the tree constructed in round k compared to the tree constructed in round $k-1$.
$d(i)$ is the number of agents that connects through i.
q_i^k is the extra payment or reimbursement according to charge rules (a)-(d).
$b(V^k)$ is the benefit of the agents in V^k as a group.
$\tilde{\alpha}_i^k$ is the lower bound on i's extra cost share, for $i \in V^k$.
μ_i^k is the upper bound on i's extra cost share, for $i \in V^k$.
p_i^k is the extra cost share, for $i \in V^k$.
P is the total amount that $i \in \mathbf{V}^k$ receive and that $i \in V^k$ pay extra.
p^* and γ_i^k are values that agents are charged extra on top of $\tilde{\alpha}_i^k$.
α_i^k is the upper bound on i's reimbursement, $i \in \mathbf{V}^k$.
r_i^k is the reimbursement, for $i \in \mathbf{V}^k$.
ρ is the proportion of α_i^k that agents will receive, for $i \in \mathbf{V}^k$.
r^* and β_i^k are values of reimbursements, for $i \in \mathbf{V}^k$.

Bibliography

H. Aarts. *Minimum Cost Spanning Tree Games and Set Games*. PhD thesis, Univeristy of Twente, 1994.

S. Airiau. Cooperative games. Lecture notes, `http://www.lamsade.dauphine.fr/~airiau/Teaching/CoopGames/2012/`, 2012.

K. Ando. Computation of the Shapley value of minimum cost spanning tree games: #P-hardness and polynomial cases. *Japan Journal of Industrial and Applied Mathematics*, 29(3):385–400, 2012.

E. Angel, E. Bampis, L. Blin, and L. Gourvès. Fair cost-sharing methods for the minimum spanning tree game. *Information Processing Letters*, 100(1): 29–35, 2006.

G. Bergantiños and M. Gómez-Rúa. Minimum cost spanning tree problems with groups. *Economic Theory*, 43(2):227–262, 2010.

G. Bergantiños and L. Lorenzo. A non-cooperative approach to the cost spanning tree problem. *Mathematical Methods of Operations Research*, 59 (3):393–403, 2004.

G. Bergantiños and L. Lorenzo. Non-cooperative cost spanning tree games with budget restrictions. *Naval Research Logistics (NRL)*, 55(8):747–757, 2008a.

G. Bergantiños and S. Lorenzo. Optimistic weighted Shapley rules in minimum cost spanning tree problems. *European Journal of Operational Research*, 185(1):289–298, 2008b.

G. Bergantiños and J. Vidal-Puga. A fair rule in minimum cost spanning tree problems. *Journal of Economic Theory*, 137(1):326–352, 2007a.

G. Bergantiños and J. Vidal-Puga. The optimistic TU game in minimum cost spanning tree problems. *International Journal of Game Theory*, 36 (2):223–239, 2007b.

G. Bergantiños and J. Vidal-Puga. On some properties of cost allocation rules in minimum cost spanning tree problems. *Czech Economic Review*, 2 (3):251–267, 2008.

G. Bergantiños and J. Vidal-Puga. Additivity in minimum cost spanning tree problems. *Journal of Mathematical Economics*, 45(1):38–42, 2009.

G. Bergantiños and J. Vidal-Puga. Realizing fair outcomes in minimum cost spanning tree problems through non-cooperative mechanisms. *European Journal of Operational Research*, 201(3):811–820, 2010.

G. Bergantiños and J. Vidal-Puga. The folk solution and Boruvka's algorithm in minimum cost spanning tree problems. *Discrete Applied Mathematics*, 159(12):1279–1283, 2011.

G. Bergantiños, M. Gómez-Rúa, N. Llorca, M. Pulido, and J. Sánchez-Soriano. A cost allocation rule for k-hop minimum cost spanning tree problems. *Operations Research Letters*, 40(1):52–55, 2012.

C. G. Bird. On cost allocation for a spanning tree: A game theoretic approach. *Networks*, 6(4):335–350, 1976.

A. Bogomolnaia and H. Moulin. Sharing a minimal cost spanning tree: Beyond the folk solution. *Games and Economic Behavior*, 69(2):238–248, 2010.

A. Bogomolnaia, R. Holzman, and H. Moulin. Sharing the cost of a capacity network. *Mathematics of Operations Research*, 35(1):173–192, 2010.

A. Bondy and U. Murty. *Graph Theory*. Graduate Texts in Mathematics. Springer, 2008.

O. Boruvka. O Jistém Problému Minimálním (About a Certain Minimal Problem) (in Czech, German summary). *Práce Mor. Prírodoved. Spol. v Brne III*, 3, 1926.

R. Brânzei, S. Moretti, H. Norde, and S. Tijs. The P-value for cost sharing in minimum cost spanning tree situations. Technical report, Tilburg University, 2004.

J. Brenner and G. Schäfer. Online cooperative cost sharing. In *Proceedings of the 7th international conference on Algorithms and Complexity*, CIAC'10, pages 252–263. Springer-Verlag, 2010.

R. Campos and M. Ricardo. A fast algorithm for computing minimum routing cost spanning trees. *Computer Networks*, 52(17):3229–3247, 2008.

B. Çiftçi and S. Tijs. A vertex oriented approach to minimum cost spanning tree problems. Discussion Paper 2007-89, Tilburg University, Center for Economic Research, 2007.

A. Claus and D. J. Kleitman. Cost allocation for a spanning tree. *Networks*, 3(4):289–304, 1973.

T. Driessen. *Cooperative Games, Solutions and Applications*. Theory and decision library: Series C, Game theory, mathematical programming, and operations research. Springer, 1988.

B. Dutta and A. Kar. Cost monotonicity, consistency and minimum cost spanning tree games. *Games and Economic Behavior*, 48(2):223–248, 2004.

D. Eppstein, Z. Galil, and G. F. Italiano. Dynamic graph algorithms. Technical Report TR-056-96, ALCOM-IT, 1996.

V. Feltkamp, S. Tijs, and S. Muto. On the irreducible core and the equal remaining obligations rule of minimum cost spanning extension problems. Discussion Paper 1994-106, Tilburg University, Center for Economic Research, 1994.

F. Fernández, M. Hinojosa, A. Mármol, and J. Puerto. Opportune moment strategies for a cost spanning tree game. *Mathematical Methods of Operations Research*, 70(3):451–463, 2009.

F. R. Fernández, M. A. Hinojosa, and J. Puerto. Multi-criteria minimum cost spanning tree games. *European Journal of Operational Research*, 158 (2):399–408, 2004.

M. L. Fredman and R. E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM*, 34(3):596–615, 1987.

R. L. Graham and P. Hell. On the history of the minimum spanning tree problem. *IEEE Ann. Hist. Comput.*, 7(1):43–57, 1985.

D. Granot and G. Huberman. Minimum cost spanning tree games. *Mathematical Programming*, 21(1):1–18, 1981.

A. Gupta, J. Könemann, S. Leonardi, R. Ravi, and G. Schäfer. An efficient cost-sharing mechanism for the prize-collecting Steiner forest problem. In *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, SODA '07, pages 1153–1162. Society for Industrial and Applied Mathematics, 2007.

A. Kar. Axiomatization of the Shapley value on minimum cost spanning tree games. *Games and Economic Behavior*, 38(2):265–277, 2002.

J. Kruskal, J. B. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical Society*, 7(1):48–50, 1956.

S. Leonardi and G. Schäfer. Cross-monotonic cost sharing methods for connected facility location games. *Theoretical Computer Science*, 326(1): 431–442, 2004.

C. Martel. The expected complexity of Prim's minimum spanning tree algorithm. *Information Processing Letters*, 81(4):197–201, 2002.

S. Moretti, H. Norde, K. Pham Do, and S. Tijs. Connection problems in mountains and monotonic allocation schemes. Discussion Paper 2001-12, Tilburg University, Center for Economic Research, 2001.

S. Moretti, S. Tijs, R. Brânzei, and H. Norde. Cost monotonic 'Cost and Charge' rules for connection situations. Discussion Paper 2005-104, Tilburg University, Center for Economic Research, 2005.

H. Moulin. Pricing traffic in a spanning network. In *Proceedings of the 10th ACM conference on Electronic commerce*, EC '09, pages 21–30, 2009.

H. Moulin. Cost sharing in networks: some open questions. *International Game Theory Review*, 15(02):1–10, 2013.

H. Moulin and F. Laigret. Equal-need sharing of a network under connectivity constraints. *Games and Economic Behavior*, 72(1):314–320, 2011.

H. Norde, S. Moretti, and S. Tijs. Minimum cost spanning tree games and population monotonic allocation schemes. Open access publications from Tilburg University, Tilburg University, 2004.

R. C. Prim. Shortest connection networks and some generalizations. *Bell System Technology Journal*, 36:1389–1401, 1957.

A. Schrijver. Matchings and coverings. Lecture notes, `http://homepages.cwi.nl/~lex/files/agt1.pdf`, 2012.

L. Shapley. *A value for n-person game*, pages 305–317. 1953a.

L. S. Shapley. Stochastic games. *Proceedings of the National Academy of Sciences of the United States of America*, 39(10):1095–1100, 1953b.

W. W. Sharkey. Chapter 9 network models in economics. In C. M. M.O. Ball, T.L. Magnanti and G. Nemhauser, editors, *Network Routing*, volume 8 of *Handbooks in Operations Research and Management Science*, pages 713–765. Elsevier, 1995.

S. Tazari. Cross-monotonic cost-sharing schemes for combinatorial optimization games: A survey. *University of British Columbia*, 2005.

S. Tijs, R. Brânzei, S. Moretti, and H. Norde. Obligation rules for minimum cost spanning tree situations and their monotonicity properties. *European Journal of Operational Research*, 175(1):121–134, 2006.

C. Trudeau. Cost sharing with multiple technologies. *Games and Economic Behavior*, 67(2):695–707, 2009.

C. Trudeau. A new stable and more responsive cost sharing solution for minimum cost spanning tree problems. *Games and Economic Behavior*, 75 (1):402–412, 2012.

C. Trudeau. Linking the Kar and folk solutions through a problem separation property. Working Papers 1301, University of Windsor, Department of Economics, 2013a.

C. Trudeau. Characterizations of the Kar and folk solutions for minimum cost spanning tree problems. *International Game Theory Review (IGTR)*, 15(02):1–16, 2013b.

C. Trudeau. Characterizations of the cycle-complete and folk solutions for minimum cost spanning tree problems. *Social Choice and Welfare*, pages 1–17, 2013c.

C. Trudeau. Minimum cost spanning tree problems with indifferent agents. *Under review*, pages 1–16, 2013d.

S. van Zwam. *Cost Sharing Mechanisms for Network Games*. Department of Mathematics and Computer Science, Eindhoven University of Technology, 2005.

D. B. West. *Introduction to Graph Theory (2nd Edition)*. Prentice Hall, 2000.