Connected facility leasing problems

Murilo Santos de Lima[1,⋆], Mário César San Felice[2,⋆⋆], and Orlando Lee[1,⋆⋆⋆]

[1] Institute of Computing, UNICAMP, Campinas - SP, Brazil
mslima@ic.unicamp.br, lee@ic.unicamp.br
[2] Department of Computer Science, USP, São Paulo - SP, Brazil
felice@ic.unicamp.br

Abstract. We study leasing variants of the connected facility location problem, in which we wish to connect a set of clients to facilities, and facilities are connected via core edges, whose cost is a scale factor times the cost of a simple edge. We identify two aspects of the problem that can lead to different variants: (a) if there is a single or multiple commodities, and (b) if we lease facilities and buy core edges, or if we lease both facilities and core edges. Combining these aspects, we propose four variants of the problem, and we give approximation and competitive online algorithms for each of them when the (smallest) scale factor is 1. The algorithms we propose follow the technique of combining available algorithms for the underlying facility leasing and Steiner problems.

Keywords: leasing optimization, connected facility location, multi-commodity, approximation algorithms, competitive online algorithms

1 Introduction

In traditional optimization problems, a solution is built by acquiring resources that persist in time. In **leasing optimization problems**, some resources may be leased for different lengths of time and, due to economies of scale, it is more cost-effective to lease a resource for longer periods. Leasing problems arise, for example, in the current trend among start-ups that prefer to lease servers in a cloud service rather than install their own servers [1].

The **parking permit problem** (PP) is the fundamental leasing problem. It was proposed by Meyerson [15], and it has a polynomial-time exact dynamic programming algorithm. For its online version, Meyerson gave a deterministic $\Theta(K)$-competitive algorithm and a randomized $\Theta(\lg K)$-competitive algorithm, where K is the number of permit types. He also studied the leasing version of the Steiner forest problem (SF), called the **Steiner leasing problem** (SLe), and presented a $O(\lg K \lg |V|)$-competitive online algorithm, where K is the number of leasing types and V is the set of vertices. This algorithm combines his randomized online algorithm for PP with the technique of approximating a metric

⋆ Support by FAPESP/CAPES 2014/18781-1, CNPq 142161/2014-4.
⋆⋆ Support by CAPES PNPD 1522390, CNPq 456792/2014-7, FAPESP 2013/03447-6.
⋆⋆⋆ Support by CNPq 311373/2015-1, FAPESP 2015/11937-9.

by a tree metric [4, 7]. Anthony and Gupta [3] presented approximation algorithms for offline leasing versions of several NP-hard problems. For the **facility leasing problem** (FLe) and the **Steiner tree leasing problem** (STLe), they gave $O(K)$-approximation algorithms.[3] For FLe, Nagarajan and Williamson [16] obtained a 3-approximation algorithm and a $O(K \lg n)$-competitive online algorithm, where n is the number of client requests. Abshoff *et al.* [1] presented a $O(\delta_K \lg \delta_K)$-competitive online algorithm for FLe, where δ_K is the longest leasing duration. For the **facility leasing problem with penalties** (FLeP), de Lima, San Felice and Lee [12] gave a 3-approximation algorithm, and San Felice *et al.* [17] gave a $O(K \lg n)$-competitive online algorithm.

The **connected facility location problem** (CFL) is a two-layer network design problem in which we connect clients via facilities connected through a core tree. We are given a complete graph $G = (V, E)$ with a metric distance $d : V \times V \mapsto \mathbb{R}_+$, a set $F \subseteq V$ of potential facilities with corresponding opening costs, a root facility $r \in F$, a constant $M \geq 1$ and, for $t = 1, \ldots, T$, a set $D_t \subseteq V$ of clients arriving at instant t. The goal is to select a subset of facilities to open and a subset of edges that connects open facilities to r, which minimize the cost of the open facilities, plus M times the cost of the edges that connect facilities, plus the sum of the distances from each client to its closest open facility. Grandoni and Rothvoß [9] gave a 3.19-approximation algorithm for CFL. Its online version has $O(\lg n)$-competitive algorithms [22, 19] and $\Omega(\lg n)$ lower bound. The **multi-commodity connected facility location problem** (MCFL) is a generalization of CFL in which we connect pairs of clients, either directly or via facilities connected through a core forest. Grandoni and Rothvoß [9] proposed MCFL and gave a 16.2-approximation algorithm. Its online version has a $O(\lg^2 n)$-competitive algorithm and, when $M = 1$, a $O(\lg n)$-competitive algorithm, both by San Felice, Fernandes and Lintzmayer [18].

1.1 Our contribution

We propose leasing variants of CFL and MCFL, and we give approximation and competitive online algorithms for a special case of these problems. The leasing variants we propose model the problem of a network service provider, who has to install cables between routers to serve clients, but resources, such as routers and backbone cables, have different lifetimes which are not negligible.

The first leasing variant of CFL we study is the **connected facility leasing problem** (CFLe), whose online version we proposed and addressed in [13]. The input for CFLe is similar to that for CFL, but we no longer open facilities for unlimited time. Instead, we lease each facility for one of K different lengths of time, which we denote by $\delta_1, \ldots, \delta_K$. The cost γ_k^f of leasing a facility $f \in F$ depends on f, but also on the leasing type $k \in [K]$. Leasing facility r has cost zero for any leasing length. If we lease a facility f with leasing type k at instant $\hat{t}$,

[3] Since the facility location problem and the Steiner tree problem are NP-hard [20, 8], so are their leasing variants. The original problem is a particular case of the leasing variant in which there is a single leasing type with infinite duration.

then we say that **facility lease** $(f, k, \hat{t})$ is **active** during interval $[\hat{t}, \hat{t} + \delta_k)$. The goal is to minimize the sum of the costs of the facility leases, plus the sum of the distances from each client to its assigned facility lease, plus M times the cost of a set of edges connecting leased facilities to r. FLE reduces to CFLE, so this is an NP-hard problem. In Section 2, we present a 7.39-approximation algorithm for CFLE when $M = 1$, which combines the 3-approximation algorithm for FLE [16] and the 1.39-approximation algorithm for the Steiner tree problem (ST) [6]. We also presented a $\mathrm{O}(K \lg n)$-competitive online algorithm for CFLE when $M = 1$ in a previous paper [13]. That algorithm combines the $\mathrm{O}(K \lg n)$-competitive online algorithm for FLE [16] with the $\mathrm{O}(\lg n)$-competitive online algorithm for ST [10]. This problem is discussed in Section 2.

The second variant we study is the **leasing-connected facility leasing problem** (LeCFLE), in which we lease both facilities and edges connecting facilities to the root. We have K^F types of facility leases, and K^E types of core edge leases. Facilities and core edges may have different leasing lengths. Instead of a single scaling factor M, we have K^E scaling factors $\gamma_1^E, \ldots, \gamma_{K_E}^E$, one for each core edge leasing length. Thus, to lease an edge e with leasing type k_e costs $\gamma_{k_e}^E \cdot d(e)$. We wish to assign an active facility lease to each client and, for each instant in which a facility serves a client, there must exist a path of active edge leases from that facility to the root. We wish to minimize the cost of leasing facilities, plus the cost of connecting clients to facilities, plus the cost of leasing core edges. For the case in which the smallest edge leasing scaling factor γ_1^E is equal to 1, we give a $\mathrm{O}(K^E)$-approximation algorithm and a $\mathrm{O}(K^F \lg n + \lg K^E \lg |V|)$-competitive online algorithm. The approach is similar to the one we use for CFLE, but we change the building block algorithm of the core tree. The offline algorithm uses the $\mathrm{O}(K)$-approximation algorithm for STLE [3], and the online algorithm uses the $\mathrm{O}(\lg K \lg |V|)$-competitive online algorithm for SLE [15]. This problem is addressed in Section 3.

We also study two leasing variants of MCFL: the **multi-commodity connected facility leasing problem** (MCFLE), in which facilities are leased but core edges are permanent, and the **multi-commodity leasing-connected facility leasing problem** (MLeCFLE), in which both facilities and core edges are leased. For MCFLE with $M = 1$, we give an 8-approximation algorithm, which combines the 3-approximation for FLEP [12] with the 2-approximation for SF [2]. For its online version, we give a $\mathrm{O}(K \lg n)$-competitive algorithm, which combines the $\mathrm{O}(K \lg n)$-competitive online algorithm for FLEP [17] with the $\mathrm{O}(\lg n)$-competitive online algorithm for SF [5]. Note that both facility leasing and Steiner problems are different from the ones we use to solve CFLE. For MLeCFLE with $\gamma_1^E = 1$, we give a $\mathrm{O}(\lg n)$-approximation algorithm and a $\mathrm{O}(K^F \lg n + \lg K^E \lg |V|)$-competitive online algorithm. Here the underlying facility leasing problem is FLEP, as in MCFLE, and the underlying Steiner problem is SLE, as in LeCFLE. These results are detailed in Section 4.

We summarize the approximation/competitive factors we obtain in Table 1. Our technique for solving these problems for the case with $M = 1$ ($\gamma_1^E = 1$), both in offline and online settings, consists in solving the associated facility leasing problem, buying (leasing) a core network that connects the clients, and then buying (leasing) core edges between each client and its corresponding facility lease. The guarantee follows from the analysis of the corresponding building block algorithms, and from reducing the optimal solution of the connected facility leasing problem to a feasible solution of the building block problems. This approach is used in the literature to solve CFL and MCFL when $M = 1$ [21, 19, 18]. The general cases ($M > 1$ and $\gamma_1^E > 1$, respectively) for the four problems we study, both in offline and online settings, are open.

Table 1. Summary of our results for connected facility leasing problems.

problem	offline algorithm	online algorithm
CFLe	7.39 if $M = 1$	$\mathrm{O}(K \lg n)$ if $M = 1$ [13]
LeCFLe	$\mathrm{O}(K^E)$ if $\gamma_1^E = 1$	$\mathrm{O}(K^F \lg n + \lg K^E \lg \lvert V\rvert)$ if $\gamma_1^E = 1$
MCFLe	8 if $M = 1$	$\mathrm{O}(K \lg n)$ if $M = 1$
MLeCFLe	$\mathrm{O}(\lg n)$ if $\gamma_1^E = 1$	$\mathrm{O}(K^F \lg n + \lg K^E \lg \lvert V\rvert)$ if $\gamma_1^E = 1$

2 Connected facility leasing

In CFLe, we are given a complete graph $G = (V, E)$ with a metric distance $d : V \times V \mapsto \mathbb{R}_+$, a set $F \subseteq V$ of potential facilities, a root facility $r \in F$, K leasing lengths $\delta_1, \ldots, \delta_K \in \mathbb{N}$, a cost $\gamma_k^f \in \mathbb{R}_+$ for leasing facility $f \in F$ with leasing type $k \in [K]$ (ensuring $\gamma_k^r = 0$ for any k), a constant $M \geq 1$ and, for $t = 1, \ldots, T$, a set $D_t \subseteq V$ of clients arriving at instant t. For simplicity, we denote by $\mathcal{D} := \{(j, t) : j \in D_t \text{ for } t \in [T]\}$ the set of **client requests**. The goal is to find a set $X \subseteq F \times [K] \times [T]$ of facility leases, a function $a : \mathcal{D} \mapsto X$ that assigns each client request (j, t) to a facility lease $(f, k, \hat{t})$ such that $t \in [\hat{t}, \hat{t} + \delta_k)$, and a set $\mathcal{T} \subseteq E$ of edges connecting X to r; and we wish to minimize $\sum_{(f,k,\hat{t}) \in X} \gamma_k^f + \sum_{(j,t) \in \mathcal{D}} d(a(j, t), j) + M \cdot \sum_{e \in \mathcal{T}} d(e)$.

Algorithm 1 is a 7.39-approximation for CFLe when $M = 1$. First, we obtain a solution of FLe on the clients, by using the 3-approximation algorithm by Nagarajan and Williamson [16], which we denote by NW-FLe. Then we build a tree connecting the clients to the root, using an approximation algorithm for ST, and finally we add an edge in the tree between each client and the facility lease that serves it. The approximation factor of our algorithm can be expressed as $6 + \alpha_{\mathrm{ST}}$, where $\alpha_{\mathrm{ST}} \approx 1.39$ is the best approximation factor for ST [6].

Theorem 1. *Algorithm 1 is a $(6 + \alpha_{\mathrm{ST}})$-approximation when $M = 1$.*

Proof. Given a solution returned by the algorithm, let L be the facility leasing cost, C the client connection cost, and S the core tree cost. Similarly, let L^*, C^* and S^* be those costs on an optimum solution.

Let L' be the facility leasing cost and C' be the client connection cost of the solution returned by NW-FLe. We have that $L + C = L' + C' \leq 3 \cdot \mathrm{opt}_{\mathrm{FLe}}$. Since

Input: $(G, d, F, r, K, \delta, \gamma, M, D_1, \ldots, D_T)$
set $\gamma_K^r \leftarrow 0$;
$(X, a) \leftarrow \text{NW-FLE}(G, d, F, K, \delta, \gamma, D_1, \ldots, D_T)$;
$\mathcal{T} \leftarrow \text{ST}(G, d, \mathcal{D} \cup \{r\})$;
$\mathcal{T} \leftarrow \mathcal{T} \cup \{(j, a(j,t)) : (j,t) \in \mathcal{D}\}$;
return $(X, a, \mathcal{T})$;

Algorithm 1: Approximation algorithm for CFLE.

an optimum solution for CFLE induces a feasible solution for FLE, $\text{opt}_{\text{FLe}} \leq L^* + C^*$, so $L + C \leq 3 \cdot (L^* + C^*)$.

Since $M = 1$, we bound S by the cost of solving ST on $\mathcal{D}$, plus the cost of connecting each client to its assigned facility lease. Thus, $S \leq \alpha_{\text{ST}} \cdot \text{opt}_{\text{ST}} + C$. Since the optimum core tree combined with an edge between each client and its optimum facility induces a feasible solution for ST on $\mathcal{D}$, we have that $\text{opt}_{\text{ST}} \leq S^* + C^*$ and the theorem follows. □

This algorithm has approximation factor $\Omega(M)$ if $M > 1$: take an instance with a single facility f with $\gamma_1^f = 0$ and $d(f, r) = 1$. Then take one client request at the same point as f. The algorithm will lease f and connect it to the root, for a total cost M, while the optimum solution will connect the client to the root by paying 1. We do not know if there is a constant-approximation algorithm for CFLE if $M > 1$. In particular, the technique of sample-and-augment does not seem to lead to a good approximation algorithm. Suppose an algorithm which samples each client request with probability $1/M$, then solves FLE for the sampled client requests, buys a core tree between the sampled clients, buys a core edge between each sampled client and its assigned facility lease, and finally assigns non-sampled clients to the closest active facility lease. There is an example which shows that such an algorithm has approximation factor $\Omega(n)$ if $M > 1$: take a single facility f with $d(f, r) = 1$, then take $K = 1$, $\delta_1 = 1$, $\gamma_1^f = 0$, and $M = 2$. For $t = 1, \ldots, n$ with $n \geq 2$, take a single client request at instant t on the same point as f. Note that the optimum solution buys the core edge (f, r) (cost 2) and always leases facility f (cost zero), paying a total cost of 2. In expectation, the sample-and-augment algorithm will sample half of the client requests, buy the core edge, and pay a cost of $1/2$ to serve each client request, since if the client request is not sampled, then it has to be served by the root. Thus the expected cost is $n/2 + 2$. Even if one considers more clever sample-and-augment algorithms for CFL, such as in [9], the available analysis techniques seem insufficient. All the approximation algorithms for CFL in the literature bound the client connection cost via the cost of a tree connecting the clients. In CFLE, however, these costs are not so tightly related since, while core edges are permanent, facility leases cease after some time. Thus, a client which is close to the tree may have to be connected to a facility which is further apart.

In the online version of CFLE, numbers T and $n := \sum_{t=1}^{T} |D_t|$ are unknown. Sets D_t are revealed one at a time, and we cannot remove edges from $\mathcal{T}$, change facility leases, or modify $a(j,t)$ once it is chosen. The competitive factor is

$\Omega(\lg K + \lg n)$, due to the lower bounds on PP [15] and on ST [10]. We presented an online algorithm for CFLE in [13]. The idea is to maintain a virtual solution of the online algorithm for FLE [16], and to sample clients as they arrive, with probability $1/M$. Sampled clients are connected to the tree in a greedy manner, as in the online algorithm for ST [10], the corresponding facility lease given by the virtual solution is leased, and it is connected to the tree via the client. Non-sampled clients are connected to the closest active facility lease. The algorithm is $\mathrm{O}(K \lg n)$-competitive when $M = 1$, but the same example in the previous paragraph shows that its competitive factor is $\Omega(n)$ if $M > 1$.

3 Leasing-connected facility leasing

In LECFLE, we are given a complete graph $G = (V, E)$ with a metric distance $d : V \times V \mapsto \mathbb{R}_+$, a set $F \subseteq V$ of potential facilities, a root facility $r \in V$, and for $t = 1, \ldots, T$, a set $D_t \subseteq V$ of clients arriving at instant t. However, we may have different leasing types for facilities and core edges. Thus, we are given K^F facility leasing lengths $\delta_1^F, \ldots, \delta_{K^F}^F \in \mathbb{N}$, and leasing facility $f \in F$ with leasing type $k \in [K^F]$ costs $\gamma_k^f \in \mathbb{R}_+$. We also have K^E edge leasing lengths $\delta_1^E, \ldots, \delta_{K^E}^E \in \mathbb{N}$, edge leasing factors $\gamma_1^E, \ldots, \gamma_{K^E}^E \geq 1$, and leasing core edge $e \in E$ with leasing type k_e costs $d(e) \cdot \gamma_{k_e}^E$. We wish to find a set $X \subseteq F \times [K^F] \times [T]$ of facility leases, a function $a : \mathcal{D} \mapsto X$ that assigns a facility lease $a(j, t) \in X$ which is active at instant t to each client request $(j, t) \in \mathcal{D}$, and a set of edge leases $\mathcal{T} \subseteq E \times [K^E] \times [T]$. For each client request (j, t) with $a(j, t) = (f, k, \hat{t})$, there must exist a path P from f to r in G such that each edge $e \in P$ has some edge lease in $\mathcal{T}$ which is active at instant t. The goal is to find a solution which minimizes $\sum_{(f,k,\hat{t}) \in X} \gamma_k^f + \sum_{(j,t) \in \mathcal{D}} d(a(j, t), j) + \sum_{(e,k_e,\hat{t}_e) \in \mathcal{T}} \gamma_{k_e}^E \cdot d(e)$. Note that we do not lease edges connecting clients to facilities.[4]

A standard technique for solving network design problems, especially in an online setting, is to approximate a metric (or the distances in a graph) by a metric induced by the distances in a tree. Bartal [4] proposed this technique and showed how to approximate an arbitrary finite metric (V, d) by a tree metric with expected "distortion" $\mathrm{O}(\lg^2 |V|)$. After a series of improvements, Fakcharoenphol, Rao and Tawar [7] showed how to obtain a tree metric with expected distortion $\mathrm{O}(\lg |V|)$, which is asymptotically optimal. We denote this algorithm as FRT. The following result on this technique is used in our discussion.

Theorem 2 ([4, 7]). *Given a minimization problem on a finite metric* (V, d) *whose objective function is a non-negative linear combination of distances in* d*, if there is an* α*-competitive algorithm for the special case of tree metrics, then there is a randomized* $\mathrm{O}(\alpha \cdot \lg |V|)$*-competitive algorithm for the general case.*

[4] Our definition is more general than if we leased edges between clients and facilities: since those edges would always be leased with type 1 because they are not reusable, that would be equivalent to divide all edge leasing costs by γ_1^E and have $\gamma_1^E = 1$. Similarly, in this variant we do not have a scaling parameter M; this is equivalent to multiply all edge leasing costs by M, thus our definition of the problem is more general, since edge leasing costs do not necessarily share a common divisor.

This theorem is applied by Meyerson [15] to solve SLe. If the input metric is a tree, then SLe consists in solving, for each edge, an instance of PP, which is defined as follows. We are given K permit types with lengths $\delta_1, \ldots, \delta_K$ and costs $\gamma_1, \ldots, \gamma_K$, respectively, and a sequence of requests $r_0, \ldots, r_{T-1} \in \{0, 1\}$. The goal is to find a minimum-cost set $S \subseteq [K] \times \{0, \ldots, T-1\}$ such that, for each t with $r_t = 1$, there is some $(k, \hat{t}) \in S$ such that $t \in [\hat{t}, \hat{t}+\delta_k)$. PP has a polynomial-time exact algorithm. In its online version, T is unknown and $r_0, \ldots, r_{T-1}$ are revealed one at a time, and the problem has a deterministic $\Theta(K)$-competitive algorithm and a randomized $\Theta(\lg K)$-competitive algorithm [15]. By Theorem 2, SLe admits a $\mathrm{O}(\lg n)$-approximation and an online $\mathrm{O}(\lg K \lg |V|)$-competitive algorithm.

We propose, in Algorithm 2, an online algorithm for LeCFLe, which is $\mathrm{O}(K^F \lg n + \lg K^E \lg |V|)$-competitive if $\gamma_1^E = 1$, where $n := \sum_{t=1}^{T} |D_t|$. The algorithm utilizes as a subroutine the online $\mathrm{O}(K \lg n)$-competitive algorithm for FLe by Nagarajan and Williamson [16], which we denote by NW-OFLe. We also utilize as a subroutine the online randomized $\mathrm{O}(\lg K)$-competitive algorithm for PP by Meyerson [15], which we denote by RandOPP.

Input: $(G, d, F, r, K^F, \delta^F, \gamma, K^E, \delta^E, \gamma^E)$

$\mathcal{T} \leftarrow \emptyset$; **set** $\gamma^r_{K^F} \leftarrow 0$;
initialize NW-OFLe with $(G, d, F, K^F, \delta^F, \gamma)$ and let (X, a) be the virtual solution;
$\mathscr{T} \leftarrow \mathrm{FRT}(V, d)$;
foreach edge e of $\mathscr{T}$ **do**
  initialize an instance RandOPP$[e]$ with $(K^E, \delta^E, \gamma^E)$;
when D_t arrives **do**
  foreach $j \in D_t$ **do**
    $P \leftarrow \mathrm{path}_{\mathscr{T}}(j, r)$;
    foreach edge $e \in P$ **do**
      send 1 at instant t to RandOPP$[e]$, obtaining a permit (k_e, t_e);
      $\mathcal{T} \leftarrow \mathcal{T} \cup \{(e, k_e, t_e)\}$;
    send (j, t) to NW-OFLe and update the virtual solution (X, a);
    let $(f, k, \hat{t}) \leftarrow a(j, t)$; $\mathcal{T} \leftarrow \mathcal{T} \cup \{((f, j), 1, t)\}$;
return $(X, a, \mathcal{T})$;

Algorithm 2: Online algorithm for LeCFLe.

The algorithm builds, utilizing algorithm FRT, a tree $\mathscr{T}$ with $V_{\mathscr{T}} = V$ whose distances $\mathrm{O}(\lg |V|)$-approximate the distances in G. Then, for each edge of the tree, the algorithm maintains an instance of algorithm RandOPP. Besides, similarly to our online algorithm for CFLe [13], the algorithm maintains a virtual solution of algorithm NW-OFLe. Algorithm 2, however, will always follow the decisions of the virtual solution. For each client (j, t) that arrives, the algorithm leases the edges in the path from j to r in $\mathscr{T}$, using the leasing types suggested by the corresponding instances of algorithm RandOPP. The algorithm leases the facility suggested by the virtual solution of algorithm NW-OFLE, and connects the facility to the tree using an edge lease of type 1 through j.

Theorem 3. *Algorithm 2 is* $\mathrm{O}(K^F \lg n + \lg K^E \lg |V|)$*-competitive when* $\gamma_1^E = 1$.

Proof. Given a solution returned by the algorithm, let L be the facility leasing cost, C the client connection cost, and S the core tree cost. Similarly, let L^*, C^* and S^* be those costs on an optimum solution.

Let L' be the facility leasing cost and C' be the client connection cost of the virtual solution produced by NW-OFLE. We have that $L + C = L' + C' = \mathrm{O}(K \lg n) \cdot \mathrm{opt}_{\mathrm{FLe}}$. Since an optimum solution for CFLE induces a feasible solution for FLE, $\mathrm{opt}_{\mathrm{FLe}} \leq L^* + C^*$, so $L + C = \mathrm{O}(K \lg n) \cdot (L^* + C^*)$.

We bound S by the cost of solving SLE on $\mathcal{D}$, plus the cost of an edge lease of type 1 between each client and its assigned facility lease, which is the connection cost for that client since $\gamma_1^E = 1$. Thus, $S \leq \mathrm{O}(\lg K^E \lg |V|) \cdot \mathrm{opt}_{\mathrm{SLe}} + C$. Since the optimum core tree combined with an edge lease of type 1 between each client and its optimum facility induces a feasible solution for SLE on $\mathcal{D}$, we have that $\mathrm{opt}_{\mathrm{SLe}} \leq S^* + C^*$ and the theorem follows. ☐

If we substitute algorithm ST with the algorithm by Anthony and Gupta for STLE, which is a $\mathrm{O}(K)$-approximation [3], at Line 3 of Algorithm 1, then we obtain an offline algorithm for LECFLE which is a $\mathrm{O}(K^E)$-approximation. The proof is similar to that of Theorem 1; we omit it due to space constraints.

4 Multi-commodity connected facility leasing

In MCFLE, the input consists of a complete graph $G = (V, E)$, a metric distance $d : V \times V \mapsto \mathbb{R}_+$, a set $F \subseteq V$ of potential facilities, K facility leasing lengths $\delta_1, \ldots, \delta_k \in \mathbb{N}$, a cost $\gamma_k^f \in \mathbb{R}_+$ for leasing facility $f \in F$ with leasing type $k \in [K]$, a constant $M \geq 1$ and a sequence $P_1, \ldots, P_T \subseteq V \times V$ of sets of pairs of clients. We denote by $\mathcal{F} := F \times [K] \times [T]$ the set of possible facility leases. Let $X \subseteq \mathcal{F}$, $\mathcal{T} \subseteq E$, $u, v \in V$ and $t \in [T]$; we denote by $d_{X,\mathcal{T}}(u, v, t)$ the distance between u and v in the graph in which we add to G edges of cost zero between each pair of facility leases in X that are in the same tree in $\mathcal{T}$ and that are active at instant t. We also denote by $\mathcal{P} := \{(u, v, t) : (u, v) \in P_t \text{ for } t \in [T]\}$ the set of client pair requests. The goal is to find a set $X \subseteq \mathcal{F}$ of facility leases and a forest $\mathcal{T} \subseteq E$ which minimize $\sum_{(f,k,\hat{t}) \in X} \gamma_k^f + \sum_{(u,v,t) \in \mathcal{P}} d_{X,\mathcal{T}}(u, v, t) + M \cdot \sum_{e \in \mathcal{T}} d(e)$.

In this section we present an offline and an online algorithm for problem MCFLE, which are extensions of Algorithm 1 and of algorithm SIMPLEOCFLE presented in [13]. Both algorithms use as a subroutine algorithms for FLEP, in which we may choose not to serve a client request (j, t) by paying a penalty $\pi(j, t)$.

We present, in Algorithm 3, an offline algorithm for MCFLE. The algorithm runs an instance of the 3-approximation primal-dual algorithm for FLEP [12], which we denote by PRIMAL-DUALFLEP. For each pair $(u, v) \in P_t$, we assign a penalty of $d(u, v)/2$ to both (u, t) and (v, t). If both client requests are assigned to facility leases in the FLEP solution, then the algorithm includes a set consisting of these two clients in $\mathcal{D}'$; otherwise, the algorithm will connect u and v directly. Then the algorithm runs the 2-approximation primal-dual algorithm for SF [2] on the pairs in $\mathcal{D}'$, and adds an edge between each client in $\mathcal{D}'$ and the facility assigned to the corresponding client request. The algorithm then returns only facility leases that serve client requests in $\mathcal{D}'$.

Input: $(G, d, F, K, \delta, \gamma, M, P_1, \ldots, P_T)$
foreach $(u, v) \in P_t$ **do set** $\pi(u, t) \leftarrow \pi(v, t) \leftarrow d(u, v)/2$;
let $D_t \leftarrow \bigcup_{(u,v)\in P_t} \{u, v\}$;
$(X', a') \leftarrow$ Primal-DualFLeP$(G, d, F, K, \delta, \gamma, D_1, \ldots, D_T, \pi)$;
$\mathcal{D}' \leftarrow \{\{(u, t), (v, t)\} : (u, v) \in P_t : a'(u, t) \neq \mathsf{null} \text{ and } a'(v, t) \neq \mathsf{null}\}$;
$X \leftarrow \{(f, k, \hat{t}) \in X' : \exists (u, t) \in \mathcal{D}' : a'(u, t) = (f, k, \hat{t})\}$;
$\mathcal{T} \leftarrow \mathrm{SF}(G, d, \mathcal{D}') \cup \{(u, a'(u, t)) : (u, t) \in \mathcal{D}'\}$;
return $(X, \mathcal{T})$;

Algorithm 3: Approximation algorithm for MCFLe.

Theorem 4. *Algorithm 3 is an* 8*-approximation if* $M = 1$.

Proof. Given a solution returned by the algorithm, let L be the facility leasing cost, C the client connection cost, and S the core tree cost. Similarly, let L^*, C^* and S^* be those costs on an optimum solution.

Let L' be the facility leasing cost, C' be the client connection cost, and Π' be the penalty cost of the solution returned by Primal-DualFLeP on $\mathcal{D} := \bigcup_{t=1}^{T} D_t$. We have that $L' + C' + \Pi' \leq 3 \cdot \mathrm{opt}_{\mathrm{FLeP}}$. Given an optimum solution for MCFLe on $\mathcal{P}$, we obtain a feasible solution for FLeP on $\mathcal{D}$ by paying the penalties for the clients in the pairs that are connected directly, and by assigning the other clients to the closest facility lease in the path connecting the pair. Thus, $\mathrm{opt}_{\mathrm{FLeP}} \leq L^* + C^*$ and $L' + C' + \Pi' \leq 3 \cdot (L^* + C^*)$.

Since Algorithm 3 leases a subset of the facility leases in X', clearly $L \leq L'$.

For a request (u, v, t) connected directly by Algorithm 3, we have that one of (u, t) and (v, t) pays for its penalty cost. Suppose w.l.o.g. that it is (u, t); then $d_{X,\mathcal{T}}(u, v, t) \leq 2 \cdot \pi(u, t)$. For a request (u, v, t) connected through the core tree, our algorithm adds to the tree a whole path between $a'(u, t)$ and $a'(v, t)$, so $d_{X,\mathcal{T}}(u, v, t) \leq d(u, a'(u, t)) + d(v, a'(v, t))$. Thus $C \leq 2 \cdot \Pi' + C'$.

Since $M = 1$, we bound S by the cost of solving SF on $\mathcal{D}'$, plus the cost of connecting each client in $\mathcal{D}'$ to its assigned facility lease. Thus, $S \leq 2 \cdot \mathrm{opt}_{\mathrm{SF}}(\mathcal{D}') + C' \leq 2 \cdot \mathrm{opt}_{\mathrm{SF}}(\mathcal{P}) + C'$. Since the optimum core tree combined with the optimum client connection edges induces a feasible solution for SF on $\mathcal{P}$, we have that $\mathrm{opt}_{\mathrm{SF}}(\mathcal{P}) \leq S^* + C^*$. Substituting the previous inequalities, we have that

$$\begin{aligned} L + C + S &\leq L' + 2 \cdot \Pi' + C' + 2 \cdot (S^* + C^*) + C' \\ &\leq 2 \cdot (S^* + C^*) + 2 \cdot (L' + \Pi' + C') \\ &\leq 2 \cdot (S^* + C^*) + 6 \cdot (L^* + C^*) \leq 8 \cdot (L^* + C^* + S^*) \ . \end{aligned}$$

□

In the online version of MCFLE, numbers T and $n := \sum_{t=1}^{T} |D_t|$ are unknown, sets P_t are revealed one at a time, and we must return sequences of sets $X_1, \ldots, X_T \subseteq \mathcal{F}$ and $\mathcal{T}_1, \ldots, \mathcal{T}_T \subseteq E$, where $X_t \supseteq X_{t-1}$ and $\mathcal{T}_t \supseteq \mathcal{T}_{t-1}$ for $t = 2, \ldots, T$, and for $t = 1, \ldots, T$, we must build $X_t, \mathcal{T}_t$ only with the information of $P_1, \ldots, P_t, X_1, \ldots, X_t, \mathcal{T}_1, \ldots, \mathcal{T}_t$. Also, the objective function is slightly different: minimize $\sum_{(f,k,\hat{t}) \in X_T} \gamma_k^f + \sum_{(u,v,t) \in \mathcal{P}} d_{X_t, \mathcal{T}_t}(u, v, t) + M \cdot \sum_{e \in \mathcal{T}_T} d(e)$; i.e., we can only use facility leases and core edges bought up to instant t to connect clients arriving at instant t.

We give the following online algorithm for this problem. The algorithm combines ideas from Algorithm 3 and algorithm SIMPLEOCFLE [13]. We maintain a virtual solution of the online algorithm by San Felice *et al.* [17] for FLEP, which is $O(K \lg n)$-competitive and which we denote by SFCLWF-OFLEP. We also maintain a virtual solution of the online algorithm by Berman and Coulston for SF [5], which is $O(\lg n)$-competitive and which we denote by BC-OSF. For each pair of clients (u, v) that arrives at instant t, we send two requests to algorithm SFCLWF-OFLEP: (u, t) and (v, t), with $\pi(u, t) = \pi(v, t) = d(u, v)/2$. If algorithm SFCLWF-OFLEP decides to pay the penalty for at least one of those requests, then we connect them directly. Otherwise, algorithm SFCLWF-OFLEP leases a facility for each client request: we mock this behavior and lease both facilities. Then we buy core edges connecting u and v via algorithm BC-OSF. We also buy the edges between u, v and their respective facility leases.

Input: $(G, d, F, K, \delta, \gamma, M)$
$X_1 \leftarrow \emptyset$, $\mathcal{T}_1 \leftarrow \emptyset$;
initialize SFCLWF-OFLEP with $(G, d, F, K, \delta, \gamma)$; **let** (X', a') be the virtual solution;
initialize BC-OSF with (G, d) and **let** $\mathcal{T}'$ be the virtual solution;
when P_t arrives **do**
    if $t > 1$ **then** $X_t \leftarrow X_{t-1}$, $\mathcal{T}_t \leftarrow \mathcal{T}_{t-1}$;
    foreach $(u, v) \in P_t$ **do**
        set $\pi(u, t) \leftarrow \pi(v, t) \leftarrow d(u, v)/2$;
        send $(u, t, \pi(u, t))$ and $(v, t, \pi(v, t))$ to SFCLWF-OFLEP and update the virtual solution (X', a');
        if $a'(u, t) \neq \mathsf{null}$ **and** $a'(v, t) \neq \mathsf{null}$ **then**
            $X_t \leftarrow X_t \cup \{a'(u, t), a'(v, t)\}$;
            send (u, v) to BC-OSF and update the virtual solution $\mathcal{T}'$;
            $\mathcal{T}_t \leftarrow \mathcal{T}_t \cup \mathcal{T}' \cup \{(u, a'(u, t)), (v, a'(v, t))\}$;
return $(X_1, \ldots, X_T, \mathcal{T}_1, \ldots, \mathcal{T}_T)$;

Algorithm 4: Online algorithm for MCFLE.

Theorem 5. *Algorithm 4 is $O(K \lg n)$-competitive if $M = 1$.*

Proof. Given a solution returned by the algorithm, let L be the facility leasing cost, C the client connection cost, and S the core tree cost. Similarly, let L^*, C^* and S^* be those costs on an optimum solution.

Denote by $\mathcal{D}$ the set of client requests as in Line 8 of the algorithm. Let L' be the facility leasing cost, C' be the client connection cost, and Π' be the penalty cost of the virtual solution produced by SFCLWF-OFLeP on $\mathcal{D}$. We have that $L' + C' + \Pi' = \mathrm{O}(K \lg n) \cdot \mathrm{opt}_{\mathrm{FLeP}}$. Given an optimum solution for MCFLe on $\mathcal{P}$, we obtain a feasible solution for FLeP on $\mathcal{D}$ by paying the penalties for the clients in the pairs that are connected directly, and by assigning the other clients to the closest facility lease in the path connecting the pair. Thus, $\mathrm{opt}_{\mathrm{FLeP}} \leq L^* + C^*$ and $L' + C' + \Pi' = \mathrm{O}(K \lg n) \cdot (L^* + C^*)$.

Since Algorithm 4 leases a subset of the facility leases in X', clearly $L \leq L'$.

For a request (u, v, t) connected directly by Algorithm 4, we have that one of (u, t) and (v, t) pays for its penalty cost. Suppose w.l.o.g. that it is (u, t); then $d_{X_t,\mathcal{T}_t}(u, v, t) \leq 2 \cdot \pi(u, t)$. For a request (u, v, t) connected through the core tree, our algorithm adds to the tree a whole path between $a'(u, t)$ and $a'(v, t)$, so $d_{X_t,\mathcal{T}_t}(u, v, t) \leq d(u, a'(u, t)) + d(v, a'(v, t))$. Thus $C \leq 2 \cdot \Pi' + C'$.

Let $\mathcal{D}'$ be the set of client requests that are connected through the core tree in Line 12 of Algorithm 4. Since $M = 1$, we bound S by the cost of the virtual solution produced by BC-OSF on $\mathcal{D}'$, plus the cost of connecting each client in $\mathcal{D}'$ to its assigned facility lease. Thus, $S \leq \mathrm{O}(\lg n) \cdot \mathrm{opt}_{\mathrm{SF}}(\mathcal{D}') + C' \leq \mathrm{O}(\lg n) \cdot \mathrm{opt}_{\mathrm{SF}}(\mathcal{P}) + C'$. Since the optimum core tree combined with the optimum client connection edges induces a feasible solution for SF on $\mathcal{P}$, we have that $\mathrm{opt}_{\mathrm{SF}}(\mathcal{P}) \leq S^* + C^*$ and the theorem follows. $\square$

Due to space constraints, we omit the presentation of the offline and online algorithms for MLeCFLe, since they follow the ideas in the algorithms for LeCFLe and MCFLe. Note, however, that the algorithm by Anthony and Gupta [3] solves STLe, not SLe. We described a $\mathrm{O}(\lg n)$-approximation algorithm for SLe on Section 3, which uses the same approach as the online algorithm by Meyerson [15]. Replacing this algorithm with algorithm SF on Line 6 of Algorithm 3, we obtain a $\mathrm{O}(\lg n)$-approximation algorithm for MLeCFLe. The online algorithm simply combines Algorithm 2 and Algorithm 4; the proof of the competitive factor is also similar.

5 Future research directions

Future work includes to obtain good approximation and competitive online algorithms for connected facility leasing problems in the case when the (smallest) scale factor is greater than 1, which is an open question. Another research direction is to study connected facility location problems in variations of the leasing model, such as when a demand has a time window to be served after its arrival [11], or when a demand lasts for an unknown period of time [14].

References

1. Abshoff, S., Kling, P., Markarian, C., auf der Heide, F.M., Pietrzyk, P.: Towards the price of leasing online. J. Comb. Optim. pp. 1–20 (2015)

2. 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)
3. Anthony, B.M., Gupta, A.: Infrastructure leasing problems. In: Integer Programming and Combinatorial Optimization, LNCS, vol. 4513, pp. 424–438 (2007)
4. Bartal, Y.: Probabilistic approximations of metric spaces and its algorithmic applications. In: Proc. 37th Conf. Found. Comput. Sci. pp. 184–193 (1996)
5. Berman, P., Coulston, C.: On-line algorithms for Steiner tree problems (extended abstract). In: Proc. 29th Annu. ACM Symp. Theory Comput. pp. 344–353 (1997)
6. Byrka, J., Grandoni, F., Rothvoß, T., Sanità, L.: Steiner tree approximation via iterative randomized rounding. J. ACM 60(1) (2013)
7. 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)
8. Garey, M.R., Johnson, D.S.: Computers and Intractability: a Guide to the Theory of NP-Completeness. Freeman (1979)
9. Grandoni, F., Rothvoß, T.: Approximation algorithms for single and multi-commodity connected facility location. In: Integer Programming and Combinatoral Optimization, LNCS, vol. 6655, pp. 248–260 (2011)
10. Imase, M., Waxman, B.M.: Dynamic Steiner tree problem. SIAM J. Discrete Math. 4(3), 369–384 (1991)
11. Li, S., Mäcker, A., Markarian, C., auf der Heide, F. M., Riechers, S.: Towards flexible demands in online leasing problems. In: Computing and Combinatorics, LNCS, vol. 9198, pp. 277–288 (2015)
12. de Lima, M.S., San Felice, M.C., Lee, O.: Facility leasing with penalties. In: Anais do XXXVII Congresso da Sociedade Brasileira de Computação. pp. 27–30 (2017)
13. de Lima, M.S., San Felice, M.C., Lee, O.: On a leasing variant of the online connected facility location problem. In: Anais do XXXVI Congresso da Sociedade Brasileira de Computação. pp. 836–839 (2016)
14. Lotker, Z., Patt-Shamir, B., Rawitz, D.: Rent, lease, or buy: Randomized algorithms for multislope ski rental. SIAM J. Disc. Math. 26, 718–736 (2012)
15. Meyerson, A.: The parking permit problem. In: 46th Annu. IEEE Symp. Found. Comput. Sci. pp. 274–282 (2005)
16. Nagarajan, C., Williamson, D.P.: Offline and online facility leasing. Discrete Optim. 10(4), 361–370 (2013)
17. San Felice, M.C., Cheung, S.S., Lee, O., Williamson, D.P., Fernandes, C.G.: The online prize-collecting facility leasing problem. (to be published) (2016)
18. San Felice, M.C., Fernandes, C.G., Lintzmayer, C.N.: The online multicommodity connected facility location problem. To appear in: Approximation and Online Algorithms, LNCS (2017)
19. San Felice, M.C., Williamson, D.P., Lee, O.: A randomized $O(\log n)$-competitive algorithm for the online connected facility location. Algorithmica pp. 1–19 (2016)
20. Sviridenko, M.: An improved approximation algorithm for the metric uncapacitated facility location problem. In: Integer Programming and Combinatorial Optimization, LNCS, vol. 2337, pp. 240–257 (2002)
21. Swamy, C., Kumar, A.: Primal-dual algorithms for connected facility location problems. Algorithmica 40, 245–269 (2004)
22. Umboh, S.: Online network design algorithms via hierarchical decompositions. In: Proc. 26th Annu. ACM-SIAM Symp. Disc. Algorithms. pp. 1373–1387 (2015)