

Efficient cost-sharing mechanisms for prize-collecting problems

A. Gupta · J. Könemann · S. Leonardi ·
R. Ravi · G. Schäfer

Received: 9 August 2011 / Accepted: 8 April 2014 / Published online: 23 April 2014
© Springer-Verlag Berlin Heidelberg and Mathematical Optimization Society 2014

Abstract We consider the problem of designing efficient mechanisms to share the cost of providing some service to a set of self-interested customers. In this paper, we mainly focus on cost functions that are induced by prize-collecting optimization problems. Such cost functions arise naturally whenever customers can be served in two different ways: either by being part of a common service solution or by being served individually. One of our main contributions is a general *lifting technique* that allows us to extend the social cost approximation guarantee of a Moulin mechanism for the respective non-prize-collecting problem to its prize-collecting counterpart. Our lifting technique also

A preliminary version of this paper appeared under the title “An efficient cost-sharing mechanism for the prize-collecting Steiner forest problem” in the Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms, pp. 1153–1162, ACM Press, 2007.

J. Könemann received the Research support by NSERC grant no. 288340-2004 and by an IBM Faculty Award.

Part of this work was done while S. Leonardi was visiting the School of Computer Science at Carnegie Mellon University.

Part of this work was done while G. Schäfer was visiting the School of Computer Science at Carnegie Mellon University.

A. Gupta
School of Computer Science, Carnegie Mellon University, Pittsburgh, PA15213, USA
e-mail: anupamg@andrew.cmu.edu

J. Könemann
Department of Combinatorics and Optimization, University of Waterloo,
Waterloo, ONN2L 3G1, Canada
e-mail: jochen@math.uwaterloo.ca

S. Leonardi
Department of Computer, Control and Management Engineering ‘Antonio Ruberti’,
Sapienza University of Rome, Via Ariosto 25, 00185 Rome, Italy
e-mail: leon@dis.uniroma1.it

suggests a generic design template to derive Moulin mechanisms for prize-collecting problems. The approach is particularly suited for cost-sharing methods that are based on primal-dual algorithms. We illustrate the applicability of our approach by deriving Moulin mechanisms for prize-collecting variants of submodular cost-sharing, facility location and Steiner forest problems. All our mechanisms are essentially best possible with respect to budget balance and social cost approximation guarantees. Finally, we show that the Moulin mechanism by Könemann et al. (SIAM J Comput 37(5):1319–1341, 2008) for the Steiner forest problem is $O(\log^3 k)$ -approximate. Our approach adds a novel methodological contribution to existing techniques by showing that such a result can be proved by embedding the graph distances into random hierarchically separated trees.

Mathematics Subject Classification 90C27 · 68R99 · 91A10

1 Introduction

1.1 Motivation

We consider the problem of sharing the cost of providing some service to a set of self-interested customers (or users). At an abstract level, such problems can be described as follows: We are given a set U of k players and a cost function $C : 2^U \rightarrow \mathbb{R}$ which specifies a cost $C(S)$ to establish the service for player set $S \subseteq U$. Every player $i \in U$ derives a privately held *valuation* v_i from being served. We are interested in finding a *cost-sharing mechanism* M that first solicits bids $(b_i)_{i \in U}$ from all players and then determines a set $S^M \subseteq U$ of players to service and a payment $p_i \leq b_i$ for every player $i \in S$. The *utility* u_i of player i is defined as $v_i - p_i$ if $i \in S^M$ and is 0 otherwise. We assume that each player $i \in U$ wants to maximize his utility. A player might therefore misreport his actual valuation if this increases his utility.

In such settings, the servicing cost $C(S)$ of a subset of players $S \subseteq U$ often naturally relates to the cost function of an underlying optimizing problem. For example, the cost of connecting a set of users residing at terminal nodes in a network might be given as the cost of a Steiner tree on these terminals. The cost of opening warehouses at selected locations to minimize the transportation costs to all customer sites can be

R. Ravi

Tepper School of Business, Carnegie Mellon University, Pittsburgh, PA15213, USA
e-mail: ravi@cmu.edu

G. Schäfer (✉)

Networks and Optimization Group, Centrum Wiskunde & Informatica (CWI),
Science Park 123, 1098 XG Amsterdam, The Netherlands
e-mail: g.schaefer@cw.nl

G. Schäfer

Department of Econometrics and Operations Research, VU University Amsterdam, De Boelelaan 1105,
1081 HV Amsterdam, The Netherlands

defined as the optimal solution cost of a facility location problem. Several classical optimization problems have been studied in a cost-sharing context in recent years.

In this paper, we focus on a natural extension of the above cost-sharing viewpoint: We assume that every player $i \in U$ can be served in two different ways: either player i is part of a common service solution which is shared among all participants, or i is served individually. For example, in the warehouse application mentioned above one may want to consider the option of building an on-site storage facility for the customer as an alternative to outsourcing his goods to one of the available warehouses. Such extensions give rise to cost functions that are captured by so-called *prize-collecting optimization problems* (or *optimization problems with penalties*).

An example of such a problem is the *prize-collecting Steiner forest problem (PCSF)*: We are given an undirected graph $G = (V, E)$ with edge costs $c : E \rightarrow \mathbb{R}^+$, a set of k terminal pairs $R = \{(s_i, t_i)\}_{i \in [k]}$, and penalties $\pi : R \rightarrow \mathbb{R}^+$. A feasible solution (F, Q) consists of a forest $F \subseteq E$ and a subset $Q \subseteq R$ of terminal pairs such that for all $(s_i, t_i) \in R$ either s_i, t_i are connected by F or $(s_i, t_i) \in Q$. The objective is to compute a feasible solution of minimum cost $c(F) + \pi(Q)$.

In the cost-sharing context, we associate each terminal pair $(s_i, t_i) \in R$ with a player $i \in U$ whose goal is to connect s_i and t_i . The service provider can choose to either connect the terminals s_i and t_i through a common network F , or to connect s_i and t_i individually at a cost of $\pi(i)$. That is, the servicing cost $C(S)$ for player set $S \subseteq U$ is defined as the cost of an optimal solution of the prize-collecting Steiner forest instance with terminal pairs $\{(s_i, t_i)\}_{i \in S}$.

1.2 Cost-sharing mechanisms

When it comes to the design of cost-sharing mechanisms, there are several desirable properties that one would like to achieve. A cost-sharing mechanism M is called *strategyproof* if bidding truthfully (i.e., announcing $b_i = v_i$) is a dominant strategy for every player $i \in U$. If this is true even if players are permitted to collude (formal definitions will be given in Sect. 2), then we call a mechanism *group-strategyproof*. M is *budget balanced* if the total cost $C(S)$ of servicing the players in S is equal to the sum of the payments charged to the players in S . Further, M is *efficient* if it selects a set S^M of players that maximizes $v(S) - C(S)$ among all subsets $S \subseteq U$.

Classical results in economics [17, 34] state that budget balance and efficiency cannot be achieved simultaneously by any mechanism. Moreover, Feigenbaum et al. [15] showed that there is no strategyproof mechanism that always recovers a constant fraction of the optimal efficiency and a constant fraction of the incurred cost, even for the simple fixed-tree multicast problem.

In light of these hardness results, most of the previous work on mechanism design concentrated on proper subsets of the above design goals. One notable class of such mechanisms are so-called *Moulin mechanisms* [29, 30]. A Moulin mechanism $M(\xi)$ is based on a *cost-sharing method* ξ , which defines a cost share $\xi_i(S)$ for every player $i \in S$ and every $S \subseteq U$. The mechanism starts with the entire player set $S = U$. In each iteration, it proposes a cost share $\xi_i(S)$ to every remaining player $i \in S$. If all players accept their cost shares, it outputs $S^M = S$ as the set of served players together with

the payments $(\xi_i)_{i \in S}$ and stops. Otherwise, it removes all players that are not willing to pay their cost shares and continues with the next iteration. Moulin [29] showed that the mechanism $M(\xi)$ is group-strategyproof if the underlying cost-sharing method ξ is *cross-monotonic*, i.e., the cost share of each player increases as other players leave the game; formally, for every $S \subseteq T \subseteq U$, $\xi_i(S) \geq \xi_i(T)$ for every $i \in S$.

Moulin's framework has been applied to several classical optimization problems such as submodular cost-sharing [30], fixed-tree multicast [3, 14, 15], minimum spanning tree [22, 24], Steiner tree [22], Steiner forest [26], set cover [20], facility location [32], connected facility location [18, 27, 32], and machine scheduling [8, 10]. Lower bounds on the budget balance factor that is achievable by a cross-monotonic cost-sharing mechanism were studied in [20, 25, 26].

Roughgarden and Sundararajan [36] introduced an alternative measure of efficiency that circumvents the intractability results in [15, 17, 34] at least partially. The authors define the *social cost* $\Pi(S)$ of a set $S \subseteq U$ as $\Pi(S) = v(U \setminus S) + C(S)$. A mechanism M is said to be α -*approximate* if the set S^M it outputs has social cost at most $\alpha \min_{S \subseteq U} \Pi(S)$. The intuition behind this definition loosely comes from the fact that $v(U) - \Pi(S) = v(S) - C(S)$, which is the traditional definition of efficiency. Since $v(U)$ is a constant, a set S has minimum social cost if and only if it has maximum efficiency.

Roughgarden and Sundararajan [36] then developed a framework to quantify the extent to which a Moulin mechanism minimizes the social cost. They applied this framework to show that the Shapley mechanism is $O(\log k)$ -approximate for submodular cost functions and that the cost sharing method of Jain and Vazirani [22] for the Steiner tree problem gives a mechanism that is $O(\log^2 k)$ -approximate. In a subsequent paper, Chawla et al. [11] applied the framework to show that the Moulin mechanism of Könemann et al. [26] for the Steiner forest problem is $O(\log^2 k)$ -approximate. In a following paper, Roughgarden and Sundararajan [35] proved polylogarithmic upper and lower bounds for facility location, Steiner tree and rent-or-buy network design problems.

1.3 Our results and techniques

One of our main contributions is a general *lifting technique* that allows us to prove social cost approximation guarantees of Moulin mechanisms for prize-collecting problems. Basically, our approach enables us to lift the approximation guarantee of a Moulin mechanism $M(\xi)$ for an optimization problem to a Moulin mechanism $M(\tilde{\xi})$ for the prize-collecting variant of the problem. The approach naturally applies to cost-sharing methods that are based on primal-dual algorithms, where cost shares relate to dual variables of a linear programming relaxation of the problem and are grown as a process over time.

Our lifting technique also suggests a generic design template to derive Moulin mechanisms for prize-collecting problems. Here the high-level idea is to adapt the cross-monotonic cost-sharing method $\tilde{\xi}$ for the non-prize-collecting problem simply by “freezing” the cost share of a player i when it reaches his penalty $\pi(i)$. By exploiting our lifting technique, all one is left to do then is to prove cross-monotonicity and (approximate) budget balance of the resulting cost-sharing method.

We illustrate the applicability of our approach by deriving Moulin mechanisms for prize-collecting variants of submodular cost-sharing, facility location and Steiner forest problems. Our mechanisms are essentially best possible with respect to both budget balance and social cost approximation guarantees (for the latter a constant gap remains). An overview of the specific approximation guarantees obtained in this paper and the known lower bounds are given in Table 1.

Technically, the most challenging part lies in obtaining a 2-budget balanced Moulin mechanism that is $O(\log^2 k)$ -approximate for the Steiner forest problem. Our algorithm is a natural extension of the primal-dual algorithm of Agrawal et al. [1] for prize-collecting Steiner trees and the cross-monotonic cost-sharing method for Steiner forests presented by Könemann et al. [26]. Despite its simplicity, our algorithm achieves the same approximation guarantee as the combinatorial algorithm by Hajiaghayi and Jain [19].

Finally, we show that the Moulin mechanism by Könemann et al. [26] for the Steiner forest problem is $O(\log^3 k)$ -approximate. This is achieved by adding a novel methodological contribution to the framework proposed by Roughgarden and Sundararajan in [36]: We show that such a result can also be proved by embedding the graph distances into random *hierarchically separated trees (HST)* [5, 13]. Independently, Chawla et al. [11] showed (using a more involved analysis) that the Moulin mechanism derived in [26] is $O(\log^2 k)$ -approximate.

1.4 Related work

Moulin [29] showed that cross-monotonicity is a sufficient condition to obtain group-strategyproof cost-sharing mechanisms. He also shows that it is a necessary condition for submodular cost functions. Immorlica et al. [20] gave a partial characterization of group-strategyproof cost sharing mechanisms. Pountourakis and Vidali [33] recently gave a complete characterization of group-strategyproof cost-sharing mechanisms and thereby settled a major open problem. They showed that group-strategyproof cost-sharing mechanisms are completely characterized by a certain *fence monotonicity* property in combination with a stable allocation rule and a valid tie-breaking rule (see [33] for definitions). It remains open whether this characterization result can be used algorithmically in general.

Moulin and Shenker [30] showed that the Shapley value cost sharing mechanism minimizes the overall welfare loss among all group-strategyproof cost sharing mechanisms for submodular cost functions. Dutta and Ray [12] and Jain and Vazirani [23]

Table 1 Summary of results obtained for prize-collecting problems in this paper

Problem	Non-prize-collecting	Prize-collecting	Lower bounds
Submodular cost	$(1, 2H_k)$ [23]	$(1, 2H_k + 1)$	$(1, H_k)$ [36]
Facility location	$(3, H_k + 3)$ [32]	$(3, 3(H_k + 4))$	$(3, H_k/3)$ [20, 36]
Steiner forest	$(2, O(\log^2 k))$ [26]	$(3, O(\log^2 k))$	$(2, \Omega(\log^2 k))$ [26, 36]

Each pair (β, α) states the budget balance (β) and social cost (α) approximation guarantees for the respective problem

gave a cross-monotonic and budget balanced cost-sharing method for the submodular cost-sharing problem based on a primal-dual algorithm.

Pál and Tardos [32] gave a cross-monotonic and 3-budget balanced cost-sharing method for the metric facility location problem. Their cost-sharing method is based on a primal-dual interpretation of the approximation algorithm by Mettu and Plaxton [28].

The problem of computing prize-collecting Steiner trees or forests is APX-complete [4, 6], and hence neither of the two problems admits a PTAS unless $P = NP$. The first constant-factor approximation for the prize-collecting Steiner tree problem is a 3-approximation algorithm by Bienstock et al. [7] based on LP-rounding. Goemans and Williamson [16] obtain an improved $(2 - \frac{1}{k})$ -approximation algorithm using the primal-dual schema. The current best approximation algorithm for the problem is a 1.9672-approximation algorithm due to Archer et al. [2].

One can easily modify the algorithm of Bienstock et al. [7] to give a 3-approximation for the prize-collecting Steiner forest problem as well. Hajiaghayi and Jain [19] refine the LP-rounding idea of Bienstock et al. and obtain an LP-based 2.54-approximation algorithm for the problem. The authors also present a primal-dual combinatorial 3-approximation algorithm for the problem. This algorithm substantially deviates from the classical framework of Goemans and Williamson, requiring crucial use of Farkas' Lemma, wherein the dual variables are both increased and decreased throughout the execution of the algorithm.

1.5 Organization of the paper

We introduce some additional notation in Sect. 2. Our lifting technique is the subject of Sect. 3. Subsequently, we use this technique to derive Moulin mechanisms for the prize-collecting variants of submodular cost-sharing, metric facility location and Steiner forest problems in Sects. 4, 5 and 6, respectively. Finally, we prove the social cost approximation of the Moulin mechanism in [26] for the Steiner forest problem in Sect. 7.

2 Preliminaries

We use the following notation throughout the paper. Let U be the set of players. The number of players in U is denoted by $k = |U|$. Further, let $C : 2^U \rightarrow \mathbb{R}^+$ be a non-negative cost function on U that assigns to each subset $S \subseteq U$ a cost $C(S)$. The interpretation is that $C(S)$ refers to the cost of optimally serving the players in S . In particular, if the cost function C is defined implicitly by some underlying optimization problem, then $C(S)$ refers to the cost of an optimal solution to the problem with player set S . We assume that C is non-decreasing, i.e., for every $S \subseteq T$, $C(S) \leq C(T)$, and $C(\emptyset) = 0$. For every $n \in \mathbb{N}$, we use $[n]$ to refer to the set $\{1, \dots, n\}$.

Every player $i \in U$ specifies a non-negative bid b_i that he is willing to pay for the service. A *cost sharing mechanism* M takes the bids $(b_i)_{i \in U}$ of all players as input and computes a set $S^M \subseteq U$ of players that receive service and a non-negative payment p_i for every player $i \in S^M$; we implicitly assume that $p_i = 0$ for every $i \notin S^M$.

We require that a cost sharing mechanism satisfies the following three conditions:

- *Individual Rationality*: A player is charged only if he receives service and his payment is at most his bid, i.e., $p_i = 0$ if $i \notin S^M$ and $p_i \leq b_i$ if $i \in S^M$.

- *No Positive Transfers*: A player is not paid for receiving the service, i.e., $p_i \geq 0$ for all $i \in S^M$.
- *Consumer Sovereignty*: A player is guaranteed to receive service if he is willing to bid high enough, i.e., there exists a threshold value b_i^* for every player $i \in U$ such that $i \in S^M$ for all $b_i \geq b_i^*$.

Every player has a private non-negative *valuation* v_i for receiving the service. The *utility* u_i of player i is defined as $u_i = v_i - p_i$ if $i \in S^M$ and $u_i = 0$ otherwise. We assume that every player’s goal is to maximize his utility u_i . We say that M is *strategyproof* if bidding truthfully is a dominant strategy for every player. We assume that players can form coalitions in order to coordinate their bids. A mechanism M is *group-strategyproof* if no coordinated bidding of a coalition $T \subseteq U$ can ever strictly increase the utility of some player in T without strictly decreasing the utility of another player in T .

A cost sharing mechanism M is β -*budget balanced* if the output set S^M and payments $(p_i)_{i \in U}$ satisfy

$$\frac{1}{\beta} C(S^M) \leq \sum_{i \in S^M} p_i \leq C(S^M) \tag{1}$$

The requirement expressed by the first inequality is also called β -*cost recovery* and the second one is called *competitiveness*.

Remark 1 We use the β -budget balance definition in (1) throughout the paper. However, it is not hard to verify that all our mechanisms satisfy the stronger β -cost recovery condition that at least $1/\beta$ of the cost of the (not necessarily optimal) solution computed by the mechanism is recovered.

2.1 Moulin mechanisms

A *cost-sharing method* ξ is an algorithm that, given any subset $S \subseteq U$ of players, computes a solution to service S and for each $i \in S$ determines a non-negative cost share $\xi_i(S)$. In subsequent sections, we sometimes define ξ without explicit reference to the given player set S and instead define the algorithm for an arbitrary player set U ; no confusion should arise.

We say that ξ is β -*budget balanced* if for every $S \subseteq U$

$$\frac{1}{\beta} \cdot C(S) \leq \sum_{i \in S} \xi_i(S) \leq C(S).$$

We call ξ simply *budget balanced* if it is 1-budget balanced. A cost-sharing method ξ is *cross-monotonic* if for any two sets S and T such that $S \subseteq T$ and any player $i \in S$ we have $\xi_i(S) \geq \xi_i(T)$.

Moulin [29] showed that, given a β -budget balanced and cross-monotonic cost-sharing method ξ for the underlying problem, the following cost-sharing mechanism $M(\xi)$ is β -budget-balance and group-strategyproof:

```

1 Initialize  $S \leftarrow U$  ;
2 while  $S \neq \emptyset$  and  $\exists i \in S$  with  $\xi_i(S) > b_i$  do
3   | Choose a player  $i \in S$  with  $\xi_i(S) > b_i$  ;
4   |  $S \leftarrow S \setminus \{i\}$  ;
5 end
6 Let  $S^M \leftarrow S$  be the set of served players ;
7 Define  $p_i \leftarrow \xi_i(S)$  as the payment of player  $i \in S^M$  ;
8 return  $(S^M, (p_i))$ 
    
```

Theorem 1 (Moulin Mechanisms [29]) *Given a cross-monotonic and β -budget balanced cost sharing method, the Moulin mechanism $M(\xi)$ satisfies individual rationality, no positive transfers and consumer sovereignty, and is group-strategyproof and β -budget balanced.*

The following fact summarizes an important property of Moulin mechanisms (see also [29,31]).

Fact 2 (Order Invariance Property) *Given a cross-monotonic cost-sharing method ξ , the final set S^M of players output by the Moulin mechanism $M(\xi)$ is independent of the order of eviction.*

2.2 Social cost approximation

Roughgarden and Sundararajan [36] introduced an alternative notion of efficiency for cost-sharing mechanisms: For a set $S \subseteq U$, define $v(S) = \sum_{i \in S} v_i$. Define the *social cost* $\Pi(S)$ of a set $S \subseteq U$ as

$$\Pi(S) = v(U \setminus S) + C(S).$$

Note that the social cost of a set $S \subseteq U$ is evaluated with respect to the optimal servicing cost $C(S)$. A cost-sharing mechanism M is said to be α -approximate if, assuming that every player $i \in U$ bids truthfully $b_i = v_i$, the final set S^M output by M satisfies

$$\Pi(S^M) \leq \alpha \cdot \Pi(S) \quad \forall S \subseteq U. \tag{2}$$

Roughgarden and Sundararajan [36] showed that the approximation guarantee of a Moulin mechanism $M(\xi)$ is intimately related to the *summability* of the cost sharing method ξ , which is defined as follows: Assume we are given an arbitrary permutation σ on the players in U and a subset $S \subseteq U$ of players. We assume that the players in S are ordered according to σ , i.e., $S = \{i_1, \dots, i_{|S|}\}$ where $i_j \prec_\sigma i_k$ if and only if $1 \leq j < k \leq |S|$. We define $S_j \subseteq S$ as the (ordered) set of the first j players of S according to the order σ . A cost-sharing method ξ is α -summable if for every ordering σ and every subset $S \subseteq U$

$$\sum_{j=1}^{|S|} \xi_{i_j}(S_j) \leq \alpha \cdot C(S). \tag{3}$$

Theorem 3 (Social Cost Approximation [36]) *Let ξ be a cross-monotonic and β -budget balanced cost sharing method that is α -summable. Then the Moulin mechanism $M(\xi)$ is $(\alpha + \beta - 1)$ -approximate with respect to social cost.*

Roughgarden and Sundararajan [36] actually showed that the summability of ξ relates to several efficiency measures for $M(\xi)$ such as additive welfare loss, social cost, social reward and social welfare, with social cost being just one of them. However, here we focus on the social cost approximation guarantee as defined in (2) and refer to [36] for more details.

3 Lifting technique

We present a general lifting technique to prove bounds on the social cost approximation guarantee of Moulin mechanisms for prize-collecting problems.

Suppose we are given a set of players U and a non-negative cost function $\bar{C} : 2^U \rightarrow \mathbb{R}^+$, where $\bar{C}(S)$ is the cost of serving the players in $S \subseteq U$. The cost function \bar{C} is defined implicitly by an underlying optimization problem $\bar{\mathcal{P}}$, i.e., $\bar{C}(S)$ is defined as the cost of an optimal solution of $\bar{\mathcal{P}}$ for player set $S \subseteq U$. We refer to $\bar{\mathcal{P}}$ as the *non-prize collecting problem*.

We generalize $\bar{\mathcal{P}}$ to its prize-collecting variant \mathcal{P} as follows. Suppose every player $i \in U$ has a non-negative *penalty* $\pi(i)$. Given a subset $S \subseteq U$, there are two possibilities to serve each player $i \in S$: either i becomes part of a group $T \subseteq S$ whose members share the common cost $\bar{C}(T)$ or i is served individually at a cost of $\pi(i)$. Put differently, the cost function C of the corresponding *prize-collecting problem* \mathcal{P} is defined as

$$C(S) = \min_{T \subseteq S} \left(\bar{C}(T) + \sum_{i \in S \setminus T} \pi(i) \right).$$

Assume that we have a Moulin mechanism $M(\bar{\xi})$ for the non-prize-collecting problem $\bar{\mathcal{P}}$, driven by a cross-monotonic cost sharing method $\bar{\xi}$. Then, basically, our approach allows us to lift the approximation guarantee of $M(\bar{\xi})$ for the non-prize-collecting problem $\bar{\mathcal{P}}$ to a Moulin mechanism $M(\xi)$ for the prize-collecting problem \mathcal{P} . Our approach applies naturally to primal-dual cost-sharing methods, where cost shares relate to dual values that are grown over time. Subsequently, we use $\bar{\xi}^\tau$ and ξ^τ to refer to the cost shares at time τ for problems $\bar{\mathcal{P}}$ and \mathcal{P} , respectively.

The main result of this section is summarized in the following theorem.

Theorem 4 (Lifting Theorem) *Let ξ be a cross-monotonic and β -budget balanced cost-sharing method for a prize-collecting problem \mathcal{P} . Further, let $\bar{\xi}$ be a cross-monotonic cost-sharing method such that $M(\bar{\xi})$ is α -approximate for the corresponding non-prize-collecting problem $\bar{\mathcal{P}}$. Then the Moulin mechanism $M(\xi)$ is $(1 + \alpha)\beta$ -approximate for \mathcal{P} if ξ satisfies the following two properties for every $S \subseteq U$:*

1. For every player $i \in S$, $\xi_i(S) \leq \pi(i)$.
2. Let τ_0 be the first point of time τ at which $\xi_i^\tau(S) = \pi(i)$ for some player $i \in S$; let $\tau_0 = \infty$ if no such time exists. Then for every player $j \in S$

$$\xi_j^\tau(S) = \bar{\xi}_j^\tau(S) \quad \forall \tau \in [0, \tau_0].$$

Our Lifting Theorem gives rise to the following general design template to derive a Moulin mechanism $M(\xi)$ for a prize-collecting problem \mathcal{P} :

1. Let $M(\bar{\xi})$ be an α -approximate Moulin mechanism for the non-prize collecting problem $\bar{\mathcal{P}}$, driven by a cross-monotonic cost sharing method $\bar{\xi}$ that grows cost shares over time.
2. Derive a cost-sharing method ξ for \mathcal{P} by adapting $\bar{\xi}$ such that Properties 1 and 2 of the Lifting Theorem are satisfied.
 A natural way to accomplish Property 1 is to simply “freeze” the cost share of a player i as soon as he reaches his penalty $\pi(i)$. Typically, this also ensures that Property 2 holds because both cost-sharing methods $\bar{\xi}$ and ξ progress in exactly the same way until time τ_0 , when the first player reaches his penalty.
3. Prove that the resulting cost-sharing method ξ is cross-monotonic and β -budget balanced.
4. Conclude that $M(\xi)$ is a $(1 + \alpha)\beta$ -approximate and β -budget balanced Moulin mechanism for the prize-collecting problem \mathcal{P} .

The rest of this section is devoted to the proof of Theorem 4.

3.1 Decomposition lemma

Given a subset $S \subseteq U$ of players, we use $C|_S$ to refer to the restriction of C to S , i.e., $C|_S : 2^S \rightarrow \mathbb{R}^+$ with $C|_S(T) = C(T)$ for all $T \subseteq S$. Similarly, we use $\xi|_S$ to refer to the restriction of ξ to S .

The following lemma will turn out to be crucial to prove Theorem 4.

Lemma 1 (Decomposition Lemma) *Consider a universe U of players, along with a non-decreasing cost function C and a β -budget balanced and cross-monotonic cost-sharing method ξ . Given a partition of U into two parts U_1 and U_2 , if the Moulin mechanism on sub-universe U_j is α_j -approximate for all $j \in \{1, 2\}$ with respect to the induced cost-sharing method $\xi|_{U_j}$ and the cost function $C|_{U_j}$, then the Moulin mechanism is $(\alpha_1 + \alpha_2)\beta$ -approximate for the entire set U with respect to ξ and C .*

Proof Let A be the final set of players returned by the Moulin mechanism when run on U . Since ξ satisfies β -cost recovery, we have $C(A) \leq \beta \sum_{i \in A} \xi_i(A)$ and hence,

$$\Pi(A) = C(A) + v(U \setminus A) \leq \sum_{i \in A} \beta \xi_i(A) + \sum_{i \in U \setminus A} v_i. \tag{4}$$

Define $A_j = A \cap U_j$ as the set of players in U_j that were accepted by the Moulin mechanism when run on U for $j \in \{1, 2\}$. Consider a run of the Moulin mechanism on U_j and let B_j be the final set of players for $j \in \{1, 2\}$.

Claim $B_j \subseteq A_j$ for all $j \in \{1, 2\}$.

Proof Let e_1, \dots, e_p be the elements in $U_j \setminus A_j$ in the order in which they are dropped in the Moulin mechanism when run on U . Assume for the sake of contradiction that $B_j \not\subseteq A_j$. Then there exists some $l \in [p]$ such that $e_l \in B_j$. Choose l as the smallest index with this property and let $i = e_l$. Let S be the set of players that are still part of the game in the Moulin run on U just before i is dropped. Note that by the choice of l we have $S \supseteq \{e_1, \dots, e_p\} \cup A_j \supseteq B_j$. Thus

$$v_i < \xi_i(S) \leq \xi_i(\{e_1, \dots, e_p\} \cup A_j) \leq \xi_i(B_j),$$

where the last two inequalities use the cross-monotonicity of ξ . This contradicts the fact that $i = e_l$ is part of the final set B_j of the Moulin run on U_j . \square

Notice that the cost share $\xi_i(A)$ of a player $i \in A_j \setminus B_j$ is at most the valuation v_i of player i by the termination condition of the Moulin mechanism. For a set $S_j \subseteq U_j$, define $\Pi_j(S_j) = C(S_j) + v(U_j \setminus S_j)$. As B_j is an α_j -approximate set of players, we then have $\Pi_j(B_j) \leq \alpha_j \Pi_j(S_j)$ for any set $S_j \subseteq U_j$.

Using (4), we can now upper bound $\Pi(A)$ as follows:

$$\begin{aligned} \Pi(A) &= C(A) + v(U \setminus A) \leq \sum_{i \in A} \beta \xi_i(A) + \sum_{i \in U \setminus A} v_i \\ &= \left(\sum_{i \in A_1} \beta \xi_i(A) + \sum_{i \in U_1 \setminus A_1} v_i \right) + \left(\sum_{i \in A_2} \beta \xi_i(A) + \sum_{i \in U_2 \setminus A_2} v_i \right). \end{aligned} \tag{5}$$

We upper-bound the first of the two parentheses on the right-hand side of the above inequality. An upper bound for the second parentheses is obtained analogously.

$$\sum_{i \in A_1} \beta \xi_i(A) + \sum_{i \in U_1 \setminus A_1} v_i = \sum_{i \in B_1} \beta \xi_i(A) + \sum_{i \in A_1 \setminus B_1} \beta \xi_i(A) + \sum_{i \in U_1 \setminus A_1} v_i \tag{6}$$

$$\leq \sum_{i \in B_1} \beta \xi_i(A) + \sum_{i \in A_1 \setminus B_1} \beta v_i + \sum_{i \in U_1 \setminus A_1} v_i \tag{7}$$

$$\leq \sum_{i \in B_1} \beta \xi_i(B_1) + \sum_{i \in U_1 \setminus B_1} \beta v_i \tag{8}$$

$$\leq \beta C(B_1) + \sum_{i \in U_1 \setminus B_1} \beta v_i \tag{9}$$

$$= \beta \Pi_1(B_1). \tag{10}$$

Inequality (7) uses the fact that player $i \in A_1 \setminus B_1$ is part of the final set of players returned by the Moulin mechanism when run on U_1 , and hence must have valuation at least its cost share. We then use cross-monotonicity of ξ and the fact that $\beta \geq 1$ to get (8). Inequality (9) uses the competitiveness of ξ , and the final inequality follows from the definition of Π_1 . Using the resulting inequality together with (5) yields

$$\Pi(A) \leq \beta (\Pi_1(B_1) + \Pi_2(B_2)) \leq \beta (\alpha_1 \Pi_1(S_1) + \alpha_2 \Pi_2(S_2)) \tag{11}$$

for any $S_1 \subseteq U_1, S_2 \subseteq U_2$, where we use the fact that the Moulin mechanism when run on U_j is α_j -approximate for $j \in \{1, 2\}$.

Finally, for any set $S \subseteq U$ and for $j \in \{1, 2\}$, define $S_j = S \cap U_j$. Note that since C is non-decreasing, $\Pi_j(S_j) = C(S_j) + v(U_j \setminus S_j) \leq C(S) + v(U \setminus S) = \Pi(S)$. Putting these together with (11), we get that $\Pi(A) \leq (\alpha_1 + \alpha_2)\beta\Pi(S)$ for any $S \subseteq U$, and hence the Moulin mechanism is $(\alpha_1 + \alpha_2)\beta$ -approximate. \square

3.2 Partitioning the players

Armed with the above lemma, let us consider the universe of players U for an instance of \mathcal{P} and divide them into two parts as follows:

- The “high-valuation” set U_1 are those players $i \in U$ with valuation $v_i \geq \pi(i)$.
- The “low-valuation” set U_2 are the remaining players $i \in U$ with $v_i < \pi(i)$.

We now show that ξ on the sub-universes U_1 and U_2 is 1-approximate and α -approximate, respectively. This together with the Decomposition Lemma and the fact that ξ is β -budget balanced proves that $M(\xi)$ is $(1 + \alpha)\beta$ -approximate.

We first prove the following High-Valuation Lemma:

Lemma 2 (High-Valuation Lemma) *The mechanism $M(\xi)$ is 1-approximate when restricted to the players in the high-valuation set U_1 .*

Proof By Property 1 (Theorem 4), $\xi_i(S) \leq \pi(i)$ for every set $S \subseteq U$ and every $i \in S$. Since $v_i \geq \pi(i) \geq \xi_i(S)$ for any $S \subseteq U_1$ and $i \in S$, the players in U_1 will never be rejected by the mechanism $M(\xi)$ when run on U_1 . Moreover, the set achieving the optimal social cost is also U_1 , and hence the Moulin mechanism gives the social optimum on the high-valuation set. \square

Suppose we compare the two runs $M(\xi)$ and $M(\bar{\xi})$ of the Moulin mechanisms with cost-sharing methods ξ and $\bar{\xi}$ with the same set of low-valuation players $S \subseteq U_2$. An immediate consequence of Property 2 (Theorem 4) is that as long as some player is eliminated in either of the runs of the Moulin mechanisms, there must be a player that the mechanisms could eliminate in *both the runs*.

Lemma 3 *Fix some $S \subseteq U_2$. Suppose there is a player $j \in S$ with $\xi_j(S) > v_j$ or $\bar{\xi}_j(S) > v_j$. Then there is a player $i \in S$ such that $\xi_i(S) > v_i$ and $\bar{\xi}_i(S) > v_i$.*

Proof Let τ_0 be the first point of time τ at which $\xi_i^\tau(S) = \pi(i)$ for some player $i \in S$; $\tau_0 = \infty$ if no such time exists. The claim clearly holds if $\tau_0 = \infty$ as all cost shares in $M(\xi)$ and $M(\bar{\xi})$ are the same. Otherwise, there exists some player $i \in S$ and some $\tau_0 = \tau$ such that $\xi_i^\tau(S) = \pi(i)$. Property 2 (Theorem 4) then implies that $\xi_i^\tau(S) = \bar{\xi}_i^\tau(S) = \pi(i) > v_i$. \square

The next lemma essentially shows that the penalties $\pi(i)$ play no role for the low-valuation players U_2 .

Lemma 4 *When starting with a set of low-valuation players U_2 , the final output $S^{M(\xi)} \subseteq U_2$ of the Moulin mechanism $M(\xi)$ is identical to the output $S^{M(\bar{\xi})} \subseteq U_2$ of the Moulin mechanism $M(\bar{\xi})$.*

Proof Lemma 3 states that we can always identify a player $i \in S$ that we may evict in both runs of $M(\xi)$ and $M(\bar{\xi})$ as long as some player is eliminated in either of the runs of the Moulin mechanism. We can then eliminate player i in both the runs and use induction to show that both runs end with the same players if we make the right choices. However, Fact 2 implies that any choices would lead to the same outputs, as we claim. \square

We can now prove the following Low-Valuation Lemma:

Lemma 5 (Low-Valuation Lemma) *Restricting our attention to the low-valuation set U_2 , the mechanism $M(\xi)$ is α -approximate if the mechanism $M(\bar{\xi})$ is α -approximate.*

Proof On the low-valuation players, the solution with the optimal social cost for $M(\xi)$ would never service a player i by paying her penalty $\pi(i)$, since it would be better to reject the player and pay $v_i < \pi(i)$. This implies that the optimal social cost Π^* for \mathcal{P} and the optimal social cost $\bar{\Pi}^*$ for $\bar{\mathcal{P}}$ are the same on U_2 . Also note that for every player set S the cost $C(S)$ of an optimal solution for \mathcal{P} is at most the cost $\bar{C}(S)$ of an optimal solution for $\bar{\mathcal{P}}$. Let Π and $\bar{\Pi}$ denote the social cost with respect to \mathcal{P} and $\bar{\mathcal{P}}$, respectively. Given these facts together with the fact that $M(\xi)$ and $M(\bar{\xi})$ output the same set S^M on the low-valuation instances, we conclude that

$$\begin{aligned} \Pi(S^M) &= v(U_2 \setminus S^M) + C(S^M) \leq v(U_2 \setminus S^M) + \bar{C}(S^M) \\ &= \bar{\Pi}(S^M) \leq \alpha \cdot \bar{\Pi}^* = \alpha \cdot \Pi^*. \end{aligned}$$

Here the last inequality follows because $M(\bar{\xi})$ is α -approximate. \square

4 Submodular cost sharing problems with penalties

As a first application, we consider the *submodular cost-sharing problem with penalties (PSC)*. We are given a set U of players with penalties $\pi : U \rightarrow \mathbb{R}^+$ and a cost function $c : 2^U \rightarrow \mathbb{R}$ which is *submodular*,¹ i.e.,

$$\forall S, T \subseteq U : c(S) + c(T) \geq c(S \cup T) + c(S \cap T). \tag{12}$$

A feasible solution is a partition of the player set U into sets S and $P = U \setminus S$. The players in S incur a total cost of $c(S)$ and each player $i \in P$ incurs a cost equal to the penalty $\pi(i)$. The goal is to determine a partition (S, P) such that the total cost $c(S) + \pi(P)$ is minimized.

Submodular costs constitute an important class of cost functions. Several natural cost-sharing problems exhibit a cost function that is submodular. For example, in the *fixed tree multicast problem* we are given a tree $T = (V, E)$ with root node r and edge costs $c : E \rightarrow \mathbb{R}^+$. Each player $i \in U$ is uniquely associated with a node u_i in V . The cost $c(S)$ to serve a subset $S \subseteq V$ is defined as the cost $\sum_{e \in T'} c(e)$ of the

¹ An equivalent definition of submodularity is that c has non-increasing *marginal costs*, i.e., for every player $i \in U$ and every set $S \subseteq T \subseteq U \setminus \{i\}$, $c(S \cup \{i\}) - c(S) \geq c(T \cup \{i\}) - c(T)$.

smallest subtree T' of T that contains all of nodes in S . It is not hard to verify that c is submodular.

Dutta and Ray [12] and Jain and Vazirani [23] gave a cross-monotonic and budget balanced cost-sharing method ξ^{SC} for the submodular cost-sharing problem. The cost-sharing method in [23] is derived via a primal-dual algorithm which also constitutes the basis of our approach here. It is shown implicitly by Bleischwitz et al. [9] that the resulting Moulin mechanism $M(\xi^{\text{SC}})$ is $2H_k$ -approximate.²

Here we adapt the cross-monotonic cost-sharing method ξ^{SC} of Jain and Vazirani [23] and obtain a cross-monotonic and budget balanced cost-sharing method ξ^{PSC} for the submodular cost-sharing problem with penalties. With the help of our Lifting Theorem we will then be able to show that the resulting Moulin mechanism $M(\xi^{\text{PSC}})$ is $(2H_k + 1)$ -approximate. In summary, the result proven in this section is as follows.

Theorem 5 *There is a Moulin mechanism for the submodular cost-sharing problem with penalties that is budget balanced and $(2H_k + 1)$ -approximate.*

This result is almost best possible: Roughgarden and Sundararajan [35] showed that every budget balanced Moulin mechanism for the submodular cost-sharing problem is no better than H_k -approximate.

4.1 LP formulation

The following is a natural integer linear programming formulation of the submodular cost-sharing problem with penalties:

$$\min \sum_{S \subseteq U} c(S)x_S + \sum_{i \in U} \pi(i)z_i \tag{ILP}$$

$$\text{s.t.} \quad \sum_{S \subseteq U: i \in S} x_S + z_i \geq 1 \quad \forall i \in U \tag{13}$$

$$x_S, z_i \in \{0, 1\} \quad \forall S \subseteq U, \forall i \in U. \tag{14}$$

Here x_S and z_i are decision variables: x_S is equal to 1 if and only if the set of served players is $S \subseteq U$ and z_i is set to 1 for a player $i \in U$ if and only if $i \in P$. Note that because c is submodular the sets (S, P) induced by an optimal solution constitute a partition of the player set U .

By relaxing the integrality constraint (14) and dualizing the resulting linear program, we obtain the following linear program:

$$\max \sum_{i \in U} \xi_i \tag{D}$$

$$\text{s.t.} \quad \sum_{i \in S} \xi_i \leq c(S) \quad \forall S \subseteq U \tag{15}$$

² This follows from Theorem 4 in [9] and the observation that for submodular cost functions their egalitarian mechanism coincides with the Moulin mechanism $M(\xi^{\text{SC}})$.

$$\xi_i \leq \pi(i) \quad \forall i \in U \tag{16}$$

$$\xi_i \geq 0 \quad \forall i \in U. \tag{17}$$

The dual value ξ_i of player $i \in U$ can be interpreted as the *cost share* of i . Constraint (15) bounds the total cost share of all players in a set $S \subseteq U$ to be at most the cost $c(S)$ of that set. We say that a set $S \subseteq U$ is *tight* if and only if $\sum_{i \in S} \xi_i = c(S)$. Constraint (16) restricts player i 's cost share to not exceed his penalty $\pi(i)$.

4.2 Cost share definition and construction of solution

We first derive some properties of feasible solutions to the dual linear program (D) which we will then use to define our cost-sharing method. Most of the proofs below follow along similar lines as the corresponding ones in [23].

Lemma 6 [23] *Let ξ be a feasible solution to (D). If two sets $S_1, S_2 \subseteq U$ are tight, then so is $S_1 \cup S_2$.*

Proof By submodularity of c , we have

$$\begin{aligned} c(S_1 \cup S_2) &\leq c(S_1) + c(S_2) - c(S_1 \cap S_2) \\ &\leq \sum_{i \in S_1} \xi_i + \sum_{i \in S_2} \xi_i - \sum_{i \in S_1 \cap S_2} \xi_i = \sum_{i \in S_1 \cup S_2} \xi_i. \end{aligned}$$

The second inequality follows because S_1 and S_2 are tight, and because ξ is feasible and thus $\sum_{i \in S_1 \cap S_2} \xi_i \leq c(S_1 \cap S_2)$. We conclude that $S_1 \cup S_2$ is tight. \square

Corollary 1 [23] *Let ξ be a feasible solution to (D). Then there is a unique maximal tight set, which is simply the union of all tight sets.*

We next define our cost-sharing method ξ^{PSC} . Our algorithm is a natural adaptation of the cost-sharing method ξ^{SC} given by Jain and Vazirani [23]. Basically, we extend their cost-sharing method simply by ensuring that the dual constraint (16) is satisfied additionally. Both algorithms coincide if $\pi_i = \infty$ for every $i \in U$.

Our algorithm can be seen as a process over time. Let ξ_i^τ be the cost share of player i at time τ . We will make sure that $\xi^\tau = (\xi_i^\tau)_{i \in U}$ is dual feasible throughout the execution of the algorithm. Initially, the algorithm starts with all cost shares equal to 0, i.e., $\xi_i^0 = 0$ for every $i \in U$. Let S^τ be the unique maximal tight set at time τ . Note that S^τ exists by Corollary 1. We call a player $i \in U$ *active* at time τ if i is not contained in any tight set and has not reached his penalty, i.e., $i \notin S^\tau$ and $\xi_i^\tau < \pi(i)$; otherwise, we say that i is *inactive*. As time progresses, the algorithm raises the cost shares of all active players at the same rate until either a new set becomes tight or some player i reaches his penalty $\pi(i)$. In the former event, all players in the new maximal tight set S^τ become inactive (some might have been inactive before). In the latter event, player i becomes inactive. The algorithm continues in this manner until eventually all players are inactive.

Let $\xi^{\text{PSC}} = (\xi_i^{\text{PSC}})_{i \in U}$ refer to the final cost shares. Let S be the unique maximal tight set S^τ at termination of the algorithm and let $P = U \setminus S$ be the set of the remaining players. Our algorithm returns (S, P) as the final solution.

Remark 2 The algorithm can be implemented to run in polynomial time. The only non-trivial part is to determine the unique maximal tight set S^τ . This can be achieved by using a polynomial time algorithm for the minimization of a submodular function [21,37]. The idea is similar to the one described in [23]: It is not hard to verify that the function $c'(S) = c(S) - \sum_{i \in S} f_i$ is submodular, where f_i is fixed for every player $i \in S$. We set $f_i = \xi_i^\tau$ if $i \in S$ is inactive at time τ and $f_i = \tau$ otherwise. By doing a binary search on τ , we can then find the smallest time τ at which there is a set $S \subseteq U$ such that $c'(S)$ is a small negative number. Then S is the set that will become tight next.

4.3 Cross-monotonicity and budget balance

The next lemma shows that the cost shares correspond to a feasible dual solution at all times.

Lemma 7 *At any point of time τ , the cost shares $\xi^\tau = (\xi_i^\tau)$ constitute a feasible solution to (D).*

Proof Note that if a set S becomes tight at time τ then all players in S become inactive and remain inactive at all times $\tau' > \tau$. Also once a player $i \in U$ reaches his penalty $\pi(i)$ at time τ , i remains inactive at all times $\tau' > \tau$. Constraints (15) and (16) of the dual linear program (D) are therefore satisfied. □

The proof that ξ^{PSC} is cross-monotonic is similar to the one in [23].

Lemma 8 *The cost-sharing method ξ^{PSC} is cross-monotonic.*

Proof Let $U_1 \subset U_2$ and consider the two runs of the algorithm on player sets U_1 and U_2 , respectively. Let I_1^τ and I_2^τ be the set of players that are inactive at time τ in the runs on U_1 and U_2 , respectively. A crucial property of our algorithm is that all players that are active at time τ have the same cost share τ . As a consequence, in order to show that ξ^{PSC} is cross-monotonic it suffices to show that $I_1^\tau \subseteq I_2^\tau$ at all times τ .

Let S_1^τ and S_2^τ be the maximal tight sets at time τ in the runs on U_1 and U_2 , respectively. First note that every player $i \in I_1^\tau \setminus S_1^\tau$ is inactive at time τ in the U_1 -run because he reached his penalty, i.e., $\tau \geq \pi(i)$. But then i cannot be active at time τ in the U_2 -run and thus $i \in I_2^\tau$.

The claim now follows if we can show that $S_1^\tau \subseteq S_2^\tau$. For notational convenience, let $S_1 = S_1^\tau$ and $S_2 = S_2^\tau$. Also let ξ^1 and ξ^2 be the cost shares at time τ in the runs on U_1 and U_2 , respectively. We have

$$c(S_1 \cup S_2) \leq c(S_1) + c(S_2) - c(S_1 \cap S_2) \tag{18}$$

$$\leq \sum_{i \in S_1} \xi_i^1 + \sum_{i \in S_2} \xi_i^2 - \sum_{i \in S_1 \cap S_2} \xi_i^1 \tag{19}$$

$$= \sum_{i \in S_1 \setminus S_2} \xi_i^1 + \sum_{i \in S_2} \xi_i^2 \tag{20}$$

$$\leq \sum_{i \in S_1 \cup S_2} \xi_i^2. \tag{21}$$

Here inequality (18) follows from the submodularity of c . Inequality (19) holds because S_1 and S_2 are tight with respect to ξ^1 and ξ^2 , respectively, and ξ^1 is dual feasible. Inequality (21) follows from the fact that for every $i \in S_1 \setminus S_2$ we have $\xi_i^1 \leq \xi_i^2$. To see this, we distinguish two cases: (i) i is active at time τ in the U_2 -run. Then $\xi_i^2 = \tau \geq \xi_i^1$. (ii) i is inactive at time τ in the U_2 -run. Since $i \notin S_2$, i must be inactive because he reached his penalty, i.e., $\xi_i^2 = \pi(i)$. But then $\xi_i^2 = \pi(i) \geq \xi_i^1$.

The above inequality implies that $S_1 \cup S_2$ is tight with respect to ξ^2 . Since S_2 is the maximal tight set in the U_2 -run, we have $S_1 \cup S_2 \subseteq S_2$ and thus $S_1 \subseteq S_2$ which concludes the proof. □

We finally show that ξ^{PSC} is budget balanced.

Lemma 9 *The cost-sharing method ξ^{PSC} is budget balanced.*

Proof Let (S^*, P^*) be an optimal solution for the given instance of the submodular cost-sharing problem with penalties. Recall that by Lemma 7, ξ^{PSC} is a feasible solution to (D). By weak duality, we obtain

$$\sum_{i \in U} \xi_i^{PSC} \leq c(S^*) + \pi(P^*).$$

Let (S, P) be the partition output by the algorithm. When the algorithm terminates, the set S is tight and thus $\sum_{i \in S} \xi_i^{PSC} = c(S)$. Moreover, for every player $i \in P$ we have $\xi_i^{PSC} = \pi(i)$. Thus

$$\sum_{i \in U} \xi_i^{PSC} = c(S) + \pi(P).$$

The cost shares ξ^{PSC} are thus budget balanced. □

4.4 Social cost approximation

Recall that the cost-sharing method ξ^{SC} for the submodular cost-sharing problem without penalties is known to be $2H_k$ -approximate [9]. It is not hard to verify that ξ^{SC} and ξ^{PSC} satisfy the two properties of our Lifting Theorem (Theorem 4). This together with the fact that ξ^{PSC} is budget balanced proves the following lemma.

Corollary 2 *The Moulin mechanism $M(\xi^{PSC})$ is $(2H_k + 1)$ -approximate.*

5 Metric facility location problem with penalties

In the *metric facility location problem with penalties (PFL)* we are given a universe of players U with non-negative penalties $\pi : U \rightarrow \mathbb{R}^+$, a set of facilities F with a non-negative opening costs $f : F \rightarrow \mathbb{R}^+$, and a metric c on $F \cup U$. The goal is to open a subset $O \subseteq F$ of facilities and for each player $i \in U$ either connect i to the nearest open facility or pay his penalty $\pi(i)$. Let $Q \subseteq U$ be the subset of players for which we pay their penalties. The objective is to find a feasible solution (O, Q) that minimizes

$$\sum_{p \in O} f(p) + \sum_{i \notin Q} \min_{p \in O} c(i, p) + \sum_{i \in Q} \pi(i).$$

Pál and Tardos [32] gave a cross-monotonic and 3-budget balanced cost-sharing method ξ^{FL} for the metric facility location problem (without penalties). Their cost-sharing method is based on a primal-dual interpretation of the approximation algorithm by Mettu and Plaxton [28]. Roughgarden and Sundararajan [35] later showed that the Moulin mechanism $M(\xi^{FL})$ is $(H_k + 3)$ -approximate, where k refers to the number of players.

We show below that the algorithm of Pál and Tardos can naturally be adapted to give a cross-monotonic and 3-budget balanced cost-sharing method ξ^{PFL} for the metric facility location problem with penalties. Using our lifting technique introduced in Sect. 3, we will show that the resulting Moulin mechanism is $3(H_k + 4)$ -approximate. We summarize our result in the following theorem.

Theorem 6 *There is a Moulin mechanism for the metric facility location problem with penalties that is 3-budget balanced and $3(H_k + 4)$ -approximate.*

Note that the above budget balance and social cost approximation guarantees are essentially best possible: Immorlica et al. [20] showed that there is no cross-monotonic cost-sharing scheme for the metric facility location problem with budget balance factor less than 3. Moreover, Roughgarden and Sundararajan [35] showed that every β -budget balanced Moulin mechanism for the metric facility location problem cannot be better than H_k/β -approximate.

5.1 LP formulation

The following is a natural integer linear programming formulation for the metric facility location problem with penalties.

$$\min \sum_{p \in F} f(p)y_p + \sum_{i \in U} c(i, p)x_{ip} + \sum_{i \in U} \pi(i)z_i \tag{ILP}$$

$$\text{s.t. } \sum_{p \in F} x_{ip} + z_i \geq 1 \quad \forall i \in U \tag{22}$$

$$x_{ip} \leq y_p \quad \forall p \in F, \quad \forall i \in U \tag{23}$$

$$y_p, x_{ip}, z_i \in \{0, 1\} \quad \forall p \in F, \quad \forall i \in U. \tag{24}$$

The interpretation of the indicator variables is as follows. Variable y_p is set to 1 if and only if facility $p \in F$ is open. The indicator variable x_{ip} equals 1 if and only if player $i \in U$ is assigned to facility $p \in F$. Further, z_i is 1 if player $i \in Q$ and 0 otherwise. Constraint (22) ensures that every player $i \in U$ is either assigned to some facility or part of Q . Constraint (23) enforces that a player i is assigned to a facility p only if p is open.

By relaxing the integrality constraints (24) and dualizing the resulting linear program we obtain:

$$\max \sum_{i \in U} \xi_i \tag{D}$$

$$\text{s.t. } \sum_{i \in U} \kappa_{ip} \leq f(p) \quad \forall p \in F \tag{25}$$

$$\xi_i - \kappa_{ip} \leq c(i, p) \quad \forall p \in F, \forall i \in U \tag{26}$$

$$\xi_i \leq \pi(i) \quad \forall i \in U \tag{27}$$

$$\xi_i, \kappa_{ip} \geq 0 \quad \forall p \in F, \forall i \in U. \tag{28}$$

We can think of ξ_i as being the cost share of player $i \in U$ and κ_{ip} as the contribution of player i towards facility $p \in F$. Constraint (25) imposes that the total contribution of all players towards a facility p does not exceed its opening cost $f(p)$. Constraint (26) enforces that for every facility $p \in F$ the cost share ξ_i of player i is at most the connection cost $c(i, p)$ plus the contribution κ_{ip} towards p . Constraint (27) requires that the cost share ξ_i of each player i is at most the penalty $\pi(i)$.

5.2 Cost share definition and construction of solution

We obtain a cross-monotonic cost-sharing method ξ^{FFL} based on the dual linear program (D). Our cost-sharing method is similar to the cost-sharing method ξ^{FL} of Pál and Tardos [32] for the metric facility location problem. In fact, for the special case that all penalties are set to ∞ our algorithm described below coincides with the one of Pál and Tardos.

We consider the algorithm as a process over time. For each player $i \in U$ we grow a *ghost ball* uniformly and at unit rate around i . While in [32] these ghosts are grown to infinity, we cannot do this here because of the penalties of the players. Instead, we will only be able to grow the ghost of i in the time interval $[0, \pi(i)]$. That is, the ghost of i at time τ is a ball centered at i with radius $\min(\tau, \pi(i))$.

The ghost of i *touches* a facility $p \in F$ at time τ if $c(i, p) \leq \min(\tau, \pi(i))$. The ghosts that touch a facility p contribute towards *filling* p . The *contribution* of ghost i towards facility p at time τ is

$$\kappa_{ip}^\tau = \max(0, \min(\tau, \pi(i)) - c(i, p)).$$

A facility $p \in F$ is said to be *full* at time τ if $\sum_{i \in U} \kappa_{ip}^\tau \geq f(p)$. Let $\tau(p)$ be the first point of time when p becomes full; $\tau(p) = \infty$ if no such time exists. Further,

let $S_p \subseteq U$ be the set of players that contributed towards filling p at time $\tau(p)$; more formally,

$$S_p = \left\{ i \in U : \kappa_{ip}^{\tau(p)} > 0 \right\}.$$

The cost shares are defined as follows. The cost share ξ_i^{PFL} of player i is grown at unit rate until some facility it touches becomes full or i 's ghost touches some full facility or i 's cost share equals $\pi(i)$. More formally,

$$\xi_i^{\text{PFL}} = \min \left(\min_{p:i \in S_p} \tau(p), \min_{p:i \notin S_p} c(i, p), \pi(i) \right). \tag{29}$$

We use the following simple rule of Mettu and Plaxton [28] to determine which facilities are opened: Whenever a facility $p \in F$ becomes full at time $\tau(p)$, we open p if and only if there is no other facility $q \in F$ within distance $c(p, q) \leq 2\tau(p)$ that is already open. Let O be the set of facilities that are opened this way. Let Q be the set of players whose cost share equals their penalties, i.e., $Q = \{i \in U : \xi_i^{\text{PFL}} = \pi(i)\}$. We assign each remaining player $i \notin Q$ to his closest open facility in O .

5.3 Cross-monotonicity

As in [32], the proof of ξ^{PFL} being cross-monotonic follows straightforwardly from the following observation: By adding more players, each facility can only become full earlier and thus each individual cost share can only decrease.

Lemma 10 *The cost shares ξ^{PFL} are cross-monotonic.*

5.4 Budget balance

We show that the cost-sharing method ξ^{PFL} as defined above is 3-budget balanced. The following lemma establishes competitiveness of ξ^{PFL} .

Lemma 11 *Let (O^*, Q^*) be an optimal solution to an instance of the metric facility location problem with penalties. The cost shares ξ^{PFL} satisfy*

$$\sum_{i \in U} \xi_i^{\text{PFL}} \leq \sum_{p \in O^*} f(p) + \sum_{i \notin Q^*} \min_{p \in O^*} c(i, p) + \sum_{i \in Q^*} \pi(i).$$

Proof Observe that the final cost shares ξ_i^{PFL} together with their contributions κ_{ip} constitute a feasible dual solution to (D). The lemma simply follows by weak duality. □

The following lemma shows that for any two different open facilities p and q the corresponding contributor sets S_p and S_q are disjoint (see also [32]).

Lemma 12 *If facilities p and q , $p \neq q$, are both open, then their contributor sets S_p and S_q are disjoint.*

Proof Let $\tau(p) \geq \tau(q)$. For contradiction, assume that there is some player $i \in S_p \cap S_q$. Then $c(i, p) \leq \min(\tau(p), \pi(i))$ and $c(i, q) \leq \min(\tau(q), \pi(i))$. Exploiting the triangle inequality we obtain

$$c(p, q) \leq c(i, p) + c(i, q) \leq \tau(p) + \tau(q) \leq 2\tau(p),$$

which is a contradiction to p being open. □

Consider an open facility $p \in O$. We first bound the opening cost of p plus the total cost of connecting all players in S_p to p . Throughout the analysis we overestimate the connection cost of all players $i \in S_p \setminus Q$ by assuming that they are assigned to p (instead of their closest open facility). At time $\tau(p)$ when p is opened, we have

$$\begin{aligned} f(p) &= \sum_{i \in S_p} \kappa_{ip}^{\tau(p)} = \sum_{i \in S_p} \min(\tau(p), \pi(i)) - c(i, p) \\ &\leq \sum_{i \in S_p \cap Q} \xi_i^{\text{PFL}} + \sum_{i \in S_p \setminus Q} \min(\tau(p), \pi(i)) - \sum_{i \in S_p} c(i, p), \end{aligned} \tag{30}$$

where the inequality follows because $\xi_i^{\text{PFL}} = \pi(i)$ for every $i \in Q$.

Lemma 13 *Let p be an open facility and let $i \in S_p \setminus Q$. Then $\xi_i^{\text{PFL}} \geq \frac{1}{3} \min(\tau(p), \pi(i))$.*

Proof For the sake of a contradiction assume that $\xi_i^{\text{PFL}} < \frac{1}{3} \min(\tau(p), \pi(i))$. Let q be the first full facility that i touches, i.e., $\xi_i^{\text{PFL}} \geq c(i, q)$ and $\xi_i^{\text{PFL}} \geq \tau(q)$. Note that q must exist because $i \notin Q$. Recall that $c(i, p) \leq \min(\tau(p), \pi(i))$ because $i \in S_p$. Exploiting the triangle inequality, we obtain

$$c(p, q) \leq c(i, p) + c(i, q) \leq \min(\tau(p), \pi(i)) + \xi_i^{\text{PFL}} < 2 \min(\tau(p), \pi(i)) \leq 2\tau(p)$$

and thus q cannot be open. But then there must exist a facility q' that prevented q from being opened at time $\tau(q)$, i.e., $c(q, q') \leq 2t(q) \leq 2\xi_i^{\text{PFL}}$. Observe that $q' \neq p$ because $\tau(q') \leq \tau(q) \leq \xi_i^{\text{PFL}} < \tau(p)$. We obtain

$$\begin{aligned} c(p, q') &\leq c(i, p) + c(i, q) + c(q, q') \leq \min(\tau(p), \pi(i)) + 3\xi_i^{\text{PFL}} \\ &< 2 \min(\tau(p), \pi(i)) \leq 2\tau(p), \end{aligned}$$

which is a contradiction to p being open. □

Exploiting inequality (30) and Lemma 13, we obtain for every open facility $p \in O$

$$f(p) + \sum_{i \in S_p} c(i, p) \leq \sum_{i \in S_p \cap Q} \xi_i^{\text{PFL}} + 3 \sum_{i \in S_p \setminus Q} \xi_i^{\text{PFL}}. \tag{31}$$

We also need to bound the connection costs of all players $i \notin Q$ that do not belong to any contributor set of an open facility. Define $R = \cup_{p \in O} S_p$.

Lemma 14 Consider a player $i \notin Q \cup R$. Then $\xi_i^{PFL} \geq \frac{1}{3} \min_{p \in O} c(i, p)$.

Proof Let q be the first full facility that i touches, i.e., $\xi_i^{PFL} \geq c(i, q)$ and $\xi_i^{PFL} \geq \tau(q)$. Note that q must exist because $i \notin Q$. If q is open, then the claim follows because $\xi_i^{PFL} \geq c(i, q)$. If q is not open then there must exist a facility q' that prevented q from being opened and thus $c(q, q') \leq 2\tau(q)$. We obtain

$$c(i, q') \leq c(i, q) + c(q, q') \leq \xi_i^{PFL} + 2\tau(q) \leq 3\xi_i^{PFL},$$

which concludes the proof. □

Lemma 15 Let (O, Q) be the solution computed by the above algorithm. The cost shares ξ^{PFL} satisfy

$$\sum_{i \in U} \xi_i^{PFL} \geq \frac{1}{3} \left(\sum_{p \in O} f(p) + \sum_{i \notin Q} \min_{p \in O} c(i, p) + \sum_{i \in Q} \pi(i) \right).$$

Proof By exploiting inequality (31), we obtain

$$\sum_{p \in O} \left(f(p) + \sum_{i \in S_p} c(i, p) \right) \leq \sum_{p \in O} \left(\sum_{i \in S_p \cap Q} \xi_i^{PFL} + 3 \sum_{i \in S_p \setminus Q} \xi_i^{PFL} \right) \tag{32}$$

Since $\xi_i^{PFL} = \pi(i)$ for every $i \in Q$, we have

$$\sum_{i \in Q} \pi(i) = \sum_{i \in Q} \xi_i^{PFL}. \tag{33}$$

By Lemma 14, we can bound the connection costs of all players in $U \setminus (Q \cup R)$ by

$$\sum_{i \notin Q \cup R} \min_{p \in O} c(i, p) \leq 3 \sum_{i \notin Q \cup R} \xi_i^{PFL}. \tag{34}$$

Combining (32), (33) and (34) we obtain that the total cost of the solution (O, Q) is at most

$$2 \sum_{i \in Q} \xi_i^{PFL} + 3 \sum_{i \notin Q} \xi_i^{PFL} \leq 3 \sum_{i \in U} \xi_i^{PFL},$$

which concludes the proof. □

5.5 Social cost approximation

It is easy to verify that our cost-sharing method ξ^{PFL} extending the cost-sharing method ξ^{FL} of Pál and Tardos [32] satisfies Properties 1 and 2 of the Lifting Theorem (Theorem 4). Roughgarden and Sundararajan [35] showed that the Moulin mechanism $M(\xi^{\text{FL}})$ is $(H_k + 3)$ -approximate, where k refers to the number of players. By our Lifting Theorem and the fact that ξ^{PFL} is 3-budget balanced, we conclude that $M(\xi^{\text{PFL}})$ is $3(H_k + 4)$ -approximate.

Corollary 3 *The Moulin mechanism $M(\xi^{\text{PFL}})$ is $3(H_k + 4)$ -approximate.*

6 Prize-collecting Steiner forest problem

In the *prize-collecting Steiner forest problem (PCSF)* we are given an undirected graph $G = (V, E)$ with edge costs $c : E \rightarrow \mathbb{R}^+$, a set of k terminal pairs $R = \{(s_i, t_i)\}_{i \in [k]}$, and penalties $\pi : R \rightarrow \mathbb{R}^+$. A feasible solution (F, Q) consists of a forest F and a subset Q of terminal pairs such that for all $(s_i, t_i) \in R$ either s_i and t_i are connected by F or $(s_i, t_i) \in Q$. The objective is to compute a feasible solution (F, Q) of minimum cost $c(F) + \pi(Q)$. Subsequently, we identify the set U of players with the terminal pairs in R .

Könemann et al. [26] gave a cross-monotonic and 2-budget balanced cost-sharing method ξ^{SF} for the Steiner forest problem (without penalties). Their algorithm is based on the primal-dual Steiner forest algorithm of Agrawal et al. [1], but requires some non-trivial adaptations to ensure cross-monotonicity. In particular, the dual solution constructed by the algorithm is in general not feasible. Chawla et al. [11] showed that the cost-sharing method ξ^{SF} is also $O(\log^2 k)$ -approximate. A simple proof that they are $O(\log^3 k)$ -approximate is given in Sect. 7.

Here we extend the algorithm of Könemann et al. [26] and derive a cross-monotonic and 3-budget balanced cost-sharing method ξ^{PSF} for the prize-collecting Steiner forest problem. Applying the Lifting Theorem, we will be able to show that the resulting Moulin mechanism is $O(\log^2 k)$ -approximate.

Theorem 7 *There is a Moulin mechanism $M(\xi^{\text{PSF}})$ for the prize-collecting Steiner forest problem that is 3-budget balanced and $O(\log^2 k)$ -approximate.*

Note that these bounds are almost best possible: Könemann et al. [26] showed that there is no cross-monotonic cost-sharing method that achieves a budget balance factor better than 2 for the Steiner tree problem. Moreover, Roughgarden and Sundararajan [35] showed that every constant budget balanced Moulin mechanism for the Steiner tree problem cannot be better than $\Omega(\log^2 k)$ -approximate.

Hajiaghayi and Jain [19] gave a combinatorial 3-approximation algorithm for prize-collecting Steiner forest problem. Our algorithm is also combinatorial and achieves the same approximation guarantee. The following corollary might therefore be of independent interest.

Corollary 4 *There is a combinatorial 3-approximation algorithm for the prize-collecting Steiner forest problem.*

6.1 LP formulation

We assume that every node $u \in V$ belongs to at most one terminal pair in R . Note that this is without loss of generality because we can simply replace each node $u \in V$ that is contained in $l \geq 2$ terminal pairs by l copies, one for each terminal pair, which are all connected by zero cost edges. We use $V(R)$ to denote the set of terminal nodes in R . For a terminal $u \in V(R)$, let \bar{u} be the mate of u , i.e., $(u, \bar{u}) \in R$.

Consider a cut $S \subseteq V$. We say S separates a terminal pair $(u, \bar{u}) \in R$ if and only if $|\{u, \bar{u}\} \cap S| = 1$. We also write $(u, \bar{u}) \odot S$ if and only if (u, \bar{u}) is separated by S . A cut S that separates at least one terminal pair is called a *Steiner cut*. Let \mathcal{S} denote the set of all Steiner cuts. For a cut $S \subseteq V$, we use $\delta(S)$ to refer to the set of edges $(u, v) \in E$ that cross S , i.e., $\delta(S) = \{(u, v) \in E : |\{u, v\} \cap S| = 1\}$.

A natural integer programming formulation for the prize-collecting Steiner forest problem is as follows:

$$\begin{aligned} \min \quad & \sum_{e \in E} c(e) \cdot x_e + \sum_{(u, \bar{u}) \in R} \pi(u, \bar{u}) \cdot z_{u\bar{u}} & \text{(ILP)} \\ \text{s.t.} \quad & \sum_{e \in \delta(S)} x_e + z_{u\bar{u}} \geq 1 \quad \forall S \in \mathcal{S}, \forall (u, \bar{u}) \odot S & \text{(35)} \\ & x_e, z_{u\bar{u}} \in \{0, 1\} \quad \forall e \in E, \forall (u, \bar{u}) \in R. \end{aligned}$$

We have a decision variable x_e for every edge $e \in E$ and a decision variable $z_{u\bar{u}}$ for every terminal pair $(u, \bar{u}) \in R$: $x_e = 1$ if and only if $e \in F$ and $z_{u\bar{u}} = 1$ if and only if $(u, \bar{u}) \in Q$. Constraint (35) ensures that each Steiner cut $S \in \mathcal{S}$ is either crossed by an edge of F , or all separated terminal pairs $(u, \bar{u}) \odot S$ are part of Q .

In the dual of the linear programming relaxation (LP) of (ILP) we have a non-negative dual variable $\xi_{S,u\bar{u}}$ for every Steiner cut $S \in \mathcal{S}$ and every pair $(u, \bar{u}) \in R$ with $(u, \bar{u}) \odot S$:

$$\begin{aligned} \max \quad & \sum_{S \in \mathcal{S}} \sum_{(u, \bar{u}) \odot S} \xi_{S,u\bar{u}} & \text{(D)} \\ \text{s.t.} \quad & \sum_{S \in \mathcal{S}: e \in \delta(S)} \sum_{(u, \bar{u}) \odot S} \xi_{S,u\bar{u}} \leq c(e) \quad \forall e \in E & \text{(36)} \\ & \sum_{S \in \mathcal{S}: (u, \bar{u}) \odot S} \xi_{S,u\bar{u}} \leq \pi(u, \bar{u}) \quad \forall (u, \bar{u}) \in R & \text{(37)} \\ & \xi_{S,u\bar{u}} \geq 0 \quad \forall S \in \mathcal{S}, (u, \bar{u}) \odot S. \end{aligned}$$

It is convenient to associate a dual solution $(\xi_{S,u\bar{u}})_{S \in \mathcal{S}, (u, \bar{u}) \odot S}$ of (D) with dual values $(y_S)_{S \in \mathcal{S}}$ for all Steiner cuts $S \in \mathcal{S}$. To this aim, we define the dual y_S of a Steiner cut $S \in \mathcal{S}$ simply as the total cost share of all its separated terminal pairs:

$$y_S = \sum_{(u, \bar{u}) \odot S} \xi_{S,u\bar{u}}.$$

We can think of $\xi_{S,u\bar{u}}, (u, \bar{u}) \odot S$, as the *cost share* that terminal pair (u, \bar{u}) receives from dual y_S of S . Define the total cost share of (u, \bar{u}) as

$$\xi_{u\bar{u}} = \sum_{S \in \mathcal{S}: S \odot (u, \bar{u})} \xi_{S,u\bar{u}}.$$

Constraint (36) of the linear program (D) then requires that for every edge $e \in E$ the total dual of all Steiner cuts $S \in \mathcal{S}$ that cross e is at most the cost $c(e)$ of this edge. We call an edge $e \in E$ *tight* if (36) holds with equality. Constraint (37) states that the total cost share $\xi_{u\bar{u}}$ of terminal pair (u, \bar{u}) is at most its penalty $\pi(u, \bar{u})$.

6.2 Cost share definition and construction of solution

We obtain a cross-monotonic cost sharing method ξ^{PSF} based on the dual linear program (D). Our cost-sharing method generalizes the cost-sharing method by Könemann et al. [26] for the Steiner forest problem. Both algorithms coincide if the penalties are set to ∞ .

The algorithm grows dual values of certain cuts and distributes this growth among the terminal pairs. A subtle point of our algorithm is that the dual solution generated by the algorithm may be infeasible for (D). As a consequence, proving that the resulting cost-sharing method ξ^{PSF} satisfies competitiveness is more involved (see Lemma 17 below) because it does not simply follow from weak duality.

There are two reasons that cause the infeasibility of the dual solution:

1. We also raise dual values y_S of cuts $S \subseteq V$ that do *not* correspond to Steiner cuts. A terminal pair (u, \bar{u}) may therefore receive cost share $\xi_{S,u\bar{u}}$ from a non-Steiner cut $S \subseteq V$.
2. A terminal pair (u, \bar{u}) may also receive cost share $\xi_{S,u\bar{u}}$ from a cut S that does not separate (u, \bar{u}) .

However, our algorithm maintains the invariant that a terminal pair (u, \bar{u}) only receives cost share $\xi_{S,u\bar{u}}$ from a cut $S \subseteq V$ that either separates or entirely contains (u, \bar{u}) , i.e., $(u, \bar{u}) \odot S$ or $\{u, \bar{u}\} \subseteq S$.

Our algorithm PSF can be described as a process over time. At time $\tau = 0$, $x_e^\tau = 0$ for every edge $e \in E$, $z_{u\bar{u}}^\tau = 0$ for every terminal pair $(u, \bar{u}) \in R$ and $y_S^\tau = 0$ for every $S \subseteq V$.³ We use $\xi_{u\bar{u}}^\tau$ to refer to the cost share of terminal pair (u, \bar{u}) at time τ .

Let F^τ be the forest at time τ that corresponds to $(x_e^\tau)_{e \in E}$, i.e.,

$$F^\tau = \{e \in E : x_e^\tau = 1\}.$$

Similarly, let Q^τ be the set of all terminal pairs $(u, \bar{u}) \in R$ such that $z_{u\bar{u}}^\tau = 1$.

³ The latter initialization is only done implicitly. The algorithm will ensure that the cuts with positive dual value form a laminar family.

We define \bar{F}^τ as the set of all edges that are tight at time τ , i.e.,

$$\bar{F}^\tau = \left\{ e \in E : \sum_{S \subseteq V} y_S^\tau = c(e) \right\}.$$

It is important to realize the difference between F^τ and \bar{F}^τ : F^τ represents the partial solution at time τ and will constitute a forest at all times. \bar{F}^τ will be used to guide the growth of dual values. PSF maintains the invariant that only tight edges are part of the current forest at all times, i.e., $F^\tau \subseteq \bar{F}^\tau$ for every $\tau \geq 0$.

We use the term *moat* to refer to a (maximal) connected component M^τ in \bar{F}^τ . Every moat M^τ induces a cut which is defined by the set of nodes $V(M^\tau)$ spanned by M^τ . Let $M^\tau(u)$ denote the moat in \bar{F}^τ that contains terminal $u \in V(R)$ at time τ .

Crucial to PSF is the notion of *activity* of terminal pairs: For a terminal pair $(u, \bar{u}) \in R$, define the *death time* of (u, \bar{u}) as $\bar{d}(u, \bar{u}) = \frac{1}{2}d_G(u, \bar{u})$, where $d_G(u, \bar{u})$ is the cost of a shortest u, \bar{u} -path with respect to c in G . We call a terminal pair $(u, \bar{u}) \in R$ *active* at time τ if

$$\xi_{u\bar{u}}^\tau < \pi(u, \bar{u}) \quad \text{and} \quad \tau < \bar{d}(u, \bar{u}); \tag{38}$$

otherwise, (u, \bar{u}) is *inactive* at time τ . As time progresses, PSF increases (but never decreases) the cost shares of terminal pairs. Thus once a terminal pair becomes inactive it remains inactive. Define $\tau_{u\bar{u}}$ as the point of time at which terminal pair $(u, \bar{u}) \in R$ becomes inactive.

We say that a terminal $u \in V(R)$ is *active* at time τ if its pair (u, \bar{u}) is active at this time. Let \mathcal{A}^τ be the set of all terminals that are active at time τ . Finally, we call a moat M^τ of \bar{F}^τ *active* at time τ if it contains at least one active terminal, i.e., $V(M^\tau) \cap \mathcal{A}^\tau \neq \emptyset$.

At time τ our algorithm PSF increases the dual values of all cuts defined by moats M^τ in \bar{F}^τ which are active at time τ . These duals are increased simultaneously and by the same amount. Subsequently, we also say that we *grow* all active moats in \bar{F}^τ at time τ . Moreover, it is convenient to regard the growing of moats as being identical to increasing the respective duals.

The growth of an active moat M^τ is shared evenly among all active terminals in M^τ . More formally, we define the cost share $\xi_u^{\tau'}$ of a terminal $u \in V(R)$ at time τ' as follows:

$$\xi_u^{\tau'} = \int_0^{\tau'} \frac{1}{|V(M^\tau(u)) \cap \mathcal{A}^\tau|} d\tau \quad \forall \tau' \in [0, \tau_{u\bar{u}}] \tag{39}$$

and $\xi_u^{\tau'} = \xi_{u\bar{u}}^{\tau_{u\bar{u}}}$ for all $\tau' > \tau_{u\bar{u}}$. Moreover, we define $\xi_{u\bar{u}}^\tau = \xi_u^\tau + \xi_{\bar{u}}^\tau$ for every terminal pair $(u, \bar{u}) \in R$. Observe that the total contribution to the cost share of a terminal pair (u, \bar{u}) within ϵ time units is at most 2ϵ . Also, note that (u, \bar{u}) may receive cost share from a moat M^τ that contains u and \bar{u} .

We say that two active moats M_1^τ and M_2^τ *collide* at time τ if their nodes are contained in the same connected component of $\bar{F}^{\tau'}$ if and only if $\tau' \geq \tau$. In this case, PSF adds a cheapest collection of edges to F^τ such that the active nodes of M_1^τ and M_2^τ are in the same connected component of $F^{\tau'}$ for all $\tau' \geq \tau$. Note that this way of

extending the current forest F^τ ensures that only non-redundant edges are added to F^τ and thus F^τ remains a forest.⁴

Suppose a terminal pair $(u, \bar{u}) \in R$ becomes inactive at time $\tau = \tau_{u\bar{u}}$ because it reaches its penalty, i.e., $\xi_{u\bar{u}}^\tau = \pi(u, \bar{u})$. The algorithm then adds (u, \bar{u}) to Q^τ . Note that it might happen that (u, \bar{u}) is added to Q^τ even though u and \bar{u} are connected by F^τ .

PSF terminates at time τ^* when the last terminal pair becomes inactive. Let (F, Q) be the final solution computed by PSF. Subsequently, we use PSF to refer to the algorithm and denote the final cost shares by $\xi^{PSF} = (\xi_{u\bar{u}}^{\tau^*})_{(u,\bar{u}) \in R}$.

The following view of the moat-growing process will be helpful to prove some lemmas below: Consider an active moat M^τ that is grown by PSF at time τ . Then an increase by $\epsilon > 0$ of the dual value y_S of the induced cut $S = V(M^\tau)$ can be interpreted as M^τ loading all edges in $\delta(S)$ that cross the cut S by an amount of ϵ . Note that the algorithm ensures that the total load that an edge $e \in E$ receives from different moats does not exceed its cost $c(e)$. This way the cost $c(e)$ of an edge $e \in E$ is partitioned among the moats loading e and possibly some leftover piece of slack.

6.3 Some properties of PSF

The following fact follows immediately from definitions (38) and (39).

Fact 8 For every terminal pair $(u, \bar{u}) \in R$, $\xi_{u\bar{u}}^{PSF} \leq \min\{\pi(u, \bar{u}), 2d(u, \bar{u})\}$.

Because at any point of time the growth of all active moats is shared among active terminals, the following holds true.

Fact 9 For every time $\tau \geq 0$,

$$\sum_{S \subseteq V} y_S^\tau = \sum_{(u,\bar{u}) \in R} \xi_{u\bar{u}}^\tau.$$

Note that a terminal pair is added to Q^τ if and only if $\xi_{u\bar{u}}^\tau = \pi(u, \bar{u})$. Thus the following fact is immediate.

Fact 10 Let Q be the final set of terminal pairs computed by PSF. Then

$$\sum_{(u,\bar{u}) \in Q} \pi(u, \bar{u}) = \sum_{(u,\bar{u}) \in Q} \xi_{u\bar{u}}^{PSF}$$

Suppose a terminal pair (u, \bar{u}) becomes inactive at time $d(u, \bar{u})$. The next fact shows that (u, \bar{u}) must then be connected in F .

Fact 11 Let terminal pair (u, \bar{u}) become inactive at time $d(u, \bar{u})$. Then u and \bar{u} are connected in F .

⁴ Note that in PSF the decision of which edges are added to F^τ are delayed until these edges are non-redundant. This is different from other primal-dual approaches for network design problems (see, e.g., [16]) where this is ensured by a final reverse-delete step.

Proof Let $P_{u\bar{u}}$ be a shortest u, \bar{u} -path in G . Note that terminals u and \bar{u} are both active until time $d(u, \bar{u})$. Thus all edges of $P_{u\bar{u}}$ must be tight at some time $\tau \leq d(u, \bar{u})$, i.e., $P_{u\bar{u}} \subseteq \bar{F}^\tau$. Then either u and \bar{u} are already connected in F^τ or $P_{u\bar{u}}$ is added to F^τ . □

Observe that the last fact also establishes correctness of PSF: The final solution (F, Q) computed by PSF is a feasible solution for the given prize-collecting Steiner forest instance.

6.4 Cross-monotonicity

In order to show cross-monotonicity of ξ^{PSF} we compare the execution of PSF on terminal set R with the one on terminal set $R_{-st} = R \setminus \{(s, t)\}$ for any $(s, t) \in R$. We use \mathcal{G}_{-st} (where \mathcal{G} is a place holder for PSF, F, \bar{F}, M , etc.) to refer to \mathcal{G} in the run of PSF on R_{-st} . For notational convenience, throughout this section we denote by $\xi_{-st}^\tau(u, \bar{u})$ the cost share of (u, \bar{u}) in the run of PSF on R_{-st} and by $\xi(u, \bar{u})$ the respective cost share in PSF on R .

Our activity notion of terminal pairs defined in (38) turns out to be crucial to prove cross-monotonicity. As we will show below, it guarantees that if a terminal u is active at time τ in the run of PSF_{-st} on R_{-st} then it will also be active at time τ in the run of PSF on R . As a consequence, if the moat $M_{-st}^\tau(u)$ of u is active at time τ in PSF_{-st} (because u is active) then also the corresponding moat $M^\tau(u)$ is active at time τ in PSF and $M_{-st}^\tau(u) \subseteq M^\tau(u)$. Therefore, the set of edges \bar{F}_{-st}^τ that are tight at time τ in PSF_{-st} is a refinement of \bar{F}^τ . Further, this implies that the share that u receives from the growth of $M_{-st}^\tau(u)$ at time τ in PSF_{-st} is at least as large as the one it receives from the growth of $M^\tau(u)$ in PSF.

We formalize the above observations in the following lemma.

Lemma 16 *Consider the execution of PSF on R and R_{-st} , respectively. The following holds for every time $\tau \geq 0$:*

1. \bar{F}_{-st}^τ is a refinement of \bar{F}^τ , i.e., $\bar{F}_{-st}^\tau \subseteq \bar{F}^\tau$.
2. For all $(u, \bar{u}) \in R_{-st}$, $\xi_{-st}^\tau(u, \bar{u}) \geq \xi^\tau(u, \bar{u})$.

Proof We prove the lemma by induction over time τ . Clearly, the lemma holds at time $\tau = 0$. Suppose the lemma holds at time τ .

The only moats that may potentially violate the claim $\bar{F}_{-st}^{\tau+\epsilon} \subseteq \bar{F}^{\tau+\epsilon}$ at time $\tau + \epsilon$ for some small $\epsilon > 0$, are those that are active at time τ in PSF_{-st} . Let M_{-st} be a moat of \bar{F}_{-st}^τ that is active at time τ . By the induction hypothesis, there exists a moat M in \bar{F}^τ such that $M_{-st} \subseteq M$. We argue that M must be active at time τ in PSF.

Since M_{-st} is active at time τ , there exists a terminal $u \in V(M_{-st})$ such that $\pi(u, \bar{u}) - \xi_{-st}^\tau(u, \bar{u}) > 0$ and $\tau < d(u, \bar{u})$. By our induction hypothesis,

$$\pi(u, \bar{u}) - \xi^\tau(u, \bar{u}) \geq \pi(u, \bar{u}) - \xi_{-st}^\tau(u, \bar{u}) > 0.$$

Therefore, M must be active at time τ too.

As a consequence, every edge $e \in \delta(S)$ crossing the cut $S = V(M)$ induced by M is loaded at least as fast in PSF as in PSF_{-st} . Thus, if an edge $e \in \delta(S)$ gets tight at

time $\tau' \in (\tau, \tau + \epsilon]$ in PSF_{-st} then it also gets tight at time τ' in PSF . Therefore, the refinement property is preserved, which proves the first part of the lemma.

It remains to be shown that $\xi_{-st}^{\tau+\epsilon}(u, \bar{u}) \geq \xi^{\tau+\epsilon}(u, \bar{u})$ for all $(u, \bar{u}) \in R_{-st}$. Fix an arbitrary terminal pair $(u, \bar{u}) \in R_{-st}$. We consider two cases:

- (i) Assume (u, \bar{u}) is inactive at time τ in PSF_{-st} . If (u, \bar{u}) is also inactive at time τ in PSF the claim holds. Otherwise, (u, \bar{u}) is active at time τ in PSF . But then $\tau < d(u, \bar{u})$ and thus $\xi_{-st}^{\tau}(u, \bar{u}) = \pi(u, \bar{u}) \geq \xi^{\tau}(u, \bar{u})$. The claim holds.
- (ii) Suppose (u, \bar{u}) is active at time τ in PSF_{-st} . Let $M_{-st}^{\tau}(u)$ be the moat of u at time τ . By our induction hypothesis, $M_{-st}^{\tau}(u)$ is contained in the moat $M^{\tau}(u)$ of \bar{F}^{τ} in PSF . Moreover, from the discussion above we know that every terminal pair $(v, \bar{v}) \in R_{-st}$ that is active at time τ in PSF_{-st} must be active at time τ in PSF , i.e., $\mathcal{A}_{-st}^{\tau} \subseteq \mathcal{A}^{\tau}$. Therefore, $|V(M_{-st}^{\tau}(u)) \cap \mathcal{A}_{-st}^{\tau}| \leq |V(M^{\tau}(u)) \cap \mathcal{A}^{\tau}|$. Thus, the additional cost share that (u, \bar{u}) receives in the time interval $(\tau, \tau + \epsilon]$ in PSF_{-st} [see (39)] is at least as large as the one it receives in PSF . \square

6.5 Competitiveness

We next show that ξ^{PSF} satisfies competitiveness. Note that this is non-trivial here because the dual solution generated by PSF may be infeasible for (D). The proof of the following lemma is similar to the one presented in [26].

Lemma 17 *Let (F^*, Q^*) be an optimal solution to an instance of the prize-collecting Steiner forest problem with terminal pair set R . Then ξ^{PSF} satisfies competitiveness, i.e.,*

$$\sum_{(u, \bar{u}) \in R} \xi_{u\bar{u}}^{\text{PSF}} \leq c(F^*) + \pi(Q^*).$$

Proof Consider a separated terminal pair $(u, \bar{u}) \in Q^*$. By Fact 8, we have

$$\sum_{(u, \bar{u}) \in Q^*} \xi_{u\bar{u}}^{\text{PSF}} \leq \pi(Q^*).$$

It remains to be shown that the total cost share of all terminal pairs $(u, \bar{u}) \in R \setminus Q^*$ is bounded by $c(F^*)$.

Consider a tree T of F^* and let $R(T)$ be the set of terminal pairs that are connected by T . We prove that

$$\sum_{(u, \bar{u}) \in R(T)} \xi_{u\bar{u}}^{\text{PSF}} \leq c(T). \tag{40}$$

The lemma follows by summing over all trees T in F^* .

We define $\mathcal{M}^{\tau}(T)$ as the set of moats in \bar{F}^{τ} at time τ that contain at least one active terminal of $V(R(T))$, i.e.,

$$\mathcal{M}^{\tau}(T) = \{M^{\tau}(u) : u \in V(R(T)) \cap \mathcal{A}^{\tau}\}.$$

Among all terminal pairs in $R(T)$, let (w, \bar{w}) be a pair that is active longest. By our definition of activity in (38), all terminal pairs in $R(T)$ are inactive after time $\bar{d}(w, \bar{w})$. We show that the total growth of $\mathcal{M}^\tau(T)$ for all $\tau \in [0, \bar{d}(w, \bar{w})]$ is at most $c(T)$. This implies (40).

At any time τ , the moats in $\mathcal{M}^\tau(T)$ are disjoint. Moreover, T connects all terminals in $V(R(T))$. Thus, if there exists a moat $M^\tau \in \mathcal{M}^\tau(T)$ that intersects an edge of T , i.e., there is some $e \in T$ such that $e \in \delta(V(M^\tau))$, then each moat in $\mathcal{M}^\tau(T)$ must intersect an edge of T ; we say that the moats in $\mathcal{M}^\tau(T)$ load T . Moreover, each such moat M^τ loads a different part of T . Thus, the total growth of moats in $\mathcal{M}^\tau(T)$ for all τ at which $\mathcal{M}^\tau(T)$ loads T is at most $c(T)$.

Let $\tau_0 \leq \bar{d}(w, \bar{w})$ be the first point of time at which $\mathcal{M}^{\tau_0}(T)$ does not load T . If $\mathcal{M}^{\tau_0}(T) = \emptyset$, we are done. Otherwise, we must have that $\mathcal{M}^{\tau_0}(T) = \{M^{\tau_0}\}$ and $T \subseteq M^{\tau_0}$. The additional growth of M^τ for all times $\tau \in [\tau_0, \bar{d}(w, \bar{w})]$ is at most $\bar{d}(w, \bar{w}) - \tau_0$. Let $P_{w\bar{w}}$ be the unique w, \bar{w} -path in T . The additional growth of M^τ for all times $\tau \in [\tau_0, \bar{d}(w, \bar{w})]$ is at most $\bar{d}(w, \bar{w}) - \tau_0 \leq \frac{c(P_{w\bar{w}})}{2} \leq \frac{c(T)}{2}$. Combining this with the bound above gives an upper bound of $\frac{3}{2}c(T)$ on the total cost shares of pairs in $R(T)$.

We next refine the above argument to prove (40). Define $\mathcal{M}_{-w\bar{w}}^\tau \subseteq \mathcal{M}^\tau(T)$ as the set of active moats different from $M^\tau(w)$ and $M^\tau(\bar{w})$ that load $P_{w\bar{w}}$ at time $\tau < \tau_0$, i.e.,

$$\mathcal{M}_{-w\bar{w}}^\tau = \{M^\tau \in \mathcal{M}^\tau(T) \setminus \{M^\tau(w), M^\tau(\bar{w})\} : \delta(V(M^\tau)) \cap P_{w\bar{w}} \neq \emptyset\}.$$

The crucial insight is that each moat $M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau$ loads at least two edges of $P_{w\bar{w}}$ at all times $\tau \in [0, \tau_0)$. The basic idea is to use one part of this load on $P_{w\bar{w}}$ to compensate for the additional growth during the time interval $[\tau_0, \bar{d}(w, \bar{w})]$. We make this idea more precise.

Define the *degree* $\text{deg}(M^\tau)$ of a moat $M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau$ as

$$\text{deg}(M^\tau) = |\delta(V(M^\tau)) \cap P_{w\bar{w}}|.$$

Proposition 1 Consider a time $\tau < \tau_0$ and a moat $M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau$. Then $\text{deg}(M^\tau) \geq 2$.

Proof Both $M^\tau(w)$ and $M^\tau(\bar{w})$ are active at time $\tau < \tau_0$ and thus $\{M^\tau(w), M^\tau(\bar{w})\} \subseteq \mathcal{M}^\tau(T)$ (possibly $M^\tau(w) = M^\tau(\bar{w})$). By definition of $\mathcal{M}_{-w\bar{w}}^\tau$, $M^\tau \in \mathcal{M}^\tau(T)$ and $M^\tau \notin \{M^\tau(w), M^\tau(\bar{w})\}$. Furthermore, M^τ is disjoint from all other moats in $\mathcal{M}^\tau(T)$. Suppose $|M^\tau \cap P_{w\bar{w}}| = 1$. But then, moat M^τ must contain w or \bar{w} . This contradicts the disjointness of M^τ and $\{M^\tau(w), M^\tau(\bar{w})\}$. \square

By our choice of $(w, \bar{w}) \in R(T)$ as the terminal pair with largest activity time and by our assumption that $\mathcal{M}^{\tau_0}(T) \neq \emptyset$ it follows that both, $M^\tau(w)$ and $M^\tau(\bar{w})$ are active for all $0 \leq \tau \leq \tau_0$. We define $l_{w\bar{w}}$ as the total dual growth of the moats containing w and \bar{w} up to time τ_0 . Formally, let

$$\delta_{w\bar{w}}^\tau = \begin{cases} 2 & \text{if } M^\tau(w) \neq M^\tau(\bar{w}) \\ 1 & \text{otherwise} \end{cases}$$

and

$$l_{w\bar{w}} = \int_0^{\tau_0} \delta_{w\bar{w}}^\tau d\tau.$$

It follows that the cost $c(P_{w\bar{w}})$ of path $P_{w\bar{w}}$ is at least

$$l_{w\bar{w}} + \int_0^{\tau_0} \sum_{M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau} \text{deg}(M^\tau) d\tau.$$

We let $\text{slack}_{w\bar{w}}$ be the difference between $c(P_{w\bar{w}})$ and the above term and obtain

$$c(P_{w\bar{w}}) = l_{w\bar{w}} + \text{slack}_{w\bar{w}} + \int_0^{\tau_0} \sum_{M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau} \text{deg}(M^\tau) d\tau. \tag{41}$$

We define the total growth $y^{\tau_0}(T)$ produced by terminal pairs in $R(T)$ until time τ_0 as follows:

$$y^{\tau_0}(T) = \int_0^{\tau_0} |\mathcal{M}^\tau(T)| d\tau.$$

At all times $\tau \leq \tau_0$, each moat in $\mathcal{M}^\tau(T)$ loads at least one distinct edge of T ; those in $\mathcal{M}_{-w\bar{w}}^\tau$ load at least two edges of T . Thus, we have

$$c(T) \geq y^{\tau_0}(T) + \text{slack}_{w\bar{w}} + \int_0^{\tau_0} \sum_{M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau} (\text{deg}(M^\tau) - 1) d\tau. \tag{42}$$

The additional growth between time τ_0 and $d(w, \bar{w})$ is at most $d(w, \bar{w}) - \tau_0$. Using that $d(w, \bar{w}) \leq c(P_{w\bar{w}})/2$ and (41), we obtain

$$\begin{aligned} d(w, \bar{w}) - \tau_0 &\leq \frac{l_{w\bar{w}}}{2} - \tau_0 + \frac{\text{slack}_{w\bar{w}}}{2} + \int_0^{\tau_0} \sum_{M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau} \frac{\text{deg}(M^\tau)}{2} d\tau \\ &\leq \frac{\text{slack}_{w\bar{w}}}{2} + \int_0^{\tau_0} \sum_{M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau} (\text{deg}(M^\tau) - 1) d\tau, \end{aligned} \tag{43}$$

where we exploit that $\text{deg}(M^\tau) \geq 2$ for all $M^\tau \in \mathcal{M}_{-w\bar{w}}^\tau$ and the fact that $l_{w\bar{w}} \leq 2\tau_0$. Combining (42) and (43) proves that the total growth $y^{\tau_0}(T) + d(w, \bar{w}) - \tau_0$ is at most $c(T)$, which concludes the proof. \square

6.6 Cost recovery

Consider a tree T which is part of the forest F^τ . Let $R(T)$ be the set of terminal pairs that are spanned by T . We call T *active* at time τ if it contains a terminal that is active at this time, i.e., $V(R(T)) \cap \mathcal{A}^\tau \neq \emptyset$; otherwise, T is said to be *inactive*. Define the *age* of T as the last point of time $\tau_0 \in [0, \tau]$ when T was still active; more formally,

$$\text{age}^\tau(T) = \max\{\tau_0 \in [0, \tau] : V(R(T)) \cap \mathcal{A}^{\tau_0} \neq \emptyset\}.$$

Note that $\text{age}^\tau(T) = \tau$ if T is active at time τ .

The proof of the following lemma borrows ideas from the proof of Lemma 5.3 in [1].

Lemma 18 *For every time $\tau \geq 0$ and every tree T of F^τ we have*

$$c(T) \leq 2 \sum_{u \in V(R(T))} \xi_u^\tau - 2 \cdot \text{age}^\tau(T). \tag{44}$$

Proof The proof is by induction over time τ . Clearly, the claim holds at time $\tau = 0$. Suppose it holds at time τ . We show that the claim remains true at time $\tau' = \tau + \epsilon$ for some $\epsilon > 0$.

It is not hard to verify that the claim holds at all times $\tau' > \tau$ for which $F^{\tau'} = F^\tau$: Consider a tree T of F^τ . Then the left-hand side of (44) remains the same. If T is inactive at time τ then the right-hand side of (44) also remains unchanged. Otherwise, T is active at time τ . But then there is a moat M^τ in \bar{F}^τ that contains T and whose growth is shared among all active terminals in $V(R(T))$. Also, the age of T increases by the same amount. We conclude that

$$c(T) \leq 2 \sum_{u \in V(R(T))} \xi_u^\tau - 2 \cdot \text{age}^\tau(T) = 2 \sum_{u \in V(R(T))} \xi_u^{\tau'} - 2 \cdot \text{age}^{\tau'}(T).$$

Next consider the first point of time $\tau' > \tau$ for which $F^{\tau'} \supset F^\tau$. Because edges are only added to the current forest, there is a tree T' that becomes part of $F^{\tau'}$ because two active moats, say $M_1^{\tau'}$ and $M_2^{\tau'}$, collide at time τ' . Then T' is the union of the edges in $D = F^{\tau'} \setminus F^\tau$ and r inactive trees T_1, \dots, T_r of F^τ (possibly $r = 0$). Thus,

$$\begin{aligned} c(T') &= c(D) + \sum_{i=1}^r c(T_i) \\ &\leq c(D) + \sum_{i=1}^r \left(2 \sum_{u \in V(R(T_i))} \xi_u^\tau - 2 \cdot \text{age}^\tau(T_i) \right) \\ &= c(D) + \sum_{i=1}^r \left(2 \sum_{u \in V(R(T_i))} \xi_u^{\tau'} - 2 \cdot \text{age}^{\tau'}(T_i) \right), \end{aligned} \tag{45}$$

where the first inequality holds because of the induction hypothesis and the last equality follows from the discussion above. It is not hard to see that the total cost of adding the edges in D to connect the active terminals in $M_1^{\tau'}$ and $M_2^{\tau'}$ through T_1, \dots, T_r is at most

$$2\tau' + 2 \sum_{i=1}^r \text{age}^{\tau'}(T_i). \tag{46}$$

Combining (45) and (46), we obtain

$$c(T') \leq 2\tau' + 2 \sum_{i=1}^r \sum_{u \in V(R(T_i))} \xi_u^{\tau'}. \tag{47}$$

Finally, observe that the sum of the cost shares of all terminals in $M_1^{\tau'}$ and $M_2^{\tau'}$ is at least $2\tau'$. Thus, twice the cost shares of all these terminals accounts for $4\tau'$. Further, note that $\text{age}^{\tau'}(T') = \tau'$. Combining these observations with (47), we conclude

$$c(T') \leq 2 \sum_{u \in V(R(T'))} \xi_u^{\tau'} - 2\tau' = 2 \sum_{u \in V(R(T'))} \xi_u^{\tau'} - 2 \cdot \text{age}^{\tau'}(T'),$$

which proves the claim. □

Lemma 19 *Let (F, Q) be the solution computed by PSF on terminal pair set R . Then*

$$c(F) + \pi(Q) \leq 3 \sum_{(u, \bar{u}) \in R} \xi_{u\bar{u}}^{\text{PSF}}.$$

Proof By Fact 10 we have

$$\pi(Q) = \sum_{(u, \bar{u}) \in Q} \xi_{u\bar{u}}^{\text{PSF}}.$$

Moreover, by Lemma 18 the cost of the final forest satisfies

$$c(F) \leq 2 \sum_{(u, \bar{u}) \in R} \xi_{u\bar{u}}^{\text{PSF}}.$$

Combining these bounds proves the claim. □

6.7 Social cost approximation

Let ξ^{SF} be the cost-sharing method by Könemann et al. [26] for the Steiner forest problem. The only difference between our algorithm PSF and the algorithm SF in [26] is the definition of the activity notion: In [26] a terminal pair (u, \bar{u}) is *active* at time τ if and only if $\tau < \text{d}(u, \bar{u})$. This definition coincides with the one in (38) if $\pi(u, \bar{u}) = \infty$ for every $(u, \bar{u}) \in R$.

We argue that ξ^{SF} and ξ^{PSF} satisfy the two properties of our Lifting Theorem (Theorem 4). Property 1 is satisfied by Fact 8. The following lemma shows that Property 2 is fulfilled as well.

Lemma 20 *Define τ_0 as the first point of time τ at which $\xi_{v\bar{v}}^{\tau,PSF} = \pi(v, \bar{v})$ for some terminal pair $(v, \bar{v}) \in R$; let $\tau_0 = \infty$ if no such time exists. Then for every terminal pair $(u, \bar{u}) \in R$ and all $\tau \in [0, \tau_0)$, $\xi_{u\bar{u}}^{\tau,PSF} = \xi_{u\bar{u}}^{\tau,SF}$.*

Proof It is sufficient to show that for all $\tau \in [0, \tau_0)$ and every terminal pair $(u, \bar{u}) \in R$ it holds that (u, \bar{u}) is active at time τ in PSF if and only if (u, \bar{u}) is active at time τ in SF.

A necessary condition for (u, \bar{u}) being active at time τ in PSF is that $\tau < d(u, \bar{u})$. Thus, (u, \bar{u}) is active at time τ in SF if (u, \bar{u}) is active at this time in PSF. Next, suppose (u, \bar{u}) is active at time τ in SF and thus $\tau < d(u, \bar{u})$. Since $\tau < \tau_0$, we have $\xi_{w\bar{w}}^{\tau,PSF} < \pi(w, \bar{w})$ for all $(w, \bar{w}) \in R$; in particular this also holds for (u, \bar{u}) . Thus, (u, \bar{u}) is active at time τ in PSF. □

Recall that the Moulin mechanism $M(\xi^{SF})$ is $O(\log^2 k)$ -approximate [11]. By our Lifting Theorem and the fact that ξ^{PSF} is 3-budget balanced, we conclude that $M(\xi^{PSF})$ is $O(\log^2 k)$ -approximate.

Corollary 5 *The Moulin mechanism $M(\xi^{PSF})$ is $O(\log^2 k)$ -approximate.*

7 Summability of Steiner forest cost sharing method

In this section, we show that the cost-sharing method ξ^{SF} of Könemann et al. [26] for the Steiner forest problem is $O(\log^3 k)$ -summable.

Theorem 12 *The cost sharing method ξ^{SF} is $O(\log^3 k)$ -summable.*

This result is inferior to the bound of $O(\log^2 k)$ obtained by Chawla et al. [11]. However, the proof presented in this section is simpler than the one in [11] and adds a novel methodological contribution by showing that such a result can also be proved by embedding the graph distances into random *hierarchically separated trees (HST)* [5, 13].

As mentioned earlier, the algorithm SF that computes ξ^{SF} works exactly the same way as our algorithm PSF described in the previous section if we set all penalties to ∞ . We drop the superscript SF in the discussion below.

Suppose we are given an arbitrary subset $S \subseteq U$ and an ordering σ . We order the terminal pairs in S according to σ and without loss of generality assume that they are labeled such that

$$S = \{(s_1, t_1), \dots, (s_l, t_l)\}, \quad \text{where } l = |S|.$$

Let $S_i \subseteq S$ be the set of the first i terminal pairs of S . We use $\xi_i(S_i)$ to refer to the cost share of terminal pair (s_i, t_i) , $i \in [l]$, computed by SF when run on terminal pair set S_i . We need to prove that

$$\sum_{i=1}^l \xi_i(S_i) = O(\log^3 k) \cdot C(S), \tag{48}$$

where $C(S)$ is the cost of an optimal Steiner forest cost for terminal set S .

We assume that the distance between every two nodes in G is at least 1, i.e., $d_G(u, v) \geq 1$ for all $u, v \in U$. This assumption is without loss of generality as we may scale the edge costs appropriately.

Recall that in SF each terminal pair $(s_i, t_i) \in U$ has a *death time* $\bar{d}(s_i, t_i)$ which is defined as half the distance between s_i and t_i in G . We partition terminal pairs in S into classes, depending on their death times: A terminal pair $(s_i, t_i) \in S$ is of *class* $r \geq 0$ if $\bar{d}(s_i, t_i) \in (2^{r-1}, 2^r]$. Let $r(i)$ be the class to which terminal pair (s_i, t_i) belongs. We use S^r to refer to the (ordered) set of terminal pairs in S that belong to class r . Moreover, we define $S_i^r \subseteq S_i$ to be the set of class r terminal pairs in S_i , i.e., $S_i^r = S_i \cap S^r$ for every $i \in [l]$. Let Δ_S be the maximum death time among all terminal pairs in S . Clearly, there are at most $\log(\Delta_S) + 1$ classes.

Since ξ is cross-monotonic, we have for every $(s_i, t_i), i \in [l], \xi_i(S_i) \leq \xi_i(S_i^{r(i)})$. Thus,

$$\sum_{i=1}^l \xi_i(S_i) \leq \sum_{i=1}^l \xi_i(S_i^{r(i)}) = \sum_{r=0}^{\log(\Delta_S)+1} \sum_{(s_i, t_i) \in S^r} \xi_i(S_i^r). \tag{49}$$

We first consider all terminal pairs of classes $0, 1, \dots, \log(\Delta_S/k) + 1$. Note that every such terminal pair has death time at most $2\Delta_S/k$. The cost share of a terminal pair is at most twice its death time and thus

$$\sum_{r=0}^{\log(\Delta_S/k)+1} \sum_{(s_i, t_i) \in S^r} \xi_i(S_i^r) \leq k \cdot \frac{4\Delta_S}{k} \leq 4\Delta_S \leq 2C(S). \tag{50}$$

That is, all terminal pairs of class at most $\log(\Delta_S/k) + 1$ contribute at most $2C(S)$ to the left-hand side of (48). We can therefore concentrate on terminal pairs in classes $\log(\Delta_S/k) + 2, \dots, \log(\Delta_S) + 1$. Note that these are at most $\log k$ different classes. For each class $r > \log(\Delta_S/k) + 1$, we prove

$$\sum_{(s_i, t_i) \in S^r} \xi_i(S_i^r) = O(\log^2(|S^r|) \cdot C(S)) = O(\log^2 k) \cdot C(S).$$

This together with (49) and (50) proves (48). The next lemma states that for each class, we can assume that all death times are rounded up to the nearest power of 2.

Lemma 21 (Rounding Lemma) *Fix some r and suppose we set all death times of terminal pairs in S^r to 2^r . Let $\tilde{\xi}$ be the cost shares computed by SF with these modified death times. Then*

$$\sum_{(s_i, t_i) \in S^r} \xi_i(S_i^r) \leq 3 \sum_{(s_i, t_i) \in S^r} \tilde{\xi}_i(S_i^r)$$

The proof of Lemma 21 is deferred to the end of this section.

7.1 Summability of SF with identical death times

We next show that the cost shares of SF are $O(\log^2 k)$ -summable if all death times are equal. Eventually, we apply the result presented in this section together with the Rounding Lemma to each class $r > \log(\Delta_S/k) + 1$ separately. For notational convenience, we use S instead of S^r here.

Suppose that the death time of all terminal pairs in S is v , i.e., $d(s_i, t_i) = v$ for all $i \in [l]$; as before, we define $l = |S|$. Let F^* be a minimum cost Steiner forest for terminal pair set S . For a tree $T \in F^*$, let $S(T)$ be the set of terminal pairs in S that are spanned by T . Consider a terminal pair (s_i, t_i) , $1 \leq i \leq l$, of S and let $T \in F^*$ be the tree that contains s_i, t_i , i.e., $(s_i, t_i) \in S(T)$. Define $S_i(T)$ as the set of terminal pairs in S that precede (s_i, t_i) (with respect to σ) and are also part of T ; more precisely $S_i(T) = S_i \cap S(T)$. Run SF on $S_i(T)$ and let $\xi_i(S_i(T))$ be the respective cost share of (s_i, t_i) . As $S_i(T) \subseteq S_i$ and the cost shares computed by SF are cross-monotonic, we have

$$\xi_i(S_i(T)) \geq \xi_i(S_i). \tag{51}$$

We prove that for each tree $T \in F^*$, we have

$$\sum_{(s_i, t_i) \in S(T)} \xi_i(S_i(T)) = O(\log^2(|S(T)|) \cdot c(T)). \tag{52}$$

Summing over all trees $T \in F^*$ together with (51) then shows that

$$\sum_{i=1}^l \xi_i(S_i) = O(\log^2 k) \cdot C(S).$$

Given tree T , we construct a rooted tree $T' = (V', E')$, also called *Shapley tree* in the following, and a non-negative length function $\ell : E' \rightarrow \mathbb{R}^+$ on the edges of T' . We use $T'(e)$ to refer to the subtree of T' below edge $e \in E'$. Moreover, for a node $u \in V'$ let P_{ur} be the unique u, r -path from u to the root r of T' . We construct T' such that the following four *Shapley tree properties* hold:

- (ST1) The leaves of T' are the terminals in $S(T)$.
- (ST2) For every two terminals that are contained in the subtree $T'(e)$ for some $e \in E'$, their distance in G is at most $\ell(e)$, i.e., $d_G(u, v) \leq \ell(e)$ for all $u, v \in S(T) \cap T'(e)$.
- (ST3) For every path $P_{ur} = (e_1, \dots, e_m)$ from terminal $u \in S(T)$ to the root r of T' , we have
 1. $\ell(e_1) = 1$,
 2. $\ell(e_j) = 2\ell(e_{j-1})$ for all $1 < j \leq m$, and
 3. $\ell(e_m) \geq v$.
- (ST4) The total length of T' is at most $O(\log |S(T)|)$ times the total cost of T , i.e., $\ell(T') = O(\log(|S(T)|) \cdot c(T))$.

We use tree T' to define *Shapley cost shares* for all terminals in $S(T)$: Let $T'[S_i(T)]$ be the induced subtree of T' on terminals pair set $S_i(T)$. For a terminal pair $(s_i, t_i) \in$

$S(T)$, we define $\xi'_i(S_i(T))$ to be the sum of the respective Shapley cost shares of terminals s_i and t_i in $T'[S_i(T)]$.

Lemma 22 *Let T' be the Shapley tree of T and let ξ' be the respective Shapley cost shares. Then*

$$\sum_{(s_i, t_i) \in S(T)} \xi'_i(S_i(T)) \leq H_{|S(T)|} \cdot \ell(T').$$

Proof As $T'[S_1(T)] \subseteq T'[S_2(T)] \subseteq \dots \subseteq T'[S_l(T)]$, the cost share contribution of an edge $e \in E'$ to the left-hand side of the inequality is at most $H_{|S(T)|} \cdot \ell(e)$. Summing over all edges $e \in E'$ of tree T' proves the lemma. \square

We next show that the cost share $\xi_i(S_i(T))$ of terminal pair (s_i, t_i) is upper bounded by its corresponding Shapley cost share $\xi'_i(S_i(T))$ in $T'[S_i(T)]$. This together with Lemma 22 and Property (ST4) establishes (52).

Lemma 23 *The cost share $\xi_i(S_i(T))$ of terminal pair $(s_i, t_i) \in S(T)$ is at most its Shapley cost share $\xi'_i(S_i(T))$.*

Proof All terminals in $S(T)$ are active until time v . The cost share $\xi_u(S_i(T))$ of a terminal $u \in \{s_i, t_i\}$ in SF is then defined as

$$\xi_u(S_i(T)) = \int_{\tau=0}^v \frac{d\tau}{a_i^\tau(u)}$$

where $a_i^\tau(u)$ is the number of active terminals in u 's moat at time τ in the run of SF($S_i(T)$). We bound the cost share that $u = s_i$ receives in SF($S_i(T)$) by its Shapley cost share. An analogous argument holds for $u = t_i$.

Consider the induced subtree $T'_i = T'[S_i(T)]$ on $S_i(T)$. Let $P_{ur} = (e_1, \dots, e_m)$ be the unique u, r -path in T'_i . Consider an edge $e_j, 1 < j \leq m$ and let $T'_i(e_j)$ be the subtree of T'_i below edge e_j . We use $z_i(e_j)$ to refer to the number of terminals in $T'_i(e_j)$; define $z_i(e_1) = 1$. The Shapley cost share that u received for edge e_j is $\ell(e_j)/z_i(e_j)$. Thus,

$$\xi'_u(S_i(T)) = \sum_{j=1}^m \frac{\ell(e_j)}{z_i(e_j)}.$$

Let x be any terminal in $T'_i(e_j)$. By Property (ST2) $d_G(u, x) \leq \ell(e_j)$. Since both x and u are active until time v , their respective moats in SF($S_i(T)$) must have met by time at most $d_G(u, x)/2 \leq \ell(e_j)/2 = \ell(e_{j-1})$. Thus, $a_i^\tau(u) \geq z_i(e_j)$ for all $\tau \geq \ell(e_{j-1})$ for all $1 < j \leq m$.

Note that the cost share that u receives up to time 1 is at most 1. As $\ell(e_1) = 1$ and $\ell(e_m) \geq v$, we can write

$$\begin{aligned} \xi_u(S_i(T)) &= \int_{\tau=0}^v \frac{d\tau}{a_i^\tau(u)} \leq 1 + \sum_{j=2}^m \int_{\tau=\ell(e_{j-1})}^{\ell(e_j)} \frac{d\tau}{a_i^\tau(u)} \\ &\leq 1 + \sum_{j=2}^m \int_{\tau=\ell(e_{j-1})}^{\ell(e_j)} \frac{d\tau}{z_i(e_j)} = 1 + \sum_{j=2}^m \frac{\ell(e_{j-1})}{z_i(e_j)} \leq \xi'_u(S_i(T)). \end{aligned}$$

□

7.2 Tree construction

There are several ways to obtain a tree T' that satisfies Properties (ST1)–(ST4). For example, the HSTs construction given by Fakcharoenphol et al. [13] satisfies all Properties (ST1)–(ST3) and Property (ST4) on expectation.

Alternatively, using ideas similar to the one presented in [36], we may insert terminals one-by-one and obtain a tree T' whose nodes are terminals in $S(T)$ and that satisfies Properties (ST2), (ST3(b)), (ST3(c)) and (ST4). In order to achieve Property (ST1) and (ST3(a)), we simply replace each non-leaf terminal u with parent edge e in T' by a path (e_1, \dots, e_m) with $\ell(e_m) = \ell(e)/2$ and $\ell(e_1) = 1$. Clearly, this construction will add an additional cost of at most $\ell(T')$.

7.3 Rounding lemma

Consider the set $S = S^r$ of class r terminals and let $l = |S|$. As before we assume that S is ordered according to σ and S_i refers to the set of the first i terminal pairs of S . For a terminal $u \in \{s_i, t_i\}$, we also use S_u to refer to the corresponding set of terminal pairs S_i . Define $\mu = 2^{r-1}$, i.e., $d(s_i, t_i) \in (\mu, 2\mu]$ for all $1 \leq i \leq l$.

Recall that in SF a terminal $u \in \{s_i, t_i\}$ is called *active* at time τ if $\tau \leq d(s_i, t_i)$; it is said to be *inactive* otherwise. A terminal receives cost share only if it is active. For a terminal u that is active at time τ in $\text{SF}(S)$, define $a_u^\tau(S)$ as the number of active terminals in u 's moat. The cost share that an active terminal u receives at time τ is defined as $\xi_u^\tau(S) = 1/a_u^\tau(S)$. The cost share $\xi_{s_i t_i}^\tau(S)$ of terminal pair (s_i, t_i) at time τ is defined as $\xi_{s_i}^\tau(S) + \xi_{t_i}^\tau(S)$.

Fix a point of time $\tau \in (\mu, 2\mu]$. Without loss of generality, let $\xi_{s_i}^\tau(S_i) \geq \xi_{t_i}^\tau(S_i)$ for every terminal pair (s_i, t_i) , $1 \leq i \leq l$. We say s_i is the *dominating* terminal of (s_i, t_i) . Note that $\xi_{s_i t_i}^\tau(S_i) \leq 2\xi_{s_i}^\tau(S_i)$. Let D^τ be the set of all dominating terminals that are active at time τ . The following technical lemma is the key to proving Lemma 21. It shows that for every terminal $s_i \in D^\tau$ the cost share $\xi_{s_i}^\tau(S_i)$ that s_i receives at time τ can be charged to the cost share that some terminal $f^\tau(s_i)$ in S_i received at time $\tau - \mu$. Moreover, the mapping f^τ is injective. This will enable us to charge the total cost share collected by terminals in D^τ at time τ in $\text{SF}(S_i)$ to the total cost share of terminals in S_i at time $\tau - \mu$.

Lemma 24 *Let D^τ be the set of all dominating terminals that are active at time $\tau \in (\mu, 2\mu]$. There exists a mapping $f^\tau : D^\tau \rightarrow S$ such that the following conditions hold:*

1. For each $s_i \in D^\tau$ we have $\xi_{s_i}^\tau(S_i) \leq \xi_{f^\tau(s_i)}^{\tau-\mu}(S_{f^\tau(s_i)})$.
2. For all $s_i, s_j \in D^\tau, i \neq j$, we have $f^\tau(s_i) \neq f^\tau(s_j)$.

Proof We use $M_u^\tau(S)$ to refer to the moat of u at time τ in the run of SF on terminal set $S \subseteq R$. Subsequently, we exploit the following two properties of SF which can easily be proven given the results in Sect. 6.

Fact 13 Let $S \subseteq R$ and consider a terminal $u \in S$. For every $\tau' \leq \tau$ we have $M_u^{\tau'}(S) \subseteq M_u^\tau(S)$.

Fact 14 Let $S' \subseteq S \subseteq R$ and consider a terminal $u \in S'$. For every τ we have $M_u^\tau(S') \subseteq M_u^\tau(S)$.

We assume that the set of dominating terminals D^τ is ordered according to σ . We define f^τ inductively. Suppose f^τ satisfies Conditions 1 and 2 of the lemma for the first $n - 1$ terminals in D^τ . (Let f^τ be the empty mapping for $n = 0$). We define $f^\tau(s_i)$ of the n th terminal s_i of D^τ while maintaining Conditions 1 and 2.

Assume $\xi_{s_i}^\tau(S_i) = 1/x$. Let $C_{s_i} = M_{s_i}^\tau(S_i)$ be the set of all terminals that are contained in s_i 's moat at time τ . Note that $|C_{s_i}| \geq x$. Order the set C_{s_i} according to σ and delete all terminals except the first x ones. We call the resulting terminal set C_{s_i} the candidate set of s_i . Note that $C_{s_i} \subseteq M_{s_i}^\tau(S_i)$. We will eventually define $f(s_i) = \hat{u}$ for some $\hat{u} \in C_{s_i}$.

Consider the m th terminal u of $C_{s_i}, 1 \leq m \leq x$. Note that all terminals in $M_u^{\tau-\mu}(S_u)$ are active at time $\tau - \mu$ because all terminal death times are larger than μ . By Facts 13 and 14 we have $M_u^{\tau-\mu}(S_u) \subseteq M_u^\tau(S_u) \subseteq M_u^\tau(S_i) = M_{s_i}^\tau(S_i)$. Therefore, the moat $M_u^{\tau-\mu}(S_u)$ contains at most m terminals. Since $m \leq x$, we have $\xi_u^{\tau-\mu}(S_u) \geq 1/x$ for all $u \in C_{s_i}$.

Next we show that there always exists a choice of a terminal $\hat{u} \in C_{s_i}$ such that $f^\tau(s_j) \neq \hat{u}$ for all $s_j \in D^\tau, j < i$. The proof is by contradiction. Suppose that for each terminal $u \in C_{s_i}$ there exists a terminal $s_j \in D^\tau, j < i$, with $f^\tau(s_j) = u$. Note that by our induction hypothesis, $f^\tau(s_j) \neq f^\tau(s_k)$ for all $j \neq k$ and $j, k < i$. Consider some $u \in C_{s_i}$ and let $s_j \in D^\tau, j < i$, with $f^\tau(s_j) = u$. By our construction of the candidate set, we have $u = f^\tau(s_j) \in C_{s_j} \subseteq M_{s_j}^\tau(S_j)$. Moreover, $M_{s_j}^\tau(S_j) \subseteq M_{s_i}^\tau(S_i)$ by Fact 14. This implies that both $M_{s_j}^\tau(S_j)$ and $M_{s_i}^\tau(S_i)$ contain u and therefore must be identical. As a consequence, s_j is an active terminal of $M_{s_i}^\tau(S_i)$ (recall that s_j is active because $s_j \in D^\tau$). Because this holds for every $u \in C_{s_i}$, this leads to a contradiction to the assumption that $\xi_{s_i}^\tau(S_i) = 1/x$ since we have identified $|C_{s_i}| \geq x$ active terminals in $M_{s_i}^\tau(S_i)$ that are different from s_i . \square

We can then proof Lemma 21:

Proof (Lemma 21) First observe that the executions of SF with and without rounded death times are identical until time μ . Thus

$$\sum_{i=1}^l \int_{\tau=0}^{\mu} \xi_{s_i t_i}^\tau(S_i) d\tau = \sum_{i=1}^l \int_{\tau=0}^{\mu} \tilde{\xi}_{s_i t_i}^\tau(S_i) d\tau \leq \sum_{i=1}^l \tilde{\xi}_{s_i t_i}(S_i). \tag{53}$$

For time $\tau \in (\mu, 2\mu]$ let f^τ be a mapping as constructed in Lemma 24. Since s_i is the dominating terminal of (s_i, t_i) , we have

$$\begin{aligned} \sum_{i=1}^l \xi_{s_i t_i}^\tau(S_i) d\tau &\leq 2 \sum_{i=1}^l \xi_{s_i}^\tau(S_i) d\tau \leq 2 \sum_{i=1}^l \xi_{f^\tau(s_i)}^{\tau-\mu}(S_{f^\tau(s_i)}) d\tau \\ &\leq 2 \sum_{i=1}^l \xi_{s_i t_i}^{\tau-\mu}(S_i) d\tau, \end{aligned}$$

where we used Condition 1 and 2 of Lemma 24 for the second and last inequality, respectively. Integrating over all time instants in $(\mu, 2\mu]$, we obtain

$$\sum_{i=1}^l \int_{\tau=\mu}^{2\mu} \xi_{s_i t_i}^\tau(S_i) d\tau \leq 2 \sum_{i=1}^l \int_{\tau=0}^{\mu} \xi_{s_i t_i}^\tau(S_i) d\tau \stackrel{(53)}{\leq} 2 \sum_{i=1}^l \tilde{\xi}_{s_i t_i}(S_i).$$

□

8 Conclusions

In this paper we presented a general lifting technique to establish social cost approximation guarantees for prize-collecting problems. We applied our technique to derive Moulin mechanisms for three optimization problems. Our mechanisms are basically best possible both with respect to the budget balance and social cost approximation guarantee. It would be interesting to see further examples where our technique can be used to derive optimal mechanisms for prize-collecting problems. Moreover, it would be interesting to see whether our lifting technique can be applied to cost-sharing methods which are not defined through primal-dual algorithms.

Acknowledgments We thank anonymous reviewers of Mathematical Programming for their valuable comments and most helpful feedback.

References

1. Agrawal, A., Klein, P., Ravi, R.: When trees collide: an approximation algorithm for the generalized Steiner problem on networks. *SIAM J. Comput.* **24**(3), 440–456 (1995)
2. Archer, A., Bateni, M., Hajiaghayi, M., Karloff, H.: Improved approximation algorithms for prize-collecting steiner tree and TSP. *SIAM J. Comput.* **40**(2), 309–332 (2011)
3. Archer, A., Feigenbaum, J., Krishnamurthy, A., Sami, R., Shenker, S.: Approximation and collusion in multicast cost sharing. *Games Econ. Behav.* **47**(1), 36–71 (2004)
4. Arora, S., Lund, C., Motwani, R., Sudan, M., Szegedy, M.: Proof verification and the hardness of approximation problems. *J. ACM* **45**(3), 501–555 (1998)
5. Bartal, Y.: Probabilistic approximations of metric spaces and its algorithmic applications. In: *Proceedings of the 37th Symposium on the Foundations of Computer, Science*, pp. 184–193 (1996)
6. Bern, M., Plassman, P.: The Steiner problem with edge lengths 1 and 2. *Inf. Process. Lett.* **32**, 171–176 (1989)

7. Bienstock, D., Goemans, M.X., Simchi-Levi, D., Williamson, D.P.: A note on the prize collecting traveling salesman problem. *Math. Program.* **59**, 413–420 (1993)
8. Bleischwitz, Y., Monien, B.: Fair cost-sharing methods for scheduling jobs on parallel machines. *J. Discrete Algorithms* **7**(3), 280–290 (2009)
9. Bleischwitz, Y., Monien, B., Schopmann, F.: To be or not to be (served). In: *Proceedings of the 3rd International Conference on Internet and Network Economics*, pp. 515–528 (2007)
10. Brenner, J., Schäfer, G.: Cost sharing methods for makespan and completion time scheduling. In: *Proceedings of the 24th International Symposium on Theoretical Aspects of Computer, Science*, pp. 670–681 (2007)
11. Chawla, S., Roughgarden, T., Sundararajan, M.: Optimal cost-sharing mechanisms for steiner forest problems. In: *Proceedings of the 2nd International Workshop on Internet and Network Economics*, pp. 112–123 (2006)
12. Dutta, B., Ray, D.: A concept of egalitarianism under participation constraints. *Econometrica* **57**(3), 615–635 (1989)
13. Fakcharoenphol, J., Rao, S., Talwar, K.: A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.* **69**(3), 485–497 (2004)
14. Feigenbaum, J., Krishnamurthy, A., Sami, R., Shenker, S.: Hardness results for multicast cost-sharing. *Theor. Comput. Sci.* **304**, 215–236 (2003)
15. Feigenbaum, J., Papadimitriou, C.H., Shenker, S.: Sharing the cost of multicast transmissions. *J. Comput. Syst. Sci.* **63**(1), 21–41 (2001)
16. Goemans, M.X., Williamson, D.P.: A general approximation technique for constrained forest problems. *SIAM J. Comput.* **24**(2), 296–317 (1995)
17. Green, J., Kohlberg, E., Laffont, J.J.: Partial equilibrium approach to the free rider problem. *J. Public Econ.* **6**, 375–394 (1976)
18. Gupta, A., Srinivasan, A., Tardos, É.: Cost-sharing mechanisms for network design. *Algorithmica* **50**(1), 98–119 (2008)
19. Hajiaghayi, M.T., Jain, K.: The prize-collecting generalized Steiner tree problem via a new approach of primal-dual schema. In: *Proceedings of the 17th Annual ACM–SIAM Symposium on Discrete Algorithms*, pp. 631–640 (2006)
20. Immorlica, N., Mahdian, M., Mirrokni, V.S.: Limitations of cross-monotonic cost sharing schemes. In: *Proceedings of the 16th Annual ACM–SIAM Symposium on Discrete Algorithms*, pp. 602–611 (2005)
21. Iwata, S., Fleischer, L., Fujishige, S.: A combinatorial strongly polynomial algorithm for minimizing submodular functions. *J. ACM* **48**(4), 761–777 (2001)
22. Jain, K., Vazirani, V.: Applications of approximation algorithms to cooperative games. In: *Proceedings of the 33rd Annual ACM Symposium on the Theory of Computing*, pp. 364–372 (2001)
23. Jain, K., Vazirani, V.V.: Equitable cost allocations via primal-dual-type algorithms. In: *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pp. 313–321 (2002)
24. Kent, K.J., Skorin-Kapov, D.: Population monotonic cost allocations on MSTs. In: *Proceedings of the 6th International Conference on Operational Research*, pp. 43–48 (1996)
25. Könemann, J., Leonardi, S., Schäfer, G., van Zwam, S.: From primal-dual to cost shares and back: a stronger LP relaxation for the Steiner forest problem. In: *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming*, pp. 930–942 (2005)
26. Könemann, J., Leonardi, S., Schäfer, G., van Zwam, S.H.M.: A group-strategyproof cost sharing mechanism for the Steiner forest game. *SIAM J. Comput.* **37**(5), 1319–1341 (2008)
27. Leonardi, S., Schäfer, G.: Cross-monotonic cost sharing methods for connected facility location games. *Theoret. Comput. Sci.* **326**(1–3), 431–442 (2004)
28. Mettu, R.R., Plaxton, C.G.: The online median problem. *SIAM J. Comput.* **32**(3), 816–832 (2003)
29. Moulin, H.: Incremental cost sharing: characterization by coalition strategy-proofness. *Soc. Choice Welfare* **16**, 279–320 (1999)
30. Moulin, H., Shenker, S.: Strategyproof sharing of submodular costs: budget balance versus efficiency. *Econ. Theor.* **18**(3), 511–533 (2001)
31. Nisan, N., Roughgarden, T., Tardos, E., Vazirani, V.V. (eds.): *Algorithmic Game Theory*. Cambridge University Press, Cambridge (2007)
32. Pál, M., Tardos, E.: Group strategyproof mechanisms via primal-dual algorithms. In: *Proceedings of the 44th Symposium on the Foundations of Computer, Science*, pp. 584–593 (2003)

33. Pountourakis, E., Vidali, A.: A complete characterization of group-strategyproof mechanisms of cost-sharing. In: Proceedings of the 18th Annual European Symposium on Algorithms, pp. 146–157 (2010)
34. Roberts, K.: The characterization of implementable choice rules. In: Laffont, J.J. (ed.) Aggregation and Revelation of Preferences. North-Holland, Amsterdam (1979)
35. Roughgarden, T., Sundararajan, M.: Optimal efficiency guarantees for network design mechanisms. In: Proceedings of the 12th International Conference on Integer Programming and Combinatorial Optimization, pp. 469–483 (2007)
36. Roughgarden, T., Sundararajan, M.: Quantifying inefficiency in cost-sharing mechanisms. *J. ACM* **56**(4), 1–33 (2009)
37. Schrijver, A.: A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *J. Comb. Theory Ser. B* **80**(2), 346–355 (2000)