

Article

Online Facility Location in Evolving Metrics

Dimitris Fotakis ^{1,†} , Loukas Kavouras ^{1,*,†} and Lydia Zakyntinou ^{2,†} 
¹ School of Electrical and Computer Engineering, National Technical University of Athens, 15780 Athens, Greece; fotakis@cs.ntua.gr

² Khoury College of Computer Science, Northeastern University, Boston, MA 02115, USA; zakyntinou.l@northeastern.edu

* Correspondence: lkavouras@corelab.ntua.gr; Tel.: +30-210-772-1550

† These authors contributed equally to this work.

Abstract: The Dynamic Facility Location problem is a generalization of the classic Facility Location problem, in which the distance metric between clients and facilities changes over time. Such metrics that develop as a function of time are usually called “evolving metrics”, thus Dynamic Facility Location can be alternatively interpreted as a Facility Location problem in evolving metrics. The objective in this time-dependent variant is to balance the trade-off between optimizing the classic objective function and the *stability* of the solution, which is modeled by charging a switching cost when a client’s assignment changes from one facility to another. In this paper, we study the online variant of Dynamic Facility Location. We present a randomized $O(\log m + \log n)$ -competitive algorithm, where m is the number of facilities and n is the number of clients. In the first step, our algorithm produces a fractional solution, in each timestep, to the objective of Dynamic Facility Location involving a regularization function. This step is an adaptation of the generic algorithm proposed by Buchbinder et al. in their work “Competitive Analysis via Regularization.” Then, our algorithm rounds the fractional solution of this timestep to an integral one with the use of exponential clocks. We complement our result by proving a lower bound of $\Omega(m)$ for deterministic algorithms and lower bound of $\Omega(\log m)$ for randomized algorithms. To the best of our knowledge, these are the first results for the online variant of the Dynamic Facility Location problem.

Keywords: dynamic facility location; online convex optimization; competitive analysis



Citation: Fotakis, D.; Kavouras, L.; Zakyntinou, L. Online Facility Location in Evolving Metrics. *Algorithms* **2021**, *14*, 73. <https://doi.org/10.3390/a14030073>

Academic Editor: Frank Werner

Received: 18 January 2021

Accepted: 22 February 2021

Published: 25 February 2021

Publisher’s Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Facility Location problem is an extensively studied combinatorial optimization problem, which has many practical applications. In this problem, we are given a single metric on a set of clients and facilities, where each facility is associated with an opening cost. The goal is to find a set of facility locations that minimize the opening cost for all facilities plus the connection cost for all clients, where a client’s connection cost is its distance to the nearest facility.

In many natural location and network design settings, client locations are not known in advance. Motivated by this fact, Ref. [1] introduced online facility location problems, where clients arrive one-by-one and must be irrevocably assigned to a facility upon arrival. In practical settings related to online data clustering, new data points arrive and the decision of clustering some data points together should not be regarded as irrevocable [2].

Understanding the dynamics of temporally evolving social or infrastructure networks has been the central question in many applied areas recently. Eisenstat et al. [3] introduced the Dynamic Facility Location problem, which models the temporal aspects of such networks. In this time-dependent variant of Facility Location problem, clients or facilities may change their location over time and the goal is to achieve the best trade-off between the optimal connections of clients to facilities and the stability of solutions between consecutive timesteps. The temporal aspect of the Dynamic Facility Location problem is modeled by

T metrics given on the same set of clients and facilities, each representing the metric at time step $t \in \{1, \dots, T\}$. This is the key difference between the classic Facility Location problem, which is solved in a fixed metric space M that does not develop over time. Therefore, the Dynamic Facility Location is a Facility Location problem in an evolving metric $M(t) = \{M_1, \dots, M_T\}$, which changes at each timestep $t \in \{1, \dots, T\}$.

In this paper, we study the online variant of the Dynamic Facility Location problem, denoted as ODFL, where the metrics on clients and facilities are revealed one by one at each round. The online algorithm must make its decision before the metric of the next round is revealed and without knowing the total number of rounds. This is the key difference between ODFL and the offline variant of Dynamic Facility Location in [3], where the T metrics are known beforehand. Therefore, ODFL attempts to capture realistic settings, where the input data are revealed piece-by-piece and the algorithm must make its decision before upcoming input pieces are revealed. The online aspect of the problem poses new challenges and precludes many algorithmic techniques used to deal with offline problems. Next, we give a formal definition of ODFL.

Model. In Online Dynamic Facility Location, we are given a set of facilities F , $|F| = m$, a set of clients C , $|C| = n$, a switching cost g and a facility opening cost f . At each round $t \in \{1, \dots, T\}$, a new metric between clients and facilities is revealed with the form of a $n \times m$ dimensional vector d_t , which has entries corresponding to distances over $F \times C$. We denote by $d_t(i, j)$ the distance between client j and facility i at time t . At each round t , the goal is to find a subset $A_t \subseteq F$ of open facilities and an assignment $\phi_t : C \rightarrow A_t$ of clients to open facilities so as to minimize the objective:

$$f \sum_{t=1}^T |A_t| + \sum_{t=1, j \in C}^T d_t(\phi_t(j), j) + g \cdot \sum_{t=1, j \in C}^T \mathbb{1}\{\phi_t(j) \neq \phi_{t-1}(j)\}$$

where $\mathbb{1}\{p\}$ is the indicator function of proposition p . The assignment ϕ_t of round t is chosen without knowing the distance vectors $d_{t+1}, \dots, d_T$ of upcoming rounds. The objective function is the sum of the hourly opening costs for each open facility plus the connection costs of each client plus the switching costs (g per change of facility per client). We note that any solution pays switching cost gn at round $t = 1$, since it switches to an initial assignment of the clients to facilities.

We measure the performance of our online algorithm using the notion of the competitive ratio. Given a request sequence σ , let $\text{ALG}(\sigma)$ denote the cost paid by an online algorithm on σ and let $\text{OPT}(\sigma)$ denote the cost paid on σ by an optimal offline algorithm, which knows σ in advance. The online algorithm is c -competitive if there exists a constant a such that:

$$\text{ALG}(\sigma) \leq c \cdot \text{OPT}(\sigma) + a$$

for all request sequences σ . The factor c is called the competitive ratio.

Related work. The offline and online variants of Facility Location have been studied extensively in the literature. For the offline Facility Location problem, the approximability is $\Theta(\log n)$ [4] for the non-metric case while, for the metric case, the best lower bound is 1.463 [5], and the best algorithm has an approximation ratio 1.488 [6]. Online Facility Location is known to have a competitive ratio of $\Theta(\log n / \log \log n)$ in the adversarial case for both deterministic and randomized algorithms [7] and a constant competitive ratio if the clients are drawn from a known distribution [8].

The study of Dynamic Facility Location so far concerns the offline case where the changes between distances of clients and facilities are known in advance. Eisenstat et al. [3] showed an upper bound of $O(\log nT)$ for the most interesting variant of Dynamic Facility Location with hourly facility costs, where facilities can be closed and are paid for all rounds in which they remain open. This result was later improved by [9], which gave an $O(1)$ -approximation algorithm.

In [10], they provided a framework for designing competitive online algorithms using regularization, which is a widely used technique in online learning. They designed a $O(\log m)$ -competitive deterministic algorithm for generating a fractional solution that satisfies a time-varying set of constraints, where m is the number of variables. Then, they provided an $O(\log m \log n)$ -competitive randomized algorithm for the online set cover problem with a service cost, where m is the number of sets and n is the number of elements. The first step of our online algorithm, which provides a $O(\log m)$ -competitive fractional solution, where m is the number of facilities, is inspired by the approach of [10] and a large part of the analysis follows their proof. In the second step of our online algorithm, we show a rounding scheme that works favorably with the fractional solution to obtain a non-trivial additive competitive ratio of $O(\log m + \log n)$, where n is the number clients. To the best of knowledge, this is the first upper bound for ODFL.

2. Results

In this work, we study the competitive ratio of ODFL. We start by proving lower bounds for deterministic and randomized algorithms for ODFL. Our first result is the following.

Theorem 1. *The competitive ratio of any deterministic online algorithm is $\Omega(m)$ and the competitive ratio of any randomized online algorithm against the oblivious adversary is $\Omega(\log m)$ for the Online Dynamic Facility Location problem, where m is the number of facilities.*

Our second result is a randomized algorithm, which is $O(\log m + \log n)$ -competitive. In order to achieve this, we express the offline Dynamic Facility Location problem as a linear program P (Figure 1a). Then, we apply the following two algorithms at each round t .

1. **Algorithm 1 (Regularization algorithm):** It solves a linear program minimizing the objective function of P modified to include a smooth convex regularization term and obtains the fractional solution $Sol(t)$.
2. **Algorithm 2 (Rounding algorithm):** It rounds the fractional solution $Sol(t)$ of Algorithm 1 to an integral solution using competing exponential clocks.

Algorithm 1 The regularization algorithm

Parameters: $\epsilon > 0, \eta = \ln(1 + n/\epsilon)$

Initialization: Set $y_i^0 = 0 \forall i \in [m]$ and $x_{ij}^0 = 0 \forall i \in [m], j \in [n]$.

At each round t : Let $d_t \in \mathbb{R}_+^{m \times n}$ be the distance cost vector and let S be the set of feasible solutions. Solve the following linear program (P^*) to obtain the fractional solution (y^t, x^t) :

$$(y^t, x^t) = \arg \min_{(y, x) \in S} \left\{ f \sum_{i=1}^m y_i + \sum_{j=1}^n \sum_{i=1}^m x_{ij} \cdot d_t(i, j) + \frac{1}{\eta} \sum_{j=1}^n \sum_{i=1}^m \left[\left(\left(x_{ij} + \frac{\epsilon}{n} \right) \ln \frac{x_{ij} + \frac{\epsilon}{n}}{x_{ij}^{t-1} + \frac{\epsilon}{n}} \right) - x_{ij} \right] \right\}$$

Algorithm 2 The rounding algorithm

1: **Initialization:** Choose i.i.d. random variables $Z_{ij} \sim \exp(1), \forall i, j$.

2: **At each round t :**

3: Let x_{ij}^t be the fractional value of round t obtained by Algorithm 1.

4: For each client j , open $i = \arg \min_{i'} \frac{Z_{i'j}}{x_{i'j}^t}$ and connect j to i .

Algorithm 1 solves online a linear program to produce a fractional solution at each round t involving the current distance vector d_t . This algorithm is essentially the general

algorithm presented in [10], which we adapt to ODFL. The performance of the general regularization algorithm is proved by Theorem 1.1 in [10] for the case of time varying covering constraints. Although we follow the same steps to prove the existence of a $O(\log m)$ -competitive fractional solution for ODFL, where m is the number of facilities, we must also address the presence of both covering and precedence constraints in ODFL.

Algorithm 2 is the randomized procedure that rounds the fractional solution provided from Algorithm 1 to an integral solution. Our contribution here is that we use an appropriate rounding which works favorably with Algorithm 1 so as to produce a solution, which is $O(\log m + \log n)$ -competitive for ODFL. The rounding algorithm makes use of competing exponential clocks, which have been applied in many similar problems like the Dynamic Facility Location problem [9] and the Online Set Cover with a Service Cost problem [10].

Theorem 2. *There is a randomized algorithm which is $O(\log m + \log n)$ -competitive for the Online Dynamic Facility Location problem, where m denotes the number of facility locations and n denotes the number of clients.*

Organization. In Section 3, we present lower bounds on the competitive ratio of deterministic and randomized algorithms for ODFL. Then, we present Algorithm 1 in Section 4 and Algorithm 2 in Section 5 and prove their guarantees in the respective sections.

$$\begin{array}{ll}
 \min & f \sum_{t=1}^T \sum_{i=1}^m y_i^t + \sum_{t=1}^T \sum_{j=1}^n \sum_{i=1}^m x_{ij}^t \cdot d_t(i, j) + g \sum_{t=1}^T \sum_{j=1}^n \sum_{i=1}^m z_{ij}^t \\
 \text{s.t.} & x_{ij}^t \leq y_i^t \quad \forall t \in [T], \forall i \in [m], \forall j \in [n] \\
 & \sum_{i=1}^m x_{ij}^t \geq 1 \quad \forall t \in [T], \forall j \in [n] \\
 & z_{ij}^t \geq x_{ij}^t - x_{ij}^{t-1} \quad \forall t \in [T], \forall i \in [m], \forall j \in [n] \\
 & y_i^t \geq 0, x_{ij}^t \geq 0, z_{ij}^t \geq 0 \quad \forall t \in [T], \forall i \in [m], \forall j \in [n]
 \end{array}$$

(a) The linear program P for offline Dynamic Facility Location.

$$\begin{array}{ll}
 \max & \sum_{t=1}^T \sum_{j=1}^n a_j^t \\
 \text{s.t.} & \sum_{j=1}^n e_{ij}^t \leq f \quad \forall t \in [T], \forall i \in [m] \\
 & b_{ij}^t \leq g \quad \forall t \in [T], \forall i \in [m], \forall j \in [n] \\
 & b_{ij}^{t+1} - b_{ij}^t \leq d_t(i, j) + e_{ij}^t - a_j^t \quad \forall t \in [T], \forall i \in [m], \forall j \in [n] \\
 & b_{ij}^t \geq 0, e_{ij}^t \geq 0, a_j^t \geq 0 \quad \forall t \in [T], \forall i \in [m], \forall j \in [n]
 \end{array}$$

(b) The dual D of the linear program P.

Figure 1. (a) The linear program P for offline Dynamic Facility Location. (b) The dual D of the linear program P.

3. Lower Bounds

In this section, we prove lower bounds of deterministic and randomized algorithms for ODFL. In both cases, the metric space is a star graph with a client lying on the center of the star for all rounds.

The core idea of the proofs is to force the online algorithm to pay the switching cost at each round. By carefully selecting the parameters of ODFL, we can prove that any deterministic online algorithm is $O(m)$ -competitive. For the randomized lower bound, we use Yao's principle (see examples in Chapter 8 in [11]). Specifically, we choose a randomized instance such that the expected performance of any deterministic algorithm against the optimal offline algorithm is $\Omega(\log m)$. By Yao's principle, any randomized algorithm has the same lower bound.

Proof. Let OPT denote the optimal cost and ALG denote the cost of an online algorithm. The instance consists of a star graph with m edges and the number of rounds is $T = m$. Facilities can only be opened in the leaves (a total of m leaves), and there is one client ($n = 1$) sitting at the center of a graph for all rounds. The distance of every leaf j to the

center is initially $d_j = d$. Then, the adversary has the following simple strategy at each round $1 \leq t \leq T - 1$:

For every leaf j , such that the online algorithm connects the client to the facility in j , d_j becomes arbitrarily large. At round T , the distances remain the same as in the previous round.

Observe that there is only one leaf with distance d from the center of a star for all rounds. The optimal offline solution just opens a facility at this leaf and connects the client to it for all rounds, thus paying $g + Tf + Td$. On the other side, any competitive online algorithm will prefer to open a new facility at distance d and connect the client to this facility at the start of each round instead of paying the large distance. Therefore, the cost incurred by any online algorithm is at least $Tg + Tf + Td$. By setting $g \gg d = f$, we have that

$$\frac{\text{ALG}}{\text{OPT}} \geq \frac{Tg + Tf + Td}{g + Tf + Td} = \Omega(T) = \Omega(m)$$

Turning to the randomized case, the instance consists of the same metric as the deterministic case and the only difference is that we will use randomized adversarial requests. Then, by showing that any deterministic algorithm has a competitive ratio of at least $\log m$ and by Yao's principle, we will prove the lower bound for randomized algorithms.

Now, the adversary chooses uniformly at random an edge e , which has length d (has not yet become arbitrarily large) at each round $1 \leq t \leq T - 1$ and makes its length arbitrarily large. At round T , where only one leaf has distance d , the distances remain the same as in the previous round. Again, the optimal solution uses the leaf in distance d at all rounds and pays $g + Tf + Td$. However, the expected switching cost of any competitive algorithm is:

$$\mathbb{E}[\text{switching cost}] = g + \sum_{t=1}^{T-1} \Pr[\text{switches at round } t] \cdot g = g + \sum_{t=1}^{T-1} \frac{1}{m - t + 1} \cdot g > H_T \cdot g$$

since at each round t the edge that the algorithm uses becomes arbitrarily large with probability $1/(m - t + 1)$. By setting $g > Td = Tf$,

$$\frac{\mathbb{E}[\text{ALG}]}{\text{OPT}} \geq \frac{g \cdot H_T + Tf + Td}{g + Tf + Td} = \Omega(\log T) = \Omega(\log m).$$

□

This concludes this section with the lower bounds. In the following sections, we present a randomized algorithm for the ODFL problem with a nearly matching bound of $O(\log m + \log n)$.

4. The Regularization Algorithm

In this section, we show that the regularization algorithm of [10] can be applied to ODFL and that it produces a fractional solution at each round, which is $O(\log m)$ -competitive, where m is the number of facility locations. We will prove the following theorem:

Theorem 3. *The Regularization Algorithm produces an $O(\log m)$ -competitive fractional solution for the Online Dynamic Facility Location problem, where m is the number of facilities.*

Before proceeding to the details of Algorithm 1, we first express the offline Dynamic Facility Location as a linear program, denoted as P (Figure 1a). Algorithm 1 will solve a linear program P^* at each round, which will be constructed from P combined with a regularization function. Finally, we will show that the fractional solution of P^* is $O(\log m)$ -

competitive with respect to the solution of the dual program D (Figure 1b) of P, which serves as lower bound on the optimal offline solution.

Now, we express offline Dynamic Facility Location as an integer program, which will be relaxed to obtain the linear program P. Recall that T, n, m are the number of rounds, clients, and facility locations, respectively. The first term of the objective function is the total facility opening cost, where f is the cost to open a facility. The second term is the total connection cost, where $d_t(i, j)$ is the distance between facility i and client j in round t , and the third term is the total switching cost, where each change of a client's connection to a facility costs g .

We use the decision variables y_i^t , x_{ij}^t and z_{ij}^t , where $i \in [m], j \in [n], t \in [T]$; $y_i^t = 1$ if facility i is open at round t and $y_i^t = 0$ otherwise, $x_{ij}^t = 1$ if client j is connected to facility i at round t and $x_{ij}^t = 0$, otherwise, $z_{ij}^t = 1$ if client j was connected to facility i at round t but not connected to the same facility i at round $t - 1$ and $z_{ij}^t = 0$, otherwise. The value of the variable z_{ij}^t is imposed from the third constraint, which expresses the switching cost. The first constraint ($x_{ij}^t \leq y_i^t$) ensures that, whenever a client j is connected to a facility i , the facility i is open. The second constraint ($\sum_{i=1}^m x_{ij}^t \geq 1$) guarantees that every client is connected to a facility. Finally, relaxing the decision variables to take non-negative real values, we obtain the LP of Figure 1a, denoted as P.

Next, we are ready to present Algorithm 1. The algorithm is given at each round t a distance vector $d_t \in \mathbb{R}_+^{m \times n}$ containing the distances between clients and facilities. Then, Algorithm 1 finds the minimizer (y^t, x^t) of the linear program P^* at each round t , which has two differences from P. The first one is that the last term of the objective function in P (the switching cost) is substituted by the regularization function in P^* . The second is that the constraint relative to the switching cost ($z_{ij}^t \geq x_{ij}^t - x_{ij}^{t-1}$) in P is omitted in P^* . We note that the regularized objective function includes both the previous solution as well as the current cost vector. Thus, the solution in each round is determined greedily and independently of rounds prior to $t - 1$.

To analyze the performance of Algorithm 1, we will need to construct a lower bound on the optimal offline solution. Therefore, we derive the dual D of P (Figure 1b), which has the following variables (corresponding to the primal constraints on the left):

- $x_{ij}^t \leq y_i^t \rightarrow e_{ij}^t$ for all $t \in [T], i \in [m], j \in [n]$
- $\sum_{i=1}^m x_{ij}^t \geq 1 \rightarrow a_j^t$ for all $t \in [T], j \in [n]$
- $z_{ij}^t \geq x_{ij}^t - x_{ij}^{t-1} \rightarrow b_{ij}^t$ for all $t \in [T], i \in [m], j \in [n]$

We will prove Theorem 3 by showing that the set of dual variables of the solutions that P^* returns is a feasible solution for D within a factor of $(1 + (1 + \epsilon') \ln(1 + \frac{m}{\epsilon'}))$ of the optimal offline solution, where ϵ' is a small constant. Specifically, we will use the KKT optimality conditions of P^* (the regularized LP) in each round. The constraints define dual variables, which will be plugged in the formulation of the dual D in Figure 1b. This way, we will construct a dual solution to the original online problem, which will serve as a lower bound on the optimal offline solution. Ref. [10] mentions that their technique can be generalized to facility location problems, without providing any further technical details. In the next lemmas, we verify their claim, by adjusting their approach and proof techniques to ODFL. Recall that the constraint $z_{ij}^t \geq x_{ij}^t - x_{ij}^{t-1}$ is omitted in P^* . In order to define a feasible solution for D, we introduce the variable b_{ij}^t corresponding to this constraint and we let e_{ij}^*, a_j^* be the optimal dual variables of D^* corresponding to the precedence and covering constraints, respectively.

Lemma 1. *The set of optimal solutions for each round t of the dual LP D^* of P^* ($a^{*,t}, e^{*,t}$), which satisfy the KKT conditions for an appropriate b_{ij}^t , consist of a feasible solution for D.*

Proof. Let $x^{*,t}$ be the optimal solution of P^* at round t . Set the variables of D at time t to be:

$$a_j^t = a_j^{*,t}, e_{ij}^t = e_{ij}^{*,t} \text{ and } b_{ij}^{t+1} = \frac{g}{\eta} \ln \frac{1 + \frac{\epsilon}{n}}{x_{ij}^{*,t} + \frac{\epsilon}{n}}$$

To prove that the solution above is feasible for D, we prove that it satisfies its constraints one by one. This is achieved using the following KKT conditions that hold for P* and its dual:

$$a_j^* \geq 0, \forall j \in [n] \quad (1)$$

$$e_{ij}^* \geq 0, \forall i \in [m], \forall j \in [n] \quad (2)$$

$$f - \sum_{j=1}^n e_{ij}^* \geq 0, \forall i \in [m] \quad (3)$$

$$d_t(i, j) + \frac{g}{\eta} \ln \frac{x_{ij}^* + \frac{\epsilon}{n}}{x_{ij}^{t-1} + \frac{\epsilon}{n}} + e_{ij}^* - a_j^* \geq 0, \forall i \in [m], \forall j \in [n] \quad (4)$$

The first group of constraints of the dual D (Figure 1b) ($\sum_{j=1}^n e_{ij}^t \leq f$) follows easily from KKT condition (3). The same holds for the last two groups of constraints ($e_{ij}^t \geq 0$ and $a_j^t \geq 0$) due to KKT conditions (1) and (2). Furthermore, by (4) and the construction of b_{ij}^t , we have that:

- $b_{ij}^t = \frac{g}{\eta} \ln \frac{1 + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} = \frac{g}{\ln(1 + \frac{\epsilon}{n})} \ln \frac{1 + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \stackrel{x^{*,t} \geq 0}{\leq} \frac{g}{\ln(1 + \frac{\epsilon}{n})} \ln \frac{1 + \frac{\epsilon}{n}}{\frac{\epsilon}{n}} = \frac{g}{\ln(\frac{n}{\epsilon} + 1)} \ln(\frac{n}{\epsilon} + 1) = g$
- $b_{ij}^{t+1} - b_{ij}^t = -\frac{g}{\eta} \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \stackrel{(4)}{\leq} d_t(i, j) + e_{ij}^t - a_j^t.$
- $b_{ij}^t = \frac{g}{\eta} \ln \frac{1 + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \stackrel{x^{*,t} \leq 1}{\geq} \frac{g}{\eta} \ln \frac{1 + \frac{\epsilon}{n}}{1 + \frac{\epsilon}{n}} = 0$

The above inequalities prove that the second, third, and fourth group of constraints of D also hold, thus completing the proof of the lemma. $\square$

We are now ready to prove Theorem 3, by showing that the dual we constructed can pay for the facility, connection, and switching cost of Algorithm 1. Since we bound together the facility cost and connection cost, we will simply refer to them as the service cost. Throughout the proofs, we will use the following relations:

$$a_j^*(1 - \sum_{i=1}^m x_{ij}^*) = 0, \forall j \in [n] \quad (5)$$

$$y_i^*(f - \sum_{j=1}^n e_{ij}^*) = 0, \forall i \in [m] \quad (6)$$

$$x_{ij}^*(d_t(i, j) + \frac{g}{\eta} \ln \frac{x_{ij}^* + \frac{\epsilon}{n}}{x_{ij}^{t-1} + \frac{\epsilon}{n}} + e_{ij}^* - a_j^*) = 0, \forall i \in [m], \forall j \in [m] \quad (7)$$

$$e_{ij}^*(x_{ij}^* - y_i^*) = 0, \forall j \in [n], \forall i \in [m] \quad (8)$$

$$h - k \leq h \ln(h/k) \text{ for any } h, k > 0 \quad (9)$$

$$\sum_i h_i \ln(h_i/k_i) \leq \left(\sum_i h_i \right) \log \frac{\sum_i h_i}{\sum_i k_i} \quad (10)$$

Equalities (5)–(8) are the KKT conditions of P* and its dual and the remaining two inequalities are standard logarithmic inequalities. Theorem 3 will follow from the next two lemmas. The analysis is similar to that of Theorem 1.1 in [10] adapted to the objective of ODFL and also dealing with the presence of precedence constraints.

Lemma 2. The switching cost M of Algorithm 1 is at most $\eta(1 + \frac{\epsilon m}{n})$ times the cost of the dual feasible solution of Lemma 1.

Proof. Let M_t be the switching cost of Algorithm 1 at round t . The summation below is taken over increasing values of connection variables, i.e., $x_{ij}^{*,t} > x_{ij}^{*,t-1}$, since decreasing values only decrease the fractional switching cost:

$$\begin{aligned}
 M_t &= g \sum (x_{ij}^{*,t} - x_{ij}^{*,t-1}) = \eta \cdot \frac{g}{\eta} \sum (x_{ij}^{*,t} - x_{ij}^{*,t-1}) \\
 &= \eta \cdot \frac{g}{\eta} \sum \left(x_{ij}^{*,t} + \frac{\epsilon}{n} - \left(x_{ij}^{*,t-1} + \frac{\epsilon}{n} \right) \right) \\
 &\leq \eta \sum \left(x_{ij}^{*,t} + \frac{\epsilon}{n} \right) \frac{g}{\eta} \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \quad (\text{by inequality (9)}) \\
 &\leq \eta \sum_{i=1}^m \sum_{j=1}^n \left(x_{ij}^{*,t} + \frac{\epsilon}{n} \right) (a_j^{*,t} - e_{ij}^{*,t} - d_t(i, j)) \quad (\text{by (7)}) \\
 &\leq \eta \sum_{j=1}^n a_j^{*,t} \left(\sum_{i=1}^m \left(x_{ij}^{*,t} + \frac{\epsilon}{n} \right) \right) = \eta \sum_{j=1}^n a_j^{*,t} \left(\sum_{i=1}^m x_{ij}^{*,t} + \frac{\epsilon m}{n} \right) \\
 &= \eta \sum_{j=1}^n a_j^{*,t} \left(1 + \frac{\epsilon m}{n} \right) = \eta \left(1 + \frac{\epsilon m}{n} \right) \sum_{j=1}^n a_j^{*,t} \quad (\text{by (5)})
 \end{aligned}$$

Hence,

$$M = \sum_{t=1}^T M_t \leq \eta \left(1 + \frac{\epsilon m}{n} \right) \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t} \quad (11)$$

This concludes the proof of the lemma. $\square$

Lemma 3. The total service cost S of Algorithm 1 is less than the cost of the dual feasible solution of Lemma 1:

Proof.

$$\begin{aligned}
 S &= \sum_{t=1}^T \left[f \sum_{i=1}^m y_i^{*,t} + \sum_{j=1}^n \sum_{i=1}^m x_{ij}^{*,t} d_t(i, j) \right] \\
 &= \sum_{t=1}^T \left[\sum_{i=1}^m y_i^{*,t} \left(\sum_{j=1}^n e_{ij}^{*,t} \right) + \sum_{j=1}^n \sum_{i=1}^m x_{ij}^{*,t} \left(a_j^{*,t} - e_{ij}^{*,t} - \frac{g}{\eta} \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \right) \right] \quad (\text{by (7) and (6)}) \\
 &= \sum_{t=1}^T \left[\sum_{i=1}^m \sum_{j=1}^n (y_i^{*,t} - x_{ij}^{*,t}) e_{ij}^{*,t} + \sum_{j=1}^n \sum_{i=1}^m x_{ij}^{*,t} a_j^{*,t} - \frac{g}{\eta} \sum_{j=1}^n \sum_{i=1}^m x_{ij}^{*,t} \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \right] \\
 &= \sum_{t=1}^T \left[\sum_{j=1}^n a_j^{*,t} \left(\sum_{i=1}^m x_{ij}^{*,t} \right) - \frac{g}{\eta} \sum_{j=1}^n \sum_{i=1}^m x_{ij}^{*,t} \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \right] \quad (\text{by (8)}) \\
 &= \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t} - \frac{g}{\eta} \sum_{j=1}^n \sum_{i=1}^m \left[\sum_{t=1}^T \left(x_{ij}^{*,t} + \frac{\epsilon}{n} \right) \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} - \frac{\epsilon}{n} \sum_{t=1}^T \ln \frac{x_{ij}^{*,t} + \frac{\epsilon}{n}}{x_{ij}^{*,t-1} + \frac{\epsilon}{n}} \right] \quad (\text{by (5)}) \\
 &\leq \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t} - \frac{g}{\eta} \sum_{j=1}^n \sum_{i=1}^m \left[\sum_{t=1}^T \left(x_{ij}^{*,t} + \frac{\epsilon}{n} \right) \ln \frac{\sum_{t=1}^T (x_{ij}^{*,t} + \frac{\epsilon}{n})}{\sum_{t=1}^T (x_{ij}^{*,t-1} + \frac{\epsilon}{n})} - \frac{\epsilon}{n} \ln \frac{x_{ij}^{*,T} + \frac{\epsilon}{n}}{x_{ij}^{*,0} + \frac{\epsilon}{n}} \right] \quad (\text{by (10)})
 \end{aligned}$$

Notice that the two terms in the bracket of the right-hand side of the inequality above cancel each other out, since:

$$-\frac{\epsilon}{n} \ln \frac{x_{ij}^{*,T} + \frac{\epsilon}{n}}{x_{ij}^{*,0} + \frac{\epsilon}{n}} \stackrel{x_{ij}^{*,0}=0}{=} \left(x_{ij}^{*,0} + \frac{\epsilon}{n}\right) \ln \frac{x_{ij}^{*,0} + \frac{\epsilon}{n}}{x_{ij}^{*,T} + \frac{\epsilon}{n}} \stackrel{(9)}{\geq} x_{ij}^{*,0} - x_{ij}^{*,T}$$

$$\left(\sum_{t=1}^T (x_{ij}^{*,t} + \frac{\epsilon}{n})\right) \ln \frac{\sum_{t=1}^T (x_{ij}^{*,t} + \frac{\epsilon}{n})}{\sum_{t=1}^T (x_{ij}^{*,t-1} + \frac{\epsilon}{n})} \stackrel{(9)}{\geq} \sum_{t=1}^T \left(x_{ij}^{*,t} + \frac{\epsilon}{n}\right) - \sum_{t=1}^T \left(x_{ij}^{*,t-1} + \frac{\epsilon}{n}\right) = x_{ij}^{*,0} - x_{ij}^{*,T}$$

Therefore, it holds that

$$S \leq \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t}$$

□

We can now easily prove the performance of Algorithm 1 stated in Theorem 3.

Proof of Theorem 3. Let OPT(D) and OPT(P) denote the optimal solutions of the D and P, respectively. By Lemmas 2 and 3, the total cost of Algorithm 1 is:

$$\begin{aligned} S + M &\leq \left[1 + \eta \left(1 + \frac{\epsilon m}{n}\right)\right] \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t} \\ &= \left[1 + \ln \left(1 + \frac{n}{\epsilon}\right) \left(1 + \frac{\epsilon m}{n}\right)\right] \sum_{t=1}^T \sum_{j=1}^n a_j^{*,t} \\ &\leq \left[1 + \ln \left(1 + \frac{n}{\epsilon}\right) \left(1 + \frac{\epsilon m}{n}\right)\right] \text{OPT(D)} && \text{(by Lemma 1)} \\ &= \left[1 + (1 + \epsilon') \ln \left(1 + \frac{m}{\epsilon'}\right)\right] \text{OPT(D)} && \text{(since } \epsilon' = \frac{\epsilon m}{n} \text{)} \\ &= \left[1 + (1 + \epsilon') \ln \left(1 + \frac{m}{\epsilon'}\right)\right] \text{OPT(P)} \end{aligned}$$

□

The proof of Theorem 3 concludes this section.

5. The Rounding Algorithm

In this section, we present Algorithm 2, which makes use of the exponential distribution to round the fractional solution to an integral solution at each round. The analysis shows that the fractional solution grows up to a factor logarithmic in n regarding the facility cost and up to constant factors regarding the switching and connection cost. Before proceeding to the details of Algorithm 2, we give the definition of an exponential random variable and some of their properties.

Definition 1. A random variable X is distributed according to the exponential distribution with rate λ , denoted as $X \sim \exp(\lambda)$, if it has density $f_X(x) = \lambda e^{-\lambda x}$ for every $x \geq 0$, and $f_X(x) = 0$ otherwise. We will use the following properties of exponential random variables:

1. If $X \sim \exp(\lambda)$ and $c > 0$, then $X/c \sim \exp(\lambda c)$.
2. Let $X_1, \dots, X_k$ be independent random variables with $X_i \sim \exp(\lambda_i)$:
 - (a) $\min\{X_1, \dots, X_k\} \sim \exp(\lambda_1 + \dots + \lambda_k)$
 - (b) $\Pr[X_i \leq \min_{j \neq i} X_j] = \frac{\lambda_i}{\lambda_1 + \dots + \lambda_k}$
3. If $X \sim \exp(\lambda)$ and $Y \sim \exp(\mu)$ are independent, then $\forall t \geq 0$:

$$\Pr[X \leq Y \mid X \geq t] = \frac{\lambda}{\lambda + \mu} \cdot e^{\mu t}$$

The rounding algorithm samples independently a total of $n \cdot m$ (one for each client-facility connection) random variables Z_{ij} from the exponential distribution with rate $\lambda = 1$ at the beginning of its execution, which will be used throughout all rounds. Then, at each round t , it chooses for each client j the connection $\{i, j\}$ minimizing the ratio $\frac{Z_{ij}}{x_{ij}^t}$, where x_{ij}^t is the fractional variable of this connection obtained by Algorithm 1. Notice that, by the properties of Definition 1, the ratio $\frac{Z_{ij}}{x_{ij}^t}$ is also an exponential random variable. This technique is referred to as competing exponential clocks, since a random variable wins the competition if it has the smallest value among all others (minimizes the ratio $\frac{Z_{ij}}{x_{ij}^t}$ in our case).

The high level idea of the analysis is that connection and switching cost of the rounded solution add only constant factors to the cost of the connection and switching cost of the fractional solution at each round t . The reason is that they favor connections to facilities that are dependent on the increase/decrease of the fractional variables x_{ij}^t . This fact combined with the properties of the exponential distribution leads to a rounding of the *right* connections indicated by the fractional solution. On the other side, this leads to more open facilities, since we prove that the rounding adds a factor logarithmic in n to the cost of the fractional solution.

Next, we will analyze the performance of Algorithm 2 by bounding separately the facility, connection, and switching cost. We will simply calculate the probabilities of opening any facility, connecting a client to a facility and changing a connection.

Facility cost . We start with the facility cost of the rounding algorithm, which is $O(\log n)$ -competitive with respect to the facility cost of the fractional solution.

Proof. Let E_{ij} denote the event that $i = \arg \min_{i'} \frac{Z_{i'j}}{x_{i'j}^t}$ for some client j and let $a > 0$ be chosen later. The probability of E_{ij} equals:

$$\begin{aligned}
 \Pr[\exists j : E_{ij}] &= \Pr\left[\exists j : E_{ij} \mid \frac{Z_{ij}}{x_{ij}} < a\right] \cdot \Pr\left[\frac{Z_{ij}}{x_{ij}} < a\right] + \Pr\left[\exists j : E_{ij} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] \cdot \Pr\left[\frac{Z_{ij}}{x_{ij}} \geq a\right] \\
 &\leq \Pr\left[\frac{Z_{ij}}{x_{ij}} < a\right] + \Pr\left[\exists j : E_{ij} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] \cdot \Pr\left[\frac{Z_{ij}}{x_{ij}} \geq a\right] \\
 &\leq \Pr\left[\frac{Z_{ij}}{x_{ij}} < a\right] + \sum_{j=1}^n \Pr\left[E_{ij} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] \cdot \Pr\left[\frac{Z_{ij}}{x_{ij}} \geq a\right] \quad (\text{By the union bound}) \\
 &= 1 - e^{-ax_{ij}} + \sum_{j=1}^n \Pr\left[E_{ij} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] e^{-ax_{ij}} \\
 &\leq a x_{ij} + \sum_{j=1}^n \frac{x_{ij}}{\sum_{i=1}^m x_{ij}} e^{-a(\sum_{i' \neq i} x_{i'j} - x_{ij})} e^{-ax_{ij}} \quad (1 - e^{-x} \leq x, \forall x \text{ and by Definition 1}) \\
 &\leq a x_{ij} + \sum_{j=1}^n e^{-a} x_{ij} \leq a y_i + n e^{-a} y_i. \quad (\text{since } x_{ij} \leq y_i)
 \end{aligned}$$

By choosing $a = \log n$, we have the result. $\square$

Connection cost. Next, we show that the connection cost of the rounding algorithm is $O(1)$ -competitive with the connection cost of the fractional solution. Again, let $a > 0$ be chosen later.

Proof. Similar arguments to the previous proof show that the probability to choose connection ij is:

$$\begin{aligned}
 \Pr[ij] &\leq \Pr\left[\frac{Z_{ij}}{x_{ij}} < a\right] + \Pr\left[\frac{Z_{ij}}{x_{ij}} = \min_{i'} \frac{Z_{i'j}}{x_{i'j}} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] \cdot \Pr\left[\frac{Z_{ij}}{x_{ij}} \geq a\right] \\
 &\leq 1 - e^{-ax_{ij}} + \Pr\left[\frac{Z_{ij}}{x_{ij}} = \min_{i'} \frac{Z_{i'j}}{x_{i'j}} \mid \frac{Z_{ij}}{x_{ij}} \geq a\right] e^{-ax_{ij}} \\
 &\quad (1 - e^{-x} \leq x, \forall x \text{ and by Definition 1}) \\
 &\leq a x_{ij} + \frac{x_{ij}}{\sum_{i=1}^m x_{ij}} e^{-a(\sum_{i' \neq i} x_{i'j} - x_{ij})} e^{-ax_{ij}} \\
 &\leq a x_{ij} + e^{-a} x_{ij}.
 \end{aligned}$$

By choosing a sufficiently small a (for example $a = 1$), we have the result. $\square$

Switching cost. Finally, we show that every step that incurs a fractional switching cost of d in a connection variable x_{ij} incurs an expected increase of at most $\frac{d}{d+1}$ in the randomized solution. Thus, the expected number of new connections is $O(1)$.

Proof. We break down the total movement from time $t - 1$ to t in the fractional solution into $m \times n$ intermediate steps, on each of which only the value of exactly one x_{ij} is changed. We take first all the x_{ij} 's whose value increases and then all the x_{ij} 's whose value decreases, thus managing to preserve a feasible solution in all the intermediate steps. This way, the total switching cost from time $t - 1$ to time t of the fractional solution does not change while the integral switching cost could only increase due to possible changes in the intermediate steps. First, we will prove the bound in the case the connection variable decreases: $x_{ij}^t = x_{ij}^{t-1} - d$.

Let $Y_{ij} = \min_{i' \neq i} \frac{Z_{i'j}}{x_{i'j}^t}$, where $Y_{ij} \sim \exp(\lambda)$ for $\lambda = 1 - x_{ij}^t$. When the value of x_{ij} decreases, the value of $\frac{Z_{ij}}{x_{ij}}$ increases. Therefore, connection ij cannot be chosen a time t if is not chosen at time $t - 1$. However, due to the increase of $\frac{Z_{ij}}{x_{ij}}$, another connection could turn minimal that had not been chosen in the previous time step. This is the only case, when a switching cost is incurred. The probability of this event is bounded by:

$$\Pr\left[\frac{Z_{ij}}{x_{ij}^{t-1}} \leq Y_{ij} \leq \frac{Z_{ij}}{x_{ij}^{t-1}-d}\right] = F_{Y_{ij}}[\infty] - F_{Y_{ij}}[0] = \frac{\lambda}{x_{ij}^{t-1}-d+\lambda} - \frac{\lambda}{x_{ij}^{t-1}+\lambda}$$

This expression is maximized when $x_{ij}^{t-1} - d = 0$, $\lambda = 1$, and therefore is less than

$$1 - \frac{1}{d+1} = \frac{d}{d+1}$$

Now, we turn to the case where $x_{ij}^t = x_{ij}^{t-1} + d$. When x_{ij} increases, the ratio $\frac{Z_{ij}}{x_{ij}}$ decreases. Therefore, if facility i was chosen in the previous step, it will be chosen again in this step, thus not incurring switching cost. If facility was not chosen in the previous step, it will be chosen in this step with probability

$$\Pr\left[\frac{Z_{ij}}{x_{ij}^{t-1}+d} \leq Y_{ij} \leq \frac{Z_{ij}}{x_{ij}^{t-1}}\right]$$

which is no more than $\frac{d}{d+1}$, following the exact same analysis with the case of the decreasing connection variables. $\square$

Finally, it is easy to provide the proof of Theorem 2, which concludes this section.

Proof of Theorem 2. First, notice that, by Lemma 3, the fractional solution of Algorithm 1 is optimal with respect to the facility and connection cost. Therefore, Algorithm 2 will round the solution to an integral one only losing a factor of $O(\log n)$ in the facility cost and a factor $O(1)$ in the connection cost, thus being $O(\log n)$ competitive with the optimal offline solution. Regarding the switching cost, by Lemma 2, the fractional solution is $O(\log m)$ competitive with the fractional solution. The cost of this solution will only increase by a factor of $O(1)$ after the randomized rounding, thus proving the result. $\square$

6. Discussion

We studied the online variant of the Dynamic Facility Location problem, where we proved lower bounds for deterministic and randomized algorithms and an almost matching upper bound for randomized algorithms.

An interesting future direction is to determine the competitive ratio for ODFL in the randomized case and close the gap between the lower bound and the upper bound. Another interesting direction is to study the ODFL problem in the case where the vector of positions between clients and facilities is drawn from a known distribution. Finally, it would be interesting to consider the online variant of other dynamic problems like the Dynamic Sum-Radii Clustering problem.

Author Contributions: Conceptualization, D.F., L.K., and L.Z.; methodology, D.F., L.K., and L.Z.; validation, D.F., L.K., and L.Z.; formal analysis, D.F., L.K., and L.Z.; investigation, D.F., L.K., and L.Z.; writing—original draft preparation, D.F., L.K., and L.Z.; writing—review and editing, D.F., L.K., and L.Z.; supervision, D.F. All authors have read and agreed to the published version of the manuscript.

Funding: Dimitris Fotakis was partially supported by the Hellenic Foundation for Research and Innovation (H.F.R.I.), project BALSAM, HFRI-FM17-1424. Loukas Kavouras is partially supported by a scholarship from the State Scholarships Foundation, granted by the action “Scholarships Grant Programme for second cycle graduate studies”, which is co-financed by Greece and the European Union (European Social Fund—ESF) through the Operational Programme “Human Resources Development, Education and Lifelong Learning 2014–2020”.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Meyerson, A. Online Facility Location. In Proceedings of the 42nd Annual Symposium on Foundations of Computer Science (FOCS), Las Vegas, NV, USA, 14–17 October 2001; pp. 426–431. [\[CrossRef\]](#)
2. Fotakis, D. Incremental algorithms for Facility Location and K -Median. *Theor. Comput. Sci.* **2006**, *361*, 275–313. [\[CrossRef\]](#)
3. Eisenstat, D.; Mathieu, C.; Schabanel, N. Facility Location in Evolving Metrics. In *Automata, Languages, and Programming, Proceedings of the 41st International Colloquium (ICALP 2014), Copenhagen, Denmark, 8–11 July 2014; Part II*; Esparza, J., Fraigniaud, P., Husfeldt, T., Koutsoupias, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2014.
4. Hochbaum, D.S. Heuristics for the fixed cost median problem. *Math. Program.* **1982**, *22*, 148–162. [\[CrossRef\]](#)
5. Guha, S.; Khuller, S. Greedy Strikes Back: Improved Facility Location Algorithms. *J. Algorithms* **1999**, *31*, 228–248. [\[CrossRef\]](#)
6. Li, S. A 1.488 approximation algorithm for the uncapacitated facility location problem. *Inf. Comput.* **2013**, *222*, 45–58. [\[CrossRef\]](#)
7. Fotakis, D. On the Competitive Ratio for Online Facility Location. *Algorithmica* **2008**, *50*, 1–57. [\[CrossRef\]](#)
8. Anagnostopoulos, A.; Bent, R.; Upfal, E.; Hentenryck, P.V. A simple and deterministic competitive algorithm for online facility location. *Inf. Comput.* **2004**, *194*, 175–202. [\[CrossRef\]](#)
9. An, H.; Norouzi-Fard, A.; Svensson, O. Dynamic Facility Location via Exponential Clocks. *ACM Trans. Algorithms* **2017**, *13*, 21:1–21:20. [\[CrossRef\]](#)
10. Buchbinder, N.; Chen, S.; Naor, J. Competitive Analysis via Regularization. In Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2014, Portland, OR, USA, 5–7 January 2014; pp. 436–444.
11. Borodin, A.; El-Yaniv, R. *Online Computation and Competitive Analysis*; Cambridge University Press: Cambridge, UK, 1998.