An Algorithmic Approach to Online Multi-Facility Location Problems

Christine Markarian[1], Abdul-Nasser Kassar[2] and Manal Yunis[2]
[1]*Department of Engineering and Information Technology, University of Dubai, U.A.E.*
[2]*Department of Information Technology and Operations Management, Lebanese American University, Lebanon*

Keywords: Robustness, Multi-Facility Location, Online Algorithms, Competitive Analysis, Randomized Rounding.

Abstract: Facility Location problems ask to optimally place facilities with respect to some objective so that all clients requesting a facility service are served. These are one of the most well-studied optimization problems spanning many research areas, such as operations research, computer science, and management science. Classical algorithmic study of Facility Location problems is based on the assumption that clients need to be served with one facility each. Nevertheless, in many real-world applications, facilities experience disruptions and to overcome their failures, a robust service is desired. To obtain this, clients are served with more than one facility, and this is commonly represented by an additional input parameter. The aim of the algorithm is then to provide a robust service to all clients while minimizing costs. This is known as the Multi-Facility Location problem (MFL), a well-known optimization problem, studied in the offline setting in which the entire input sequence is known to the algorithm in advance. In this paper, we address MFL in the online setting, in which client requests are not known in advance but revealed over time. We refer to it as the *Online Multi-Facility Location* problem (OMFL) and study its metric and non-metric variants. We propose the first online algorithms for these variants and measure their performance using the standard notion of competitive analysis. The latter is a worst-case analysis that compares the cost of the online algorithm to that of the optimal offline algorithm that is assumed to know all demands in advance.

1 INTRODUCTION

Facility Location (FL) is a classical NP-hard optimization problem widely spread in the fields of computer science and operations research (Drezner, 1995; Manne, 1964). In its simplest form, we are given a set of facilities and a set of clients. Each facility has an opening cost and each client i has a connecting cost to each facility j, which is the distance between i and j. The goal is to open a subset of the facilities and connect the clients to open facilities so as to minimize the sum of the facility costs and the connecting costs. FL is known as two versions, *metric* and *non-metric*. In the metric version, the distances are assumed to be symmetric and satisfy the triangle inequality.

In this paper, we focus on the algorithmic view of FL problems and consider an *online* setting in which clients are not known in advance but revealed to the algorithm over time. As soon as one arrives, it needs to be connected. Many real-world applications, that contain FL as a sub-problem, have this online nature, in which one is expected to react to present demands whenever they arrive, without knowing about future demands. Maintaining a given optimization goal becomes more challenging in the face of this uncertainty. This encourages the study of *online algorithms* (Fiat and Woeginger, 1998) for FL problems. The standard framework to measure online algorithms is *competitive analysis*, in which demands and their arrival order are selected by an oblivious adversary, that is unaware of the choices of the algorithm. An online algorithm is c-competitive or has competitive ratio c if for all sequences of demands, the cost incurred by the algorithm is at most c times the cost incurred by an optimal offline algorithm, which knows the entire sequence of demands in advance.

The study of *Facility Location* (FL) in the online setting was initiated by Meyerson (Meyerson, 2001), who introduced the metric *Online Facility Location* problem (OFL), and proposed an $O(\log n)$-competitive randomized algorithm, where n is the number of clients. Alon *et al.* (Alon et al., 2006) studied the non-metric version and proposed an $O(\log n \log m)$-competitive randomized algorithm, where n is the number of clients and m is the number of facilities. Many other variations were known for both metric and non-metric variants in the online setting (Abshoff et al., 2016; Anagnostopoulos et al., 2004; Divéki and Imreh, 2011; Fotakis, 2007; Fotakis, 2008; Markarian and Meyer auf der Heide, 2019).

Other variants include (Fotakis, 2011), in which

Markarian, C., Kassar, A. and Yunis, M.
An Algorithmic Approach to Online Multi-Facility Location Problems.
DOI: 10.5220/0010212200290035
In *Proceedings of the 10th International Conference on Operations Research and Enterprise Systems (ICORES 2021)*, pages 29-35
ISBN: 978-989-758-485-5

the algorithm can merge existing facilities with each other, and only the decision of assigning some demands to the same facility is irrevocable. Another variant by (Feldkord and Meyer auf der Heide, 2018) allows the algorithm to adapt the position of the facilities for costs proportional to the distance by which the position is changed.

All of these works assume that clients need to be served with one facility each. In many real-world applications, a robust service, in which a client is served with more than one facility, is desirable (Snyder, 2006; Snyder and Daskin, 2006; Gerodimos, 1998). Facilities and/or connections to facilities may be prone to failure and assigning clients to multiple facilities would provide a fault-tolerant solution. Such solutions are seeked in many applications, such as providing replicated cash data in distributed networks (Byrka et al., 2010).

1.1 Our Contribution

In this paper, we explore a generalization of online *Facility Location* (Alon et al., 2006; Fotakis, 2008), in which we are additionally given a parameter k, which is the number of facilities required to serve a client. We refer to it as the *Online Multi-Facility Location* problem, defined as follows.

Definition 1. *(Online Multi-Facility Location) We are given a collection of m facilities, n clients, and a positive integer k. Each facility has an* opening cost *and each client has a* connecting cost *to each facility. Clients arrive over time. As soon as one arrives, it needs to be connected to at least k open facilities. To open a facility, we pay its opening cost. To connect a client to a facility, we pay the corresponding connecting cost. The goal is to minimize the total opening and connecting costs.*

We address the metric and non-metric Online Multi-Facility Location problems. As far as we are aware of, there are no online algorithms in the literature that solve these variants or can be trivially extended to solve them.

Lower Bounds. For the non-metric version, there is a lower bound of $\Omega(\log m \log n)$, under the assumption that NP $\not\subseteq$ BPP, where m is the number of facilities and n is the number of clients, due to the lower bound given for *Online Set Cover* (Korman, 2005). As for the metric version, there is a lower bound of $\Omega(\frac{\log n}{\log \log n})$, where n is the number of clients, due to the lower bound given for metric OFL (Fotakis, 2008).

Our results can be summarized as follows.

1. We refer to the non-metric version as *Online Non-metric Multi-Facility Location* (ONMFL). We propose an online $O(\log(kn)\log m)$-competitive randomized algorithm for ONMFL, where m is the number of facilities; n is the number of clients; and k is the number of required connections.

   The latter uses a *randomized rounding* approach that first constructs a fractional solution and then rounds it into an integral one. Its competitive analysis is based on first comparing the fractional solution constructed by the algorithm to the optimal offline solution and then measuring the fractional solution in terms of the integral one.

2. We refer to the metric version as *Online Metric Multi-Facility Location* (OMMFL). We propose an online $O(\max\{\frac{f_{max}}{f_{min}}, \frac{c_{max}}{c_{min}}\} \cdot k \cdot \frac{\log n}{\log \log n})$-competitive deterministic algorithm for OMMFL, where n is the number of clients; k is the number of required connections; c_{max} and c_{min} are the maximum and minimum connecting costs, respectively; f_{max} and f_{min} are the maximum and minimum facility costs, respectively.

   The idea of the algorithm is to ensure first that each client is connected to one facility by running an algorithm for metric *Online Facility Location* (OFL). Then, the $k-1$ remaining connections are made by choosing the cheapest possible facilities so as not to worsen the competitive ratio by much. Our approach can be seen as a general framework that transforms any given online algorithm for metric OFL into an algorithm for OMMFL, by losing a bounded factor in the competitive ratio.

1.2 Outline

The remainder of this paper is structured as follows. Section 2 gives an overview of literature related to ONMFL and presents an algorithm for ONMFL along with its competitive analyis. Section 3 gives an overview of literature related to OMMFL and presents an algorithm for OMMFL along with its competitive analyis. Section 4 concludes the paper with some future work.

2 ONLINE NON-METRIC MULTI-FACILITY LOCATION

In this section, we start with an overview of works related to ONMFL and some preliminaries. Then we present an online randomized algorithm for ONMFL and analyze its competitive ratio.

2.1 Preliminaries & Related Work

ONMFL is a generalization of the *Online Non-metric Facility Location* (ONFL) (Alon et al., 2006) with $k = 1$. Alon *et al.* (Alon et al., 2006) gave an $O(\log n \log m)$-competitive randomized algorithm for ONFL, where n is the number of clients and m is the number of facilities.

A closely related problem is the *Online Set k-Multicover* (OSMC) (Berman and DasGupta, 2008). Given a universe $\mathcal{U}$ of n elements, a family $\mathcal{S}$ of m subsets of V, each associated with a cost, and a positive integer k. A subset $D \subseteq \mathcal{U}$ of elements arrives over time. OSMC asks to select a collection $\mathcal{C} \subseteq \mathcal{S}$ of subsets, of minimum cost, such that each arriving element belongs to at least k subsets of $\mathcal{C}$. Berman and DasGubta (Berman and DasGupta, 2008) proposed an $O(\log m \log d)$-competitive randomized algorithm for OSMC, where m is the number of subsets and d is the maximum set size.

Transformations between OSMC and ONMFL instances can be made in both directions. An instance of ONMFL can be transformed into an instance of OSMC as follows. We associate each facility with each of the $2^n - 1$ possible groups of clients, and let each facility/group be a subset, with cost equal to the sum of the cost of the facility and the connecting costs of the clients in the group to the facility. We let each client be an element. Following this transformation, the algorithm of Berman and DasGubta would imply a feasible algorithm for ONMFL, but with competitive ratio $O(\log(m(2^n - 1)) \log n)$, where n is the number of clients, and m is the number of facilities. An instance of OSMC can be transformed into an instance of ONMFL as follows. We represent each subset by a facility and let the opening cost be the subset cost. We represent each element by a client and set the connecting cost to 0 if the client belongs to the subset/facility and infinity otherwise.The offline version of OSMC, in which all elements are given in advance, has an $O(\log d)$-approximation (Johnson, 1974; Virkumar, 2001), where d is the maximum set size. Similar transformations can be made in the offline setting. This implies an $O(\log n)$-approximation for the offline version of ONMFL, where n is the number of clients. These are the best achievable unless $NP \subseteq DTIME(n^{\log\log n})$ (Feige, 1998). The *Online Set Cover* (OSC) (Alon et al., 2003) is a special case of OSMC with $k = 1$. Korman's lower bound on the competitive ratio of OSC (Korman, 2005) implies an $\Omega(\log m \log n)$ lower bound on the competitive ratio of ONMFL, under the assumption that NP $\nsubseteq$ BPP, where m is the number of facilities and n is the number of clients.

For other online problems solved with similar techniques as the ones used in this paper, see (Markarian and Kassar, 2020; Hamann et al., 2018).

2.2 Online Algorithm

Our algorithm for ONMFL is based on constructing a fractional solution first and then rounding it online into an integral solution. Unlike (Alon et al., 2003) and (Alon et al., 2006) for *Online Set Cover* (OSC) and *Online Non-metric Facility Location* (ONFL), respectively, in which a potential function is used in the competitive analysis of the algorithms, our analysis is based on simple arguments, similar to those we gave in our previous work for a related problem, the *Online Node-weighted Steiner Forest* (Markarian, 2018).

We start with giving the following graph formulation for ONMFL. Given a root node r, m facility nodes, and n client nodes. We add an edge from r to each facility node j, with cost equal to the opening cost of facility j. We add an edge from each facility node j to each client node i, with cost equal to the connecting cost of client i to facility j. As soon as a client i arrives, we need to purchase k disjoint paths between r and i. The goal is to minimize the total costs of the paths purchased, where the cost of a path is the cost of its edges. To output a solution for ONMFL, each facility whose corresponding edge is purchased will be opened, and each client whose corresponding edge to an open facility is purchased will be connected to that facility.

The algorithm initially knows n, the number of clients; k, the number of required connections; and the opening costs of facilities. A subset D of $n' \leq n$ clients arrives over time. As soon as a client i arrives, the algorithm is given the connecting costs of i to each facility, and is expected to react.

We consider the graph formulation described earlier. Let r be a root node; each facility is represented as a facility node, and has an edge to r, associated with its opening cost. Upon the arrival of a new client i, the algorithm creates a client node for it and adds an edge from this node to each facility node, associated with the given connecting cost. Let $G = (V, E)$ be this graph.

Each edge added to E is given a fraction, set intitially to 0. The algorithm does not allow these fractions to decrease over time. These form a *fractional solution* for ONMFL. The *maximum flow* between node u and node v in G is the smallest total weight of edges which if removed would disconnect u from v. These edges form a *minimum cut* between u and v in G. Let c_e and f_e be the cost and fraction of edge e, respectively. A path is purchased if and only if its edges are purchased.

Random Process. The algorithm makes its random choices, based on α, the minimum among $2\lceil\log(kn+1)\rceil$ independently chosen random variables, distributed uniformly in the interval $[0,1]$.
Next, we describe how the algorithm reacts upon the arrival of a new client.

Input: $G=(V,E)$ and client node $i \in D$
Output: Set of edges purchased

Make a copy G' of G;
As long as there are $< k$ disjoint paths purchased between r and i in G, do the following:

1. While the maximum flow between r and i in G' is less than 1, construct a minimum cut Q between r and i in G'; for each edge $e \in Q$, make the following fraction increase:
$$f_e = f_e \cdot (1+1/c_e) + \frac{1}{|Q| \cdot c_e}$$
2. Purchase each edge e with $f_e > \alpha$.
3. If there is no purchased path between r and i in G', find a minimum-cost such path and purchase it.
4. Refer to all facilities whose corresponding edges were purchased as *open*; delete from G' the purchased edges between i and each open facility.

2.3 Competitive Analysis

The algorithm buys edges in the second and third steps. In the second step, its choices are made based on the random process, whereas in the third step, its choices are made to guarantee a feasible solution. We measure the expected cost of each separately. Let Opt be the cost of the optimal offline solution and let $frac$ be the cost of the *fractional* solution constructed by the algorithm in the first step.

Choices Based on Random Process: Let S' be the set of edges purchased in the second step of the algorithm and let $C_{S'}$ be its expected cost. These edges are purchased by the algorithm based on the random process described earlier. Let us fix an $l : 1 \le l \le 2\lceil\log(kn+1)\rceil$ and an edge e. We denote by $X_{e,l}$ the indicator variable of the event that e is chosen by the algorithm based on the random choice of l.

$$C_{S'} = \sum_{e\in S'} \sum_{l=1}^{2\lceil\log(kn+1)\rceil} c_e \cdot Exp\left[X_{e,l}\right]$$

$$= 2\lceil\log(kn+1)\rceil \sum_{e\in S'} c_e f_e \qquad (1)$$

Notice that $\sum_{e\in S'} c_e f_e$ is upper bounded by the cost of the fractional solution. The latter can be measured against the optimal offline solution, as follows. The idea here is that every time the algorithm performs a fraction increase, it does not exceed 2. Moreover, the total number of fraction increases can be measured in terms of the cost of the optimal offline solution.

The fraction increase contributed by each edge e in a minimum cut Q is $\left(\frac{f_e}{c_e} + \frac{1}{|Q|\cdot c_e}\right)$. The algorithm would make a fraction increase only if the maximum flow is less than 1. This means we have that $\sum_{e\in Q} f_e < 1$ before a fraction increase. Therefore, each fraction increase does not exceed:

$$\sum_{e\in Q} c_e \cdot \left(\frac{f_e}{c_e} + \frac{1}{|Q|\cdot c_e}\right) < 2 \qquad (2)$$

As long as the algorithm hasn't purchased at least k disjoint paths between r and a given client i, it enters the loop that starts by constructing a maximum flow on the graph G'. Notice how G' shrinks over time, as the algorithm purchases the paths.

Lemma 1. *Whenever the algorithm makes a fraction increase, G' contains at least one path that is also in the optimal offline solution.*

Proof. To see why this holds, fix any time t before a fraction increase. Let $s < k$ be the number of disjoint paths purchased by the algorithm at time t. G' at time t must contain at least one optimal path since G' is constructed by removing s (less than k) feasible paths from G and the optimal offline solution contains at least k disjoint paths in G. □

Finally, the algorithm would have an edge e from the optimal offline solution in every minimum cut Q of G', since Q must contain an edge from each path, by definition. Based on the equation for the fraction increase, after $O(\log|Q|)$ fraction increases, the fraction f_e of e becomes 1, and no further increases can be made, as e will not be in any future minimum cut. The size of any minimum cut is upper bounded by m, the number of facilities or the maximum available paths between r and client i. We can now bound the fractional solution:

$$frac \le O(\log m \cdot Opt) \qquad (3)$$

Combining Equations 1, 2, and 3 imply an upper bound on the expected cost $C_{S'}$ of the edges bought in the second step of the algorithm:

$$C_{S'} \le O(\log(kn)\log m \cdot Opt) \qquad (4)$$

Choices to Guarantee Feasible Solution: Let S'' be the set of edges purchased in the third step of the algorithm and let $C_{S''}$ be its expected cost. These edges are purchased by the algorithm only if a path has not been bought by the random process in the second step. Every time the algorithm purchases a path in this step, its cost does not exceed Opt since the algorithm buys the minimum-cost path in G', and as we showed earlier in Lemma 1, G' contains at least one path that is also in the optimal solution.

- (one client, one path) We start with calculating the expected cost incurred by a single client for purchasing a single path. Fix a client i. Let Q_{j+1} be a minimum cut of G' constructed after the algorithm has purchased $j < k$ disjoint paths between r and i and has completed the first step. The probability of purchasing the $(j+1)^{th}$ path for a single $1 \leq l \leq 2\lceil \log(kn+1) \rceil$ is:
$$\prod_{e \in Q_{j+1}} (1 - f_e) \leq e^{-\sum_{e \in Q_{j+1}} f_e} \leq 1/e$$
Notice that the last inequality holds because the algorithm ensures that $\sum_{e \in Q_{j+1}} f_e \geq 1$ at the end of the first step (Max-flow min-cut theorem). The expected cost of purchasing the $(j+1)^{th}$ path for all $1 \leq l \leq 2\lceil \log(kn+1) \rceil$, is less than $1/(kn)^2 \cdot Opt$.
- (one client, k paths) The expected cost of purchasing all k paths is the sum of the expected costs for each path and is less than $k \cdot 1/(kn)^2 \cdot Opt$.
- (total cost of all clients) The total expected cost incurred by all n' clients that arrived is less than:
$$\begin{aligned} n' \cdot k \cdot 1/(kn)^2 \cdot Opt &\leq n \cdot k \cdot 1/(kn)^2 \cdot Opt \\ &= 1/kn \cdot Opt \end{aligned}$$
.
Therefore, the expected cost $C_{S''}$ of the edges bought in the third step of the algorithm is:
$$C_{S''} \leq 1/kn \cdot Opt \quad (5)$$

Equations 4 and 5 yield to the following theorem.

Theorem 1. *There is an online $O(\log(kn)\log m)$-competitive randomized algorithm for the Online Non-metric Multi-Facility Location, where m is the number of facilities, n is the number of clients, and k is the number of required connections.*

3 ONLINE METRIC MULTI-FACILITY LOCATION

In this section, we start with an overview of works related to OMMFL and some preliminaries. Then we present an online randomized algorithm for OMMFL and analyze its competitive ratio.

While this problem has been intensively studied in the offline setting (Byrka et al., 2010; Yan and Chrobak, 2011), we are not aware of any online algorithm for it.

3.1 Preliminaries & Related Work

OMMFL is a generalization of metric *Online Facility Location* (OFL) (Fotakis, 2007; Fotakis, 2008; Meyerson, 2001) with $k = 1$. Meyerson (Meyerson, 2001) introduced metric OFL and proposed an $O(\log n)$-competitive randomized algorithm, where n is the number of clients. Fotakis (Fotakis, 2008) showed that no randomized online algorithm can achieve a competitive ratio better that $\Omega(\frac{\log n}{\log\log n})$ against an oblivious adversary and gave a deterministic algorithm with asymptotically matching $O(\frac{\log n}{\log\log n})$-competitive ratio. In another work later, he proposed a primal-dual deterministic algorithm with $O(\log n)$-competitive ratio, that was simpler to formulate, analyze, and implement (Fotakis, 2007). The competitive ratio we achieve for OMMFL is based on running the deterministic algorithm of Fotakis (Fotakis, 2008) for metric OFL.

The lower bound on the competitive ratio of metric OFL by Fotakis (Fotakis, 2008) implies an $\Omega(\frac{\log n}{\log\log n})$ lower bound on the competitive ratio of OMMFL.

3.2 Online Algorithm

Let A_{OFL} be any online (deterministic or randomized) algorithm for metric *Online Facility Location* (OFL), with competitive ratio r. Given an instance I of OMMFL with positive integer k. Client i arrives. Our algorithm needs to connect i to k different open facilities.

1. The algorithm starts by running A_{OFL} on instance I, where $k = 1$. This results in opening some facilities and connecting i to one open facility.
2. If i is the first client, we open the cheapest $k-1$ facilities other than the one i is connected to. From this point on, there are at least k open facilities. Until A_{OFL} opens these facilities itself, these remain closed with respect to A_{OFL}.
3. The algorithm will then connect i to any other $k-1$ open facilities.

3.3 Competitive Analysis

Let I be an instance of OMMFL with positive integer k. Let I' be the same instance as I except for $k=1$. Let Opt and Opt' be the cost of the optimal solution for I and that for I', respectively. Let C and C' be the cost of our algorithm for I and that of A_{OFL} for I', respectively. We denote by C_{fac} and C_{con} the costs incurred by our algorithm to open facilities and to connect clients, respectively. We denote by C'_{fac} and C'_{con} the costs incurred by A_{OFL} to open facilities and to connect clients, respectively.

The algorithm opens the cheapest $k-1$ facilities other than the ones opened by A_{OFL}. Let f_{max} be the maximum facility cost and f_{min} the minimum facility cost. Thus, we have that:

$$C_{fac} \leq C'_{fac} + f_{max} \cdot (k-1) \tag{6}$$

Given any client i. Apart from its connecting cost c_i incurred by A_{OFL}, our algorithm connects i to $k-1$ other facilities, each resulting in a connecting cost at most $\frac{c_{max}}{c_{min}} \cdot c_i$, where c_{max} and c_{min} are the maximum and minimum connecting costs, respectively. This implies an overall connecting cost:

$$C_{con} \leq C'_{con} \cdot \left(1 + \frac{c_{max}}{c_{min}}(k-1)\right) \tag{7}$$

We now add the two equations above and do some algebraic manipulations by using:

$$C'_{con} \leq C'$$
$$C'_{fac} \leq C'$$
$$C' \geq c_{min}$$
$$C' \geq f_{min}$$

This yields to:

$$C \leq C' \cdot \left(2 + \frac{f_{max}}{f_{min}}(k-1) + \frac{c_{max}}{c_{min}}(k-1)\right) \tag{8}$$

Recall that A_{OFL} is r-competitive and so $C' \leq r \cdot Opt'$. Since $Opt' \leq Opt$, the theorem below follows.

Theorem 2. *Given an online (deterministic or randomized) r-competitive algorithm for metric Online Facility Location. Then there is an online $O(\max\{\frac{f_{max}}{f_{min}}, \frac{c_{max}}{c_{min}}\} \cdot k \cdot r)$-competitive algorithm for the Online Metric Multi-Facility Location, where k is the number of required connections; c_{max} and c_{min} are the maximum and minimum connecting costs, respectively; f_{max} and f_{min} are the maximum and minimum facility costs, respectively.*

Running the deterministic algorithm of Fotakis (Fotakis, 2008) for metric OFL, with $O(\frac{\log n}{\log\log n})$-competitive ratio, results in the following.

Corollary 1. *There is an online $O(\max\{\frac{f_{max}}{f_{min}}, \frac{c_{max}}{c_{min}}\} \cdot k \cdot \frac{\log n}{\log\log n})$-competitive deterministic algorithm for the Online Metric Multi-Facility Location, where n is the number of clients; k is the number of required connections; c_{max} and c_{min} are the maximum and minimum connecting costs, respectively; f_{max} and f_{min} are the maximum and minimum facility costs, respectively.*

4 CONCLUDING REMARKS & FUTURE WORK

In this paper, we have assumed there is a unique positive integer k for all clients. In many application scenarios, it is likely that clients have different number of required connections. A slight modification in our algorithms would yield to $O(\log(k_{max}n)\log m)$ and $O(\max\{\frac{f_{max}}{f_{min}}, \frac{c_{max}}{c_{min}}\} \cdot k_{max} \cdot \frac{\log n}{\log\log n})$ competitive ratios for ONMFL and OMMFL, respectively, where k_{max} is the maximum required connections. One research direction is to target better competitive ratios for these variants.

This brings us to the next question, for *Online Metric Multi-Facility Location* (OMMFL), whether it is possible to get rid of the parameters c_{max}, c_{min}, f_{max}, and f_{min} from the competitive ratio or achieve lower bounds in terms of these parameters. To achieve the former, one may want to attempt a primal-dual approach, for instance, by trying to extend the algorithm of Fotakis (Fotakis, 2007) for metric *Online Facility Location*.

Finally, in our model here, demands and their arrival order are given by an oblivious adversary. One may want to consider other types of adversary for these sequences, for instance, by exploring various probability distributions.

REFERENCES

Abshoff, S., Kling, P., Markarian, C., Meyer auf der Heide, F., and Pietrzyk, P. (2016). Towards the price of leasing online. *Journal of Combinatorial Optimization*, 32(4):1197–1216.

Alon, N., Awerbuch, B., and Azar, Y. (2003). The online set cover problem. In *Proceedings of the Thirty-fifth Annual ACM Symposium on Theory of Computing*, STOC '03, pages 100–105, New York, NY, USA. ACM.

Alon, N., Awerbuch, B., Azar, Y., Buchbinder, N., and Naor, J. S. (2006). A general approach to online network optimization problems. *ACM Trans. Algorithms*, 2(4):640–660.

Anagnostopoulos, A., Bent, R., Upfal, E., and Van Hentenryck, P. (2004). A simple and deterministic competitive algorithm for online facility location. *Information and Computation*, 194(2):175–202.

Berman, P. and DasGupta, B. (2008). Approximating the online set multicover problems via randomized winnowing. *Theoretical Computer Science*, 393(1):54 – 71.

Byrka, J., Srinivasan, A., and Swamy, C. (2010). Fault-tolerant facility location: A randomized dependent lp-rounding algorithm. In Eisenbrand, F. and Shepherd, F. B., editors, *Integer Programming and Combinatorial Optimization*, pages 244–257, Berlin, Heidelberg. Springer Berlin Heidelberg.

Divéki, G. and Imreh, C. (2011). Online facility location with facility movements. *Central European Journal of Operations Research*, 19(2):191–200.

Drezner, Z. (1995). *Facility Location - A Survey of Applications and Methods*. Springer Series in Operations Research and Financial Engineering.

Feige, U. (1998). A threshold of ln n for approximating set cover. *J. ACM*, 45(4):634?652.

Feldkord, B. and Meyer auf der Heide, F. (2018). Online facility location with mobile facilities. In *Proceedings of the 30th ACM Symposium on Parallelism in Algorithms and Architectures*, SPAA '18, page 373–381, New York, NY, USA. Association for Computing Machinery.

Fiat, A. and Woeginger, G. J. (1998). *Online algorithms: The state of the art*, volume 1442. Springer.

Fotakis, D. (2007). A primal-dual algorithm for online non-uniform facility location. *Journal of Discrete Algorithms*, 5(1):141 – 148.

Fotakis, D. (2008). On the competitive ratio for online facility location. *Algorithmica*, 50(1):1 – 57.

Fotakis, D. (2011). Online and incremental algorithms for facility location. *SIGACT News*, 42(1):97–131.

Gerodimos, A. (1998). *The Journal of the Operational Research Society*, 49(12):1303–1304.

Hamann, H., Markarian, C., auf der Heide, F. M., and Wahby, M. (2018). Pick, pack, & survive: Charging robots in a modern warehouse based on online connected dominating sets. In Ito, H., Leonardi, S., Pagli, L., and Prencipe, G., editors, *9th International Conference on Fun with Algorithms, FUN 2018, June 13-15, 2018, La Maddalena, Italy*, volume 100 of *LIPIcs*, pages 22:1–22:13. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

Johnson, D. S. (1974). Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences*, 9(3):256 – 278.

Korman, S. (2005). On the use of randomization in the online set cover problem. Master's thesis, Weizmann Institute of Science, Israel.

Manne, A. S. (1964). Plant location under economies-of-scale—decentralization and computation. *Management Science*, 11(2):213–235.

Markarian, C. (2018). An optimal algorithm for online prize-collecting node-weighted steiner forest. In *Combinatorial Algorithms - 29th International Workshop, IWOCA 2018, Singapore, July 16-19, 2018, Proceedings*, pages 214 – 223.

Markarian, C. and Kassar, A. (2020). Online deterministic algorithms for connected dominating set & set cover leasing problems. In Parlier, G. H., Liberatore, F., and Demange, M., editors, *Proceedings of the 9th International Conference on Operations Research and Enterprise Systems, ICORES 2020, Valletta, Malta, February 22-24, 2020*, pages 121–128. SCITEPRESS.

Markarian, C. and Meyer auf der Heide, F. (2019). Online algorithms for leasing vertex cover and leasing non-metric facility location. In Parlier, G. H., Liberatore, F., and Demange, M., editors, *Proceedings of the 8th International Conference on Operations Research and Enterprise Systems, ICORES 2019, Prague, Czech Republic, February 19-21, 2019*, pages 315–321. SciTePress.

Meyerson, A. (2001). Online facility location. In *Proceedings 42nd IEEE Symposium on Foundations of Computer Science*, pages 426 – 431. IEEE.

Snyder, L. V. (2006). Facility location under uncertainty: a review. *IIE Transactions*, 38(7):547–564.

Snyder, L. V. and Daskin, M. S. (2006). Stochastic p-robust location problems. *IIE Transactions*, 38(11):971–985.

Virkumar, V. V. (2001). *Approximation Algorithms*. Springer-Verlag.

Yan, L. and Chrobak, M. (2011). Approximation algorithms for the fault-tolerant facility placement problem. *Information Processing Letters*, 111(11):545 – 549.