Memoryless Facility Location in One Pass ⋆

Dimitris Fotakis

Department of Information and Communication Systems Engineering
University of the Aegean, 83200 Karlovasi, Samos, Greece.
Email: fotakis@aegean.gr

Abstract. We present the first one-pass memoryless algorithm for metric Facility Location which maintains a set of facilities approximating the optimal facility configuration within a constant factor. The algorithm considers the demand points one-by-one in arbitrary order, is randomized and very simple to state and implement. It runs in linear time and keeps in memory only the facility locations currently open. We prove that its competitive ratio is less than 14 in the special case of uniform facility costs and less than 49 in the general case of non-uniform facility costs.

Keywords: Facility Location, Incremental Algorithms, Online Algorithms.

1 Introduction

In many applications dealing with massive data sets in the form of data streams, we have to maintain a set of representatives for a sequence of points in a metric space. This is performed by a *streaming algorithm* which considers the points one-by-one in arbitrary order and maintains a set of representatives of reasonable accuracy using a limited amount of computational resources (see e.g. [13] for a survey on streaming computation).

This scenario can be naturally relaxed to a Facility Location problem. Given a (multi)set of n demand points in a metric space, the Facility Location problem seeks for a set of locations to open facilities minimizing the total cost of opening facilities and assigning every demand point to the nearest facility. To model the streaming scenario above, we let facilities play the role of representatives and require that the algorithm be *memoryless* and operate in *one pass*. In addition, we consider the special case of *uniform facility costs*, where the cost of opening a facility is the same for all points.

The algorithm is charged for the inaccuracy of its facilities (assignment cost) and the cost of maintaining them in main memory (facility cost). The assignment cost is the sum of distances of the points considered so far to their closest facilities. As for the facility cost, we can regard the algorithm as *buying* its memory space. Thus keeping a point in memory as a representative is equivalent to making the point a new facility. We assume that buying memory is an *irrevocable decision* and charge the algorithm for its *worst-case memory* consumption.

A *one-pass* algorithm considers the points one-by-one in arbitrary order. When a point is considered, a *memoryless* algorithm can either discard its location or make it a new facility. In the latter case, either the new facility replaces some existing one (i.e. the existing facility is discarded and the new facility takes its place in main memory) or the algorithm buys additional memory space and incurs the irrevocable cost of increasing its facilities by one. The algorithm has no access to previously

⋆ A preliminary version of this work appeared in the Proceedings of the 23rd Annual Symposium on Theoretical Aspects of Computer Science (STACS 2006), Lecture Notes in Computer Science 3884.

considered points unless it keeps them in memory (as open facilities). Hence, the algorithm incurs an assignment cost for each discarded point equal to its distance to the nearest facility. The assignment cost is not irrevocable since it may increase or decrease every time the algorithm's facility configuration changes. Thus a one-pass memoryless algorithm can be regarded as a streaming algorithm using only $\mathrm{O}(|F_{\max}|)$ space, where $|F_{\max}|$ denotes the maximum number of facilities maintained by the algorithm.

1.1 Related Work and Motivation

There has been a significant amount of work on streaming algorithms for k-Median[1], a problem closely related to Facility Location. Guha *et al.* [10] initiated the study of streaming algorithms for k-Median. Their algorithms run in $\tilde{\mathrm{O}}(nk)$ time[2] and $\mathrm{O}(n^\epsilon)$ space and achieve an approximation ratio of $2^{\mathrm{O}(1/\epsilon)}$ provided that k is significantly smaller than n^ϵ. The best known streaming algorithm for k-Median is the randomized algorithm of Charikar *et al.* [5], which runs in $\mathrm{O}(nk \log^2 n)$ time and $\mathrm{O}(k \log^2 n)$ space and achieves a constant approximation ratio with high probability.

Indyk [11] considered k-Median, Facility Location, and other geometric problems in the discrete d-dimensional space and the turnstile model, where the data stream consists of both insertions and deletions of points. For k-Median, Indyk gave a randomized $\mathrm{O}(1)$-approximation algorithm using $\tilde{\mathrm{O}}(k)$ space and having $\tilde{\mathrm{O}}(k)$ update time per insertion/deletion and $\tilde{\mathrm{O}}(nk)$ query time. Frahling and Sohler [8] recently improved the approximation ratio to $(1 + \epsilon)$ by a randomized algorithm using $\tilde{\mathrm{O}}(k^2/\epsilon^{\mathrm{O}(d)})$ space and having polylogarithmic update time and $\mathrm{O}(k^5 \log^9 n + k^2 \log^5 n \exp(\epsilon^{-d}))$ query time.

In Facility Location, we have a facility cost instead of the constraint on the number of facilities (medians). Hence it may be advantageous to open a large number of facilities. In fact there are instances where every (near-)optimal solution needs to open $\Omega(n)$ facilities and every $\mathrm{O}(1)$-approximation algorithm needs to run for $\Omega(n^2)$ time (see e.g. the lower bound in [14, Section 4]). Thus we cannot hope for an algorithm, even a randomized one, that maintains an $\mathrm{O}(1)$-approximation to the optimal facility configuration in $\mathrm{o}(n^2)$ time or $\mathrm{o}(n)$ space.

To overcome this limitation, we may let the algorithm approximate only the *optimal cost* (i.e. maintain only the cost of an approximate facility configuration and not the configuration itself). In the discrete d-dimensional space and the turnstile model, Indyk [11] gave a randomized algorithm that approximates the optimal cost within a factor of $\mathrm{O}(d \log^2 n)$ in $\tilde{\mathrm{O}}(n)$ time and $\mathrm{O}(d^2 \log^2 n)$ space. In the *offline* setting, where there is random access to the input, Bădoiu *et al.* [3] proved that the optimal cost of Facility Location with uniform facility costs can be approximated within a constant factor in $\mathrm{O}(n \log^2 n)$ time. For non-uniform facility costs, they showed that there is no $\mathrm{o}(n^2)$-time algorithm approximating the optimal cost within any factor.

Nevertheless in practical applications dealing with data streams, we usually need an algorithm that maintains an approximate facility configuration and not just an estimation of the optimal cost (see e.g.

[1] Given a (multi)set of n demand points in a metric space, the k-Median problem seeks for k points (medians) that minimize the sum of distances from each demand point to the nearest median.

[2] The $\tilde{\mathrm{O}}$-notation is the same as the O-notation except it hides polylogarithmic terms instead of constants, e.g. $\tilde{\mathrm{O}}(nk)$ is $\mathrm{O}(nk\,\mathrm{poly}(\log n))$.

[9] for applications on clustering large data sets). Then one naturally turns to streaming algorithms using a *minimal* amount of resources. Arguably a one-pass memoryless algorithm is of this sort. It maintains an approximate facility configuration using just the memory required to store its facilities.

The *online* algorithm of Meyerson [12] was the first one-pass memoryless algorithm for metric Facility Location. In the online setting, the decisions (and the costs) of opening a facility at a particular location and assigning a demand to a particular facility are irrevocable. Meyerson's algorithm is randomized and achieves a competitive ratio of $\mathrm{O}(\log n)$. If the demand points arrive in *random order*, the algorithm achieves an expected constant competitive ratio. However, the latter version is *not a one-pass memoryless algorithm* because computing a random permutation of the demand sequence requires random access to it. Despite its logarithmic competitiveness, Meyerson's algorithm found significant applications mostly because it is very simple, memoryless, and operates in one-pass. Also motivated by practical applications, Anagnostopoulos *et al.* [1] presented a deterministic online algorithm for the d-dimensional space which achieves a competitive ratio of $\Theta(2^d \log n)$ and can be regarded as a memoryless algorithm.

On the other hand, no online algorithm for Facility Location can achieve a competitive ratio better than $\Omega(\frac{\log n}{\log\log n})$ even on the line [6]. To overcome the logarithmic lower bound, we considered the incremental setting introduced in [4], where the algorithm can also merge existing facilities and only the decision of clustering some demands together is irrevocable. In [7], we presented an $\mathrm{O}(1)$-competitive deterministic incremental algorithm for Facility Location with uniform facility costs. That algorithm runs in $\mathrm{O}(n^2|F_{\max}|)$ time and makes heavy and non-trivial use of previous demands' locations for opening and merging facilities. Thus it is *not a memoryless* algorithm.

1.2 Contribution

In previous work, Anagnostopoulos *et al.* [1] and Meyerson [12] presented one-pass memoryless algorithms of logarithmic competitive ratio and we presented a one-pass (but not memoryless) algorithm of constant competitive ratio [7]. In this work, we present the first one-pass memoryless algorithm that maintains a set of facilities approximating the optimal facility configuration within a constant factor.

The algorithm is randomized and extremely simple to state and implement. It is based on the notion of *final distance*, an upper bound on the distance of a demand to the nearest facility in the future. We use a modification of the randomized rule of [12] for opening new facilities. More specifically, each new demand becomes a new facility with probability proportional to the ratio of its *final distance* to the facility cost. To avoid facility proliferation leading to a logarithmic competitiveness, we develop a memoryless version of the merge rule of [7]. In particular, a *replacement ball* around each new facility is determined when the facility opens. Then the facility is replaced by the first new facility in its replacement ball[3].

The algorithm keeps in memory only the locations and the replacement radii of the facilities currently open. It runs in $\mathrm{O}(n|F_{\max}|)$ time and $\mathrm{O}(|F_{\max}|)$ space. The worst-case running time is

[3] Though similar in nature, the merge rule used in [7] is significantly more complicated than the replacement rule in this work and *not memoryless*. In [7], every facility maintains the exact locations of the demands initially assigned to it and gradually decreases its merge radius so as to bound the increase of their assignment cost due a merge operation. In this paper, the restriction to memoryless algorithms requires that the replacement radius of a new facility be determined once and for all when the facility opens.

linear in the size of the input because representing a metric space with n points requires $\Theta(n^2)$ bits. We prove that the algorithm's competitive ratio is less than 14.

We should note that the algorithm maintains an $\mathrm{O}(1)$-competitive facility configuration but it does not maintain *the assignment cost* of the demands considered so far with respect to it. A provably good facility configuration suffices in most practical applications (e.g. location problems, clustering problems). If an estimation of the assignment cost is required, we can use the techniques of [5] and maintain a small set of weighted points whose assignment cost is within a constant factor of the algorithm's cost. This increases the algorithm's time and space complexity by a factor of $\mathrm{O}(\log^2 n)$.

We do not know how to exclude the possibility of a one-pass *memoryless* algorithm which maintains both a competitive facility configuration and its assignment cost. However, the lower bound of [6] implies that every one-pass algorithm which maintains $\mathrm{o}(k \log n)$ facilities incurs an *initial* assignment cost of $\omega(1)$ times the optimal cost, where k is the number of optimal facility locations and a demand's initial assignment cost is the distance to its nearest facility just after the demand is considered. In other words, the initial assignment cost is not an $\mathrm{O}(1)$-approximation to the optimal assignment cost unless the algorithm maintains $\Omega(k \log n)$ facilities. Therefore, to establish a constant competitive ratio, we consider the updated assignment cost every time a new facility significantly closer to an optimal facility opens. Even though this cost can be bounded in terms of the optimal cost, its actual value cannot be maintained (even approximately) unless we keep sufficient information about the locations of the demand points. Hence a logarithmic competitive ratio seems unavoidable for every one-pass memoryless algorithm that maintains both a competitive facility configuration and its assignment cost.

We also present a generalized version of the algorithm dealing with non-uniform facility costs and prove that its competitive ratio is less than 49. The algorithm runs in $\mathrm{O}(n\,|M|)$ (i.e. linear) time and $\mathrm{O}(|M|)$ space, where $|M|$ is the number of potential facility locations. As in the case of uniform facility costs, the algorithm does not maintain any information about the demand sequence other than the locations and the replacement radii of its facilities. The rule for opening new facilities is randomized and entirely different from the corresponding rule in [12]. In particular, our rule does not require grouping facilities of almost equal costs together and is significantly simpler to state and analyze. Facility replacement is based on the notion of replacement ball, but the calculation of the replacement radius carefully takes non-uniform facility costs into account.

In the general case of non-uniform facility costs, every algorithm must be aware of the potential facility locations and their opening costs. Hence the motivation for developing a memoryless algorithm is not as strong as for uniform facility costs. Nevertheless, we believe that our result is interesting because it demonstrates that even for non-uniform facility costs, a near optimal facility configuration can be maintained by a simple one-pass algorithm that has optimal time and space complexity and does not keep any information about the demand sequence.

1.3 Problem Definition and Notation

We evaluate the performance of the algorithms presented in this paper using *competitive analysis* (see e.g. [2]). An algorithm is c-competitive if for all sequences of demand points, the cost incurred by the algorithm is at most c times the cost incurred by an optimal offline algorithm on the same instance.

We consider a metric space $\mathcal{M} = (M, d)$, where M is the set of points and $d : M \times M \mapsto \mathbb{R}_+$ is the distance function, which is non-negative, symmetric and satisfies the triangle inequality. For a point $u \in M$ and a subset of points $M' \subseteq M$, let $d(M', u) \equiv \min_{v \in M'}\{d(v, u)\}$ denote the distance between u and the closest point in M'. Let $d(\emptyset, u) \equiv \infty$. Let $B_u(r) \equiv \{v : d(u, v) \leq r\}$ denote the set of points within distance at most r to u. We slightly abuse the notation by letting the same symbol denote both a demand (facility) and the corresponding point of the metric space.

For each point $w \in M$, we are also given the cost f_w of opening a facility at w. We distinguish between the special case of *uniform* facility costs, where the cost of opening a facility, denoted f, is the same for all points, and the general case of *non-uniform* facility costs, where the cost of opening a facility at $w \in M$, denoted f_w, depends on w and there are no restrictions on the facility costs.

The demand sequence consists of points, which are not necessarily distinct. We only consider unit demands and allow multiple demands to be located at the same point. We use n to denote the total number of demands. Given a (multi)set D of demand points, we seek for a facility configuration $F \subseteq M$ that minimizes $\sum_{w \in F} f_w + \sum_{u \in D} d(F, u)$.

1.4 Roadmap

In Section 2, we formulate the algorithm for uniform facility costs. In Section 2.1 we introduce the notation required for the analysis. The main ideas behind the analysis are presented in Section 2.2. Then, the main properties of the final distance are established in Section 2.3, and the algorithm's facility and assignment costs are bounded in Sections 2.4 and 2.5 respectively. In Section 3, we present a generalization of the algorithm dealing with non-uniform facility costs. In Section 3.1, we discuss the main differences of the analysis from the special case of uniform facility costs and prove a basic property of the algorithm. Then, the algorithm's facility and assignment costs are bounded in Sections 3.2 and 3.3 respectively. We conclude with a summary and some directions for further research in Section 4.

2 The Algorithm for Uniform Facility Costs

The algorithm FFL (Fig. 1) maintains its *facility configuration*, denoted F, and its *facility replacement configuration* consisting of the *replacement radius* $m(w)$ of each facility $w \in F$. The algorithm is based on the notion of *final distance*. For every point p and facility w, the *final distance* of p to w, denoted $g(w, p)$, is equal to $d(w, p) + 2m(w)$. The *final distance* of p to the facility configuration F, denoted $g(F, p)$, is

$$g(F, p) = \min_{w \in F}\{g(w, p)\} = \min_{w \in F}\{d(w, p) + 2m(w)\}\,. \tag{1}$$

If F is the current facility configuration, we simply refer to $g(F, p)$ as the final distance of p. For each demand u, we sometimes refer to $g(F, u)$ as the *final assignment cost* of u. Intuitively, the current algorithm's configuration gives every point p the guarantee that there will exist a facility within p's final distance in the future.

When a new demand u arrives, it computes its final assignment cost $g(F, u)$ with respect to the current algorithm's configuration. Then u opens a new facility w located at the same point as u with

$F \leftarrow \emptyset$; for each new demand u: $\quad g(F, u) \leftarrow \min_{z \in F}\{d(z, u) + 2m(z)\}$; $\quad$ with probability $\min\{g(F, u)/(\alpha f), 1\}$ do $\quad\quad$ let w be the location of u; $\quad\quad$ updateConfiguration(F, w);	updateConfiguration(F, w) $\quad m(w) \leftarrow \min\{g(F, u), \alpha f\}/6$; $\quad$ for each $z \in F$ do $\quad\quad$ if $d(z, w) \leq m(z)$ then $\quad\quad\quad F \leftarrow F \setminus \{z\}$; $\quad F \leftarrow F \cup \{w\}$;

Fig. 1. The algorithm *Fast Facility Location* (FFL) parameterized by α.

probability $\min\{g(F, u)/(\alpha f), 1\}$, where α is a parameter trading off the facility against the assignment cost. The replacement radius $m(w)$ of the new facility is set to $\min\{g(F, u), \alpha f\}/6$. For every existing facility z, if w is included in z's replacement ball $B_z(m(z))$, z is *replaced by* w (i.e. z is removed from F when w opens). Intuitively, w replaces every existing facility whose removal from $F \cup \{w\}$ does not increase the final distance of any points.

Let $|F_{\max}|$ be the maximum number of facilities maintained by FFL. It is straightforward to implement FFL in $\mathrm{O}(n\,|F_{\max}|)$ time and $\mathrm{O}(|F_{\max}|)$ space. The remainder of this section is devoted to the proof of the following theorem.

Theorem 1. *For* $\alpha = \frac{19}{8}$*, the competitive ratio of* FFL *is at most* $13\frac{2}{3}$.

2.1 Preliminaries

For an arbitrary fixed sequence of demand points, we compare the algorithm's cost with the cost of a fixed optimal facility configuration, denoted F^*. To avoid confusing the algorithm's facilities with the optimal facilities in F^*, we use the term *optimal center*, or simply *center*, to refer to an optimal facility in F^* and the term *facility* to refer to an algorithm's facility in F.

In the optimal solution, each demand is assigned to the nearest center in F^*. For each demand u, we let c_u denote the optimal center to which u is assigned, let $d_u^* \equiv d(c_u, u) = d(F^*, u)$ denote the optimal assignment cost of u, let $\mathrm{Asg}^* = \sum_u d_u^*$ denote the optimal assignment cost, and let $\mathrm{Fac}^* = |F^*|\, f$ denote the optimal facility cost. For the analysis, we use the clustering induced by F^* to map the demands and the algorithm's facilities to the optimal centers. In particular, a demand u is mapped to c_u, i.e., the optimal center nearest to u. A facility w is mapped to the optimal center to which the demand opening w is mapped. We let c_w denote the optimal center to which the facility w is mapped[4]. By definition, if w is opened by a demand u, c_w is the same as c_u.

Every time we want to explicitly refer to the algorithm's configuration at the moment when a demand arrives (resp. facility opens), we use the demand's (resp. facility's) identifier as a subscript. E.g. for a demand u (resp. facility w), F_u (resp. F_w) denotes the facility configuration just before u arrives (resp. w opens). Similarly, $g(F_u, u)$ (resp. $g(F_u, c_u)$) denotes the final distance of u (resp. c_u) when u arrives. If a new demand u opens a new facility w, F_w is the same as F_u.

Let $u_1, \ldots, u_n$ be the demand sequence considered by FFL. For every $j \geq 1$, let $F_{u_{j+1}}$ denote FFL's facility configuration just after the demand u_j is considered. At that point, the facility cost of

[4] For uniform facility costs, w is located at the same point as the demand opening it and c_w is simply the optimal center nearest to w. However c_w may be different from the optimal center nearest to w in case of non-uniform facility costs.

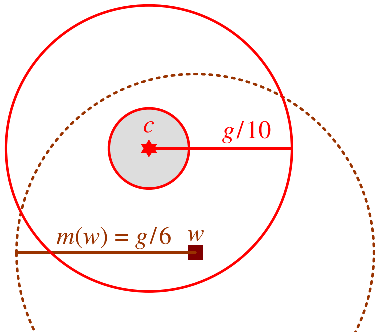

Fig. 2. Bounding the number of unsupported facilities. Let $\lambda = 10$, and let $g \equiv g(F_w, c)$ denote the final distance of c just before an unsupported facility w opens. Since w is unsupported, $d(w,c)$ is small compared to g (i.e. $d(w,c) < g/10$). As a result, $m(w) \approx g/6$ and $g(w,c) = d(w,c) + 2m(w) < g/2$. Hence, every unsupported facility opening after w lies in a ball of radius less than $g/20$ around c (the small grey ball) which is included in the replacement ball of w.

FFL is $\max_{1 \leq i \leq j}\{|F_{u_{i+1}}|\, f\}$ and the assignment cost of FFL is $\sum_{i=1}^{j} d(F_{u_{j+1}}, u_i)$. In the following, we prove that after the demand u_j is considered, FFL's expected cost is within a constant factor of the optimal facility cost and the optimal assignment cost for the demands $u_1, \ldots, u_j$.

2.2 A Sketch of the Analysis

Next we outline the proof of Theorem 1. Throughout the sketch of the analysis, we assume that the optimal solution consists of a single optimal center, denoted c. After explaining the role of the final distance in the analysis, we present the main ideas required for bounding the expected facility and assignment costs of FFL.

Final Distance. We prove that the final distance of every point p is non-increasing with time and that $B_p(g(F,p))$ will always contain a facility (cf. Proposition 1). Consequently, the (actual) assignment cost of a demand never exceeds its final assignment cost (with respect to the current or any of the past configurations). Thus we can ignore facility replacements and use the demands' final assignment cost to bound their assignment cost throughout the execution of the algorithm.

Facility Cost. The facilities are divided into *supported* (by the optimal solution) and *unsupported* ones. A facility opened by a demand u is unsupported if $d_u^* < \min\{\frac{1}{\lambda}\, g(F_u, c), \frac{\alpha}{\lambda+1}\, f\}$ and supported otherwise, where λ is a positive constant chosen sufficiently large.

By definition, the optimal assignment cost of a demand u opening a supported facility is large enough to compensate for u's expected contribution to the facility cost. More precisely, the expected contribution of every demand u to the cost of supported facilities is at most $\frac{\lambda+1}{\alpha}\, d_u^*$ and the expected cost of supported facilities is at most $\frac{\lambda+1}{\alpha}\mathrm{Asg}^*$ (cf. Lemma 1).

The parameter λ is chosen so that every demand u opening an unsupported facility w is sufficiently close to the optimal center c compared to the final distance of c. Therefore, w's replacement ball is sufficiently large to include every unsupported facility opening after w (cf. Fig. 2). Hence w is replaced no later than the moment when the first new unsupported facility opens (cf. Lemma 2). In general, the

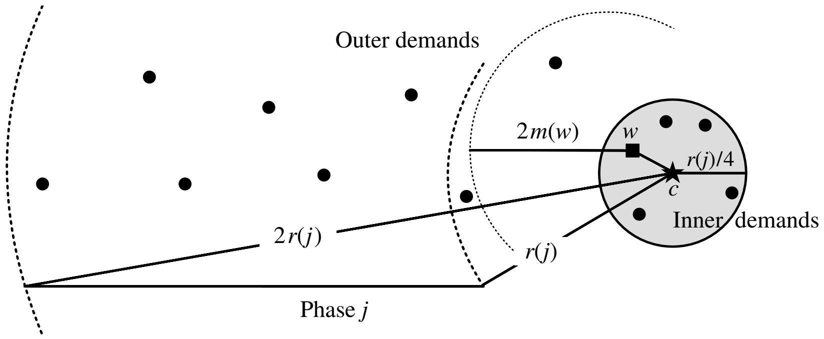

Fig. 3. Inner and outer demands in phase j (for $\mu = 4$). The optimal center c is in phase j as long as c's final distance is in $[r(j), 2r(j))$. A demand arriving in phase j is inner if it is included in the grey ball of radius $r(j)/4$ around c and outer otherwise. Let w be a new facility opened by an inner demand. Just before w opens, the final distance of w is less than $2r(j) + r(j)/4 = 9r(j)/4$. Thus, $m(w) < 9r(j)/24$ and $g(w, c) < r(j)/4 + 9r(j)/12 = r(j)$. As a result, phase j ends when w opens.

number of unsupported facilities is bounded by the number of optimal centers and their cost never exceeds Fac^*.

Putting the cost of supported and unsupported facilities together, we obtain that the expected facility cost is at most $\mathrm{Fac}^* + \frac{\lambda+1}{\alpha}\,\mathrm{Asg}^*$. We should highlight that the analysis treats every increase in the facility cost as an irrevocable decision as required by the scenario discussed in the introduction.

Assignment Cost. We break down the analysis of the assignment cost into *disjoint phases* according to the final distance of c (this is possible because c's final distance is non-increasing). For every integer j, let $r(j) = f/2^j$. The j-th phase lasts as long as the final distance of c is in $[r(j), 2r(j))$. After the beginning of phase j, there will always exist a facility within a distance of $2r(j)$ from c. We distinguish between *inner* and *outer* demands. A demand u arriving in phase j is inner if $d_u^* < r(j)/\mu$ and outer otherwise (cf. Fig. 3), where μ is a positive constant.

The parameter μ is chosen sufficiently large so that inner demands arriving in the current phase can be regarded as essentially located at the same point as c (cf. Fig. 3). Using this intuition, we prove that (i) the current phase ends as soon as an inner demand opens a new facility (cf. Proposition 3), and (ii) every inner demand arriving in phase j opens a new facility with probability at least $\frac{\mu-1}{\alpha\mu}\,2^{-j}$ (cf. Proposition 4). Therefore, the expected number of inner demands arriving in phase j is at most $\frac{\alpha\mu}{\mu-1}\,2^j$ and the expected number of inner demands arriving up to the end of phase j is at most $\frac{\alpha\mu}{\mu-1}\,2^{j+1}$ (cf. the proof of Lemma 3).

Inner demands arriving up to the end of phase j are charged with their final assignment cost with respect to any algorithm's configuration in phase j. As long as c is in phase j, the final assignment cost of every demand is at most $f/2^{j-1}$ plus its optimal assignment cost. Thus the expected final assignment cost (with respect to any configuration in phase j) of the inner demands arriving up to the end of phase j is at most $\frac{4\alpha\mu}{\mu-1}\,f$ plus their optimal assignment cost (cf. Lemma 3).

On the other hand, when an outer demand u arrives, its final assignment cost $g(F_u, u)$ is at most $(2\mu + 1)$ times its optimal assignment cost d_u^* (cf. Lemma 4). Hence by the properties of the final distance, the (actual) assignment cost of u never exceeds $(2\mu+1)d_u^*$. Putting the assignment cost of inner and outer demands together, we obtain that the algorithm's expected assignment cost is at most $\frac{4\alpha\mu}{\mu-1}\,\mathrm{Fac}^* + (2\mu+1)\mathrm{Asg}^*$.

Remark. In the sketch of the analysis above, it becomes clear that the parameter α trades off the facility against the assignment cost. Since the number of facilities is directly related to the computational resources used by FFL, the parameter α can also be regarded as a tradeoff between the computational efficiency and the quality of approximation.

2.3 Final Distance

Proposition 1. *Let F be the current facility configuration and let F' be any future facility configuration. For every point p, $g(F',p) \le g(F,p)$ and $d(F',p) \le g(F,p)$.*

Proof. Let w be the facility in F of minimum final distance to p, i.e. $g(w,p) = g(F,p)$. The proposition is true as long as w is open. We prove that if w is replaced by a new facility w', then $g(w',p) \le g(w,p)$. We first show that $2m(w') \le m(w)$:

$$2m(w') \le \tfrac{1}{3}\,g(w,w') = \tfrac{1}{3}\,[d(w,w') + 2m(w)] \le m(w) \tag{2}$$

The first inequality holds because $m(w') \le g(F_{w'}, w')/6$, since w' and the demand opening it are located at the same point, and $w \in F_{w'}$, since w is replaced by w'. The last inequality follows from $d(w,w') \le m(w)$, because w is replaced by w'. Using (2) and $d(w,w') \le m(w)$, we obtain that

$$g(w',p) = d(w',p) + 2m(w') \le d(w,p) + d(w,w') + m(w) \le d(w,p) + 2m(w) = g(w,p)$$

Consequently, the final distance of p does not increase when w is replaced by w'. In addition, $d(w',p) \le g(w',p) \le g(F,p)$ and $B_p(g(F,p))$ still contains a facility. □

In the following, we repeatedly use the equivalent of the triangle inequality for the final distance without referring to the proposition below.

Proposition 2. *For every facility configuration F and points p_1, p_2, $g(F,p_1) \le g(F,p_2) + d(p_1,p_2)$.*

Proof. Let w be the facility in F of minimum final distance to p_2, i.e. $g(F,p_2) = g(w,p_2)$. Then,

$$\begin{aligned} g(F,p_1) \le g(w,p_1) = d(w,p_1) + 2m(w) &\le d(p_1,p_2) + d(w,p_2) + 2m(w) \\ &= d(p_1,p_2) + g(w,p_2) = g(F,p_2) + d(p_1,p_2)\,, \end{aligned}$$

where the last equality follows from the definition of w. □

2.4 Facility Cost

Let u be a demand opening a new facility w. We recall that the facility w is called *unsupported* if $d_u^* < \min\{\frac{1}{\lambda}\,g(F_u,c_u), \frac{\alpha}{\lambda+1}\,f\}$ and *supported* otherwise. Since u and w are located at the same point and are mapped to the same optimal center, $d(w,c_w) \equiv d_u^*$. Hence, it is equivalent to call the facility w unsupported if $d(w,c_w) < \min\{\frac{1}{\lambda}\,g(F_w,c_w), \frac{\alpha}{\lambda+1}\,f\}$ and supported otherwise.

The following lemma charges the expected cost of supported facilities to the optimal assignment cost of the demands opening them.

Lemma 1. *The expected cost of supported facilities is at most* $\frac{\lambda+1}{\alpha}\mathrm{Asg}^*$.

Proof. We show that the expected contribution of each demand u to the cost of supported facilities is at most $\frac{\lambda+1}{\alpha}\, d_u^*$. A new demand u opens a new facility with probability $\min\{g(F_u, u)/(\alpha f), 1\}$. The new facility is supported only if $d_u^* \geq \min\{\frac{1}{\lambda}\, g(F_u, c_u), \frac{\alpha}{\lambda+1}\, f\}$. If $d_u^* \geq \frac{\alpha}{\lambda+1}\, f$, the contribution of u to the facility cost is at most $f \leq \frac{\lambda+1}{\alpha}\, d_u^*$. If $d_u^* \geq \frac{1}{\lambda}\, g(F_u, c_u)$, the expected contribution of u to the facility cost is at most

$$[g(F_u, u)/(\alpha f)]\, f = \frac{1}{\alpha}\, g(F_u, u) \leq \frac{1}{\alpha}\, [g(F_u, c_u) + d_u^*] \leq \frac{\lambda+1}{\alpha}\, d_u^*$$

Summing up over all demands u with $d_u^* \geq \min\{\frac{1}{\lambda}\, g(F_u, c_u), \frac{\alpha}{\lambda+1}\, f\}$ and using linearity of expectation, we obtain an upper bound of $\frac{\lambda+1}{\alpha}\mathrm{Asg}^*$ on the expected cost of supported facilities. □

The following lemma implies that there always exists at most one unsupported facility mapped to each optimal center. Thus, the total cost of supported facilities is at most Fac^*.

Lemma 2. *Let w be an unsupported facility, and let w' be a new unsupported facility mapped to c_w (i.e. to the same center as w). Then $d(w, w') \leq m(w)$ and w is replaced no later than w''s opening.*

Proof. We first show that for every unsupported facility w, $d(w, c_w) < \frac{6}{\lambda-1}\, m(w)$ (see also Fig. 2). By the definition of unsupported facilities, $d(w, c_w) < \min\{\frac{1}{\lambda}\, g(F_w, c_w), \frac{\alpha}{\lambda+1}\, f\}$. Also we recall that $m(w) = \min\{g(F_w, c_w), \alpha f\}/6$. If $m(w) = \alpha f/6$, we use $d(w, c_w) < \frac{\alpha}{\lambda+1}\, f$ and obtain that

$$d(w, c_w) < \frac{\alpha}{\lambda+1}\, f = \frac{6}{\lambda+1}\, m(w) < \frac{6}{\lambda-1}\, m(w)\,.$$

If $m(w) = g(F_w, w)/6$, we use $g(F_w, c_w) > \lambda\, d(w, c_w)$ and obtain that

$$m(w) \geq \frac{1}{6}\, [g(F_w, c_w) - d(w, c_w)] > \frac{1}{6}\, [\lambda\, d(w, c_w) - d(w, c_w)] = \frac{\lambda-1}{6}\, d(w, c_w)\,.$$

We conclude the proof by showing that the inequality $d(w, c_w) < \frac{6}{\lambda-1} m(w)$ implies the lemma:

$$\begin{aligned} d(w, w') &\leq d(w, c_w) + d(w', c_w) \\ &< d(w, c_w) + \tfrac{1}{\lambda}\, [d(w, c_w) + 2m(w)] && \text{as } d(w', c_w) < \tfrac{1}{\lambda}\, g(F_{w'}, c_w) \leq \tfrac{1}{\lambda}\, g(w, c_w) \\ &< [\tfrac{\lambda+1}{\lambda}\, \tfrac{6}{\lambda-1} + \tfrac{2}{\lambda}]\, m(w) \leq m(w) && \text{as } d(w, c_w) < \tfrac{6}{\lambda-1}\, m(w) \end{aligned}$$

The second inequality holds because w' is an unsupported facility mapped to c_w and the final distance of c_w is non-increasing. The last inequality holds for every $\lambda \geq 9.5$. □

Lemmas 1 and 2 imply that the expected facility cost never exceeds $\mathrm{Fac}^* + \frac{\lambda+1}{\alpha}\mathrm{Asg}^*$.

2.5 Assignment Cost

We break down the analysis of the assignment cost of the demands mapped to every optimal center c into disjoint phases according to the final distance of c. For every integer $j \geq -2$, let $r(j) = f/2^j$, and let $r(-3) = \infty$. The j-th phase starts just after the final distance of c becomes less than $r(j-1)$ and ends as soon as the final distance of c becomes less than $r(j)$. We use the convention that phase -2 starts when the first demand arrives. The optimal center c is in phase j as long as $g(F, c) \in [r(j), r(j-1))$. For every $j \geq -1$, $g(F, c) < 2r(j)$ after c enters phase j.

A demand u arrives in phase j if $g(F_u, c_u) \in [r(j), r(j-1))$, i.e. when u arrives, the optimal center to which u is mapped is in phase j. A demand u arriving in phase j is called *inner* if $d_u^* < r(j)/\mu$ and *outer* otherwise. We use the convention that every demand arriving in phase -2 is inner.

Inner Demands. Inner demands are charged with their final assignment cost with respect to any algorithm's configuration in the current phase. We first show that the current phase ends as soon as an inner demand opens a new facility (cf. Proposition 3, see also Fig. 3). Then we bound from below the probability that an inner demand opens a new facility and concludes the current phase (cf. Proposition 4). Finally, we obtain an upper bound on the expected final assignment cost of inner demands (cf. Lemma 3).

Proposition 3. *Let u be an inner demand which arrives in phase j and opens a new facility w. Then, $g(w, c_u) < r(j)$ and the j-th phase ends as soon as w opens.*

Proof. If u arrives in phase $j \geq -1$, $g(F_u, c_u) < 2r(j)$. Since u and w are located at the same point, $d(w, c_u) \equiv d^*_u < r(j)/\mu$. Therefore,

$$\begin{aligned} g(w, c_u) &= d(w, c_u) + 2m(w) \\ &\leq d(w, c_u) + \tfrac{1}{3}\left[g(F_u, c_u) + d^*_u\right] && \text{as } m(w) \leq g(F_u, u)/6 \\ &< \tfrac{4}{3}\, d(w, c_u) + \tfrac{2}{3}\, r(j) && \text{as } g(F_u, c_u) < 2r(j) \text{ and } d(w, c_u) = d^*_u \\ &< \left[\tfrac{4}{3}\,\tfrac{1}{\mu} + \tfrac{2}{3}\right] r(j) && \text{as } d^*_u < r(j)/\mu \text{ and } g(F_u, c_u) < 2r(j) \\ &\leq r(j) && \text{for every } \mu \geq 4 \end{aligned}$$

If u arrives in phase -2, $g(w, c_u) \leq (1 + \frac{\alpha}{3})f$ because $d(w, c_u) \equiv d^*_u \leq f$ and $m(w) \leq \alpha f/6$. Hence, $g(w, c_u) < 4f$ for every $\alpha < 9$. In both cases, the final distance of c_u becomes less than $r(j)$ and the j-th phase ends when w opens. □

Proposition 4. *Every inner demand arriving in phase j opens a new facility with probability at least $\min\{\frac{\mu-1}{\alpha\mu}\, 2^{-j}, 1\}$.*

Proof. If u is an inner demand arriving in phase j, $g(F_u, c_u) \geq r(j) = f/2^j$ and $d^*_u < r(j)/\mu$. Then,

$$g(F_u, u)/(\alpha f) \geq (g(F_u, c_u) - d^*_u)/(\alpha f) > \tfrac{\mu-1}{\mu}\, r(j)/(\alpha f) = \tfrac{\mu-1}{\alpha\mu}\, 2^{-j} \qquad \square$$

Lemma 3. *For every optimal center c, the expected final assignment cost of the inner demands mapped to c and considered up to the end of the current phase is at most $\frac{4\alpha\mu}{\mu-1} f$ plus their optimal assignment cost.*

Proof. By Propositions 3 and 4, every inner demand arriving in phase j opens a new facility and concludes the current phase with probability at least $\min\{\frac{\mu-1}{\alpha\mu}\, 2^{-j}, 1\}$. Let us assume that $\mu \geq 4$ (as required by Proposition 3) and that $\alpha \in [2, 3]$ (just for simplicity).

If a demand mapped to c arrives in phase -2, it concludes phase -2 with certainty (note that $\frac{4(\mu-1)}{\alpha\mu} \geq 1$ for every $\alpha \leq 3$ and $\mu \geq 4$). Hence, there is at most one (inner) demand arriving in phase -2. In addition, phase -2 has completed when the assignment cost of the first demand mapped to c is considered.

For every phase $j \geq -1$, the probability that an inner demand opens a new facility and concludes phase j is at least $\frac{\mu-1}{\alpha\mu}\, 2^{-j}$ (note that $\frac{2(\mu-1)}{\alpha\mu} \leq 1$ for all $\alpha \geq 2$). Therefore, the expected number of inner demands arriving in phase j is at most $\frac{\alpha\mu}{\mu-1}\, 2^j$.

Let $j \geq -1$ be the current phase of c. By linearity of expectation, the expected number of inner demands arriving up to the end of phase j is at most

$$1 + \sum_{i=-1}^{j} \tfrac{\alpha\mu}{\mu-1} \, 2^i = 1 + \tfrac{\alpha\mu}{\mu-1} \, (2^{j+1} - \tfrac{1}{2}) \leq \tfrac{\alpha\mu}{\mu-1} \, 2^{j+1} \tag{3}$$

where, for the last inequality we use that $1 \leq \frac{\alpha\mu}{2(\mu-1)}$. After c enters phase j, the final assignment cost of a demand u mapped to c (with respect to any configuration F in phase j) is

$$g(F, u) \leq g(F, c) + d^*_u < 2r(j) + d^*_u = f/2^{j-1} + d^*_u$$

Since the expected number of inner demands mapped to c and considered up to the end of phase j is at most $\frac{\alpha\mu}{\mu-1} \, 2^{j+1}$, their expected final assignment cost (with respect to any configuration in phase j) never exceeds $\frac{4\alpha\mu}{\mu-1} f$ plus their optimal assignment cost. □

Outer Demands. Outer demands are charged with their final assignment cost with respect to the algorithm's configuration at their arrival time.

Lemma 4. *The final assignment cost of every outer demand u is at most $(2\mu + 1)d^*_u$.*

Proof. Let $j \geq -1$ be the phase in which u arrives as an outer demand. Then,

$$g(F_u, u) \leq g(F_u, c_u) + d^*_u < 2r(j) + d^*_u \leq (2\mu + 1)d^*_u \ ,$$

where we use that $g(F_u, c_u) < 2r(j)$, because c_u is in phase $j \geq -1$ when u arrives, and $d^*_u \geq r(j)/\mu$, because u is an outer demand. □

Combining Lemmas 3 and 4 and Proposition 1, we obtain that the algorithm's expected assignment cost is at most $\frac{4\alpha\mu}{\mu-1} \, \mathrm{Fac}^* + (2\mu + 1)\mathrm{Asg}^*$.

2.6 Concluding the Proof of Theorem 1

For every $\lambda \geq 9.5$, $\mu \geq 4$, and $\alpha \in [2, 3]$, the expected cost of FFL is at most

$$(\tfrac{4\alpha\mu}{\mu-1} + 1) \, \mathrm{Fac}^* + (\tfrac{\lambda+1}{\alpha} + 2\mu + 1) \, \mathrm{Asg}^* \leq \max\{\tfrac{4\alpha\mu}{\mu-1} + 1, \tfrac{\lambda+1}{\alpha} + 2\mu + 1\} \, (\mathrm{Fac}^* + \mathrm{Asg}^*)$$

Using $\alpha = \frac{19}{8}$, $\lambda = 10$, and $\mu = 4$, we obtain a competitive ratio of $13\frac{2}{3}$. Considering an appropriate number of additional phases, we can establish the same upper bound for every $\alpha > 2(\mu - 1)/\mu$. Furthermore, the competitive ratio can be improved to 12 by optimizing the definition of $r(j)$.

3 The Algorithm for Non-Uniform Facility Costs

Then, we present a generalized version of FFL dealing with non-uniform facility costs. As before, the algorithm FNFL (Fig. 4) maintains its facility configuration F and the replacement radius $m(w)$ of each facility $w \in F$.

$F \leftarrow \emptyset$;
for each new demand u:
$g(F,u) \leftarrow \min_{z \in F}\{d(z,u) + 2m(z)\}$;
let w be the facility in $B_u(\frac{1}{\lambda-1}\, g(F,u))$
that minimizes $f_w + d(w,u)$;
with probability $\min\{g(F,u)/(\alpha f_w), 1\}$
do updateConfiguration(F, w);
	updateConfiguration(F, w)
	$g(F,w) \leftarrow \min_{z \in F}\{d(z,w) + 2m(z)\}$;
	$m(w) \leftarrow \min\{g(F,w), 3f_w + \frac{\lambda-3}{2}\, d(w,u)\}/6$;
	for each $z \in F$ do
	if $d(z,w) \leq m(z)$ then
	$F \leftarrow F \setminus \{z\}$;
	$F \leftarrow F \cup \{w\}$;

Fig. 4. The algorithm Fast Non-Uniform Facility Location (FNFL) parameterized by α and λ.

Every new demand u computes its final assignment cost $g(F,u)$ with respect to the current algorithm's configuration. In case of non-uniform facility costs, opening a facility at the same point as u may be too expensive or even infeasible. Hence u looks for the cheapest alternative in its neighborhood. More precisely, u finds the facility w in $B_u(\frac{1}{\lambda-1}\, g(F,u))$ that minimizes $f_w + d(w,u)$, where the parameter λ is chosen so that w is located sufficiently close to u. Then u opens a new facility at w with probability $\min\{g(F,u)/(\alpha f_w), 1\}$. Taking non-uniform facility costs into account, we set the replacement radius $m(w)$ of the new facility to $\min\{g(F,w), 3f_w + \frac{\lambda-3}{2}\, d(w,u)\}/6$. Similarly to FFL, every existing facility z that includes w in its replacement ball $B_z(m(z))$ is replaced by w.

Let $|M|$ be the number of potential facility locations. FNFL can be implemented in $\mathrm{O}(n\,|M|)$ time and $\mathrm{O}(|M|)$ space. The remainder of this section is devoted to proof of the following theorem.

Theorem 2. *For* $\lambda = 20$ *and* $\alpha = \frac{38}{7}$*, the competitive ratio of* FNFL *is less than* 48.6.

3.1 Preliminaries

Outline. The structure of the proof is similar to that of FFL (see also Section 2.2). However, many important details become significantly more complicated when non-uniform facility costs are taken into account. A crucial difference is a new facility may be located far away from the demand opening it. Therefore, in the analysis of the facility cost, the definition of supported and unsupported facilities must change substantially. As a result, bounding the cost of unsupported facilities becomes easier (cf. Lemma 5) and bounding the expected cost of supported facilities becomes the most interesting (and difficult) case (cf. Lemma 6).

As for the assignment cost, establishing that the current phase ends when an inner demand opens a new facility (cf. Proposition 6) and lower bounding the probability that an inner demand opens a new facility (cf. Proposition 7) become non-trivial. To make sure that there is at most one inner demand arriving in the initial phase, we must also consider outer demands arriving in it. Hence, bounding the final assignment cost of outer demands becomes a bit more interesting (cf. Lemma 8).

Notation. We use the same notation as in the analysis of FFL (see also Section 2.1). As for the facility costs, f_p denotes the cost of opening a new facility at point p. The optimal facility cost is $\mathrm{Fac}^* = \sum_{c \in F^*} f_c$. After the demand u_j is considered, the facility cost of FNFL is $\max_{1 \leq i \leq j}\{\sum_{p \in F_{u_{i+1}}} f_p\}$. For every demand u, f_{c_u} denotes the facility cost of the optimal center to which u is mapped, denoted

c_u . Also we recall that if u opens a new facility w, the optimal center to which w is mapped, denoted c_w, is the same as c_u .

Basic Properties. First of all, we observe that Propositions 1 and 2 hold for FNFL too. In particular, Proposition 1 holds because for every facility w, $m(w)$ is still no greater than $g(F_w, w)/6$. Hence, we use the notions of final distance and final assignment cost as in the proof of Theorem 1.

A basic property of FNFL is that for every demand u located close enough to c_u (formally, if $d_u^* < g(F_u, c_u)/\lambda$), the optimal center c_u is included in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ and is considered for opening by u. Consequently, the cost of a new facility w opened by a demand u with $d_u^* < g(F_u, c_u)/\lambda$ is no greater than $f_{c_u} + d_u^*$. The following proposition establishes this and some other useful properties of a facility opened by a demand located sufficiently close to an optimal center.

Proposition 5. *Let u be a new demand with $d_u^* < g(F_u, c_u)/\lambda$, and let w be the facility in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ that minimizes $f_w + d(w, u)$. Then,*
(a) $d(w, c_w) < \frac{2}{\lambda-1}\, g(F_u, c_u) = \frac{2}{\lambda-1}\, g(F_w, c_w)$,
(b) $c_u \in B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ and $f_w + d(w, u) \leq f_{c_u} + d_u^*$, and
(c) $g(F_w, w) > \frac{\lambda-3}{2}\, d(w, c_w)$.

Proof. By definition, u and w are mapped to the same optimal center (i.e. c_u is the same as c_w) and $d(w, c_u) = d(w, c_w)$. In addition, the algorithm's configuration when u arrives is identical to the algorithm's configuration just before w opens (i.e. F_u is the same as F_w). Hence $g(F_u, c_u) = g(F_w, c_w)$. To conclude the proof of (a), we observe that

$$\begin{aligned} d(w, c_w) = d(w, c_u) &\leq d(w, u) + d_u^* \\ &\leq \tfrac{1}{\lambda-1}\, g(F_u, u) + d_u^* && \text{as } w \in B_u(\tfrac{1}{\lambda-1}\, g(F_u, u)) \\ &\leq \tfrac{1}{\lambda-1}\, (g(F_u, c_u) + d_u^*) + d_u^* \\ &< (\tfrac{1}{\lambda-1} + \tfrac{\lambda}{\lambda-1}\, \tfrac{1}{\lambda})\, g(F_u, c_u) && \text{as } d_u^* < g(F_u, c_u)/\lambda \\ &= \tfrac{2}{\lambda-1}\, g(F_u, c_u) \end{aligned}$$

For (b), we first observe that $d_u^* < g(F_u, c_u)/\lambda$ implies that

$$g(F_u, u) \geq g(F_u, c_u) - d_u^* > (\lambda - 1) d_u^* \ . \tag{4}$$

Therefore, c_u is included in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$. The second part of (b) holds because w is the facility in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ that minimizes $f_w + d(w, u)$.

For (c), we observe that

$$g(F_w, w) \geq g(F_w, c_w) - d(w, c_w) > (\tfrac{\lambda-1}{2} - 1)\, d(w, c_w) = \tfrac{\lambda-3}{2}\, d(w, c_w)$$

where the second inequality follows from (a). □

3.2 Facility Cost

As before, the analysis of the facility cost is based on the notions of supported and unsupported facilities, which are now defined in a different way. Let u be a demand opening a new facility w. The facility w is *unsupported* if $d_u^* < g(F_u, c_u)/\lambda$ and $m(w) \geq \frac{\lambda-3}{12}\, d(w, c_w)$, and *supported* otherwise. Intuitively, the new facility w is unsupported only if the demand u is sufficiently close to c_u (i.e. if $d_u^* < g(F_u, c_u)/\lambda$) *and* its replacement radius is sufficiently large compared to $d(w, c_w)$ (i.e. if $m(w) \geq \frac{\lambda-3}{12}\, d(w, c_w)$). This definition implies easily that there always exists at most one unsupported facility mapped to each optimal center.

Lemma 5. *Let w be an unsupported facility, and let w' be a new unsupported facility mapped to c_w (i.e. the same center as w). Then $d(w, w') \leq m(w)$ and w is replaced no later than w''s opening.*

Proof. Since w' is an unsupported facility, the demand opening w' fulfills the hypothesis of Proposition 5. Thus, $d(w', c_w) < \frac{2}{\lambda-1}\, g(F_{w'}, c_w) \leq \frac{2}{\lambda-1}\, g(w, c_w)$ by Proposition 5.a. Therefore,

$$\begin{aligned}
d(w, w') &\leq d(w, c_w) + d(w', c_w) \\
&< d(w, c_w) + \tfrac{2}{\lambda-1}\, g(w, c_w) && \text{as } d(w', c_w) < \tfrac{2}{\lambda-1}\, g(w, c_w) \\
&= d(w, c_w) + \tfrac{2}{\lambda-1}\, [d(w, c_w) + 2m(w)] && \text{as } g(w, c_w) \equiv d(w, c_w) + 2m(w) \\
&\leq [\tfrac{\lambda+1}{\lambda-1}\, \tfrac{12}{\lambda-3} + \tfrac{4}{\lambda-1}]\, m(w) && \text{as } d(w, c_w) \leq \tfrac{12}{\lambda-3}\, m(w) \\
&\leq m(w) && \text{for every } \lambda \geq 20
\end{aligned}$$

Consequently, $d(w, w') \leq m(w)$ and w will have been replaced by the moment when w' opens. □

If the demand u is not very close to c_u (i.e. if $d_u^* \geq g(F_u, c_u)/\lambda$), its expected contribution to the cost of supported facilities is charged to its optimal assignment cost d_u^* (see also the proof of Lemma 1). On the other hand, even if u is located very close to c_u (i.e. if $d_u^* < g(F_u, c_u)/\lambda$), the new facility w may be located far away from u (and c_u) and its replacement ball may not be sufficiently large compared to $d(w, c_u)$ (i.e. $m(w) < \frac{\lambda-3}{12}\, d(w, c_u)$). Then, w is classified as a supported facility and its opening cost must be charged to the optimal assignment cost of u. In this case, the contribution of $d(w, u)$ to w's replacement radius plays a crucial role in charging w's opening cost to d_u^*.

Lemma 6. *For every demand u which can open a supported facility, the expected contribution of u to the facility cost is at most $\frac{(\lambda+1)}{\alpha}\, d_u^*$.*

Proof. Let u be a new demand and let w be the facility in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ that minimizes $f_w + d(w, u)$. The demand u opens w with probability $\min\{g(F_u, u)/(\alpha f_w), 1\}$. If w opens, it is a supported facility only if either $d_u^* \geq g(F_u, c_u)/\lambda$ or $m(w) < \frac{\lambda-3}{12}\, d(w, c_w)$.

If $d_u^* \geq g(F_u, c_u)/\lambda$, the expected contribution of u to the facility cost is

$$\min\{g(F_u, u)/(\alpha f_w), 1\}\, f_w \leq g(F_u, u)/\alpha \leq [g(F_u, c_u) + d_u^*]/\alpha \leq \tfrac{\lambda+1}{\alpha}\, d_u^*$$

The most interesting case is when $d_u^* < g(F_u, c_u)/\lambda$ and w is a supported facility. By the definition of supported facilities, $m(w) < \frac{\lambda-3}{12}\, d(w, c_w)$. In addition, $g(F_w, w) > \frac{\lambda-3}{2}\, d(w, c_w)$ by Proposition 5.c. Therefore, if $d_u^* < g(F_u, c_u)/\lambda$ and w is a supported facility,

$$m(w) < \tfrac{\lambda-3}{12}\, d(w, c_w) < g(F_w, w)/6\,.$$

Since $m(w) = \min\{g(F_w, w)/6, f_w/2 + \frac{\lambda-3}{12}\, d(w, u)\}$, we have that

$$m(w) = f_w/2 + \tfrac{\lambda-3}{12}\, d(w, u) < \tfrac{\lambda-3}{12}\, d(w, c_w) \tag{5}$$

Using (5), we conclude that

$$f_w < \tfrac{\lambda-3}{6}\,[d(w,u) + d_u^*] - \tfrac{\lambda-3}{6}\,d(w,u) = \tfrac{\lambda-3}{6}\,d_u^*\,,$$

where we use that c_u coincides with c_w and $d(w, c_w) = d(u, c_u) = d_u^*$.

Consequently, if $d_u^* < g(F_u, c_u)/\lambda$ and u opens a supported facility, its contribution to the facility cost is at most $\frac{\lambda-3}{6}\,d_u^* \leq \frac{\lambda+1}{\alpha}\,d_u^*$, for every $\alpha \leq \frac{6(\lambda+1)}{\lambda-3}$. □

Lemma 5 implies that there always exists at most one unsupported facility mapped to each optimal center. Since a demand u opening an unsupported facility w has $d_u^* < g(F_u, c_u)/\lambda$, the cost of w is at most $f_{c_u} + d_u^*$ by Proposition 5.b. In other words, the opening cost of an unsupported facility w mapped to an optimal center c is at most f_c plus the optimal assignment cost of the demand opening w. Using Lemma 5, we obtain that the total cost of unsupported facilities is at most Fac^* plus the optimal assignment cost of the demands opening them. By Lemma 6, the expected cost of supported facilities is no greater than $\frac{(\lambda+1)}{\alpha}$ times the optimal assignment cost of the remaining demands. Therefore, the algorithm's expected facility cost is at most $\mathrm{Fac}^* + \frac{(\lambda+1)}{\alpha}\mathrm{Asg}^*$.

3.3 Assignment Cost

As before, we break down the analysis of the assignment cost into disjoint phases. The difference is that the definition of phases now takes into account the opening cost of each optimal facility. Formally, for every optimal center c and every integer $j \geq -3$, let $r_c(j) = f_c/2^j$, and let $r_c(-4) = \infty$. We say that the optimal center c is in phase j as long as $g(F, c) \in [r_c(j), r_c(j-1))$. A demand u arriving in phase j is *inner* if $d_u^* < r_{c_u}(j)/\lambda$ and *outer* otherwise. In contrast to the analysis of FFL, outer demands may also arrive in the initial phase -3.

Inner Demands. For every optimal center c, we bound the expected final assignment cost of the inner demands mapped to c and considered up to the end of the current phase. More precisely, we prove that the expected final assignment cost of these demands (with respect to any configuration in the current phase) is at most $\frac{4\alpha(\lambda+8)}{\lambda-1} f_c$ plus their optimal assignment cost. We start by showing that the current phase ends as soon as an inner demand opens a new facility.

Proposition 6. *Let u be an inner demand which arrives in phase j and opens a new facility w. Then, $g(w, c_u) < r_{c_u}(j)$ and the j-th phase ends.*

Proof. Let j be the phase of c_u in which u arrives. Since the demand u is inner, $d_u^* < r_{c_u}(j)/\lambda \leq g(F_u, c_u)/\lambda$. Using Proposition 5, we obtain that $f_w + d(w, u) \leq f_{c_u} + d_u^*$ and $d(w, c_w) < \frac{2}{\lambda-1}\,g(F_u, c_u)$. Also we recall that the demand u and the facility w are mapped to the same optimal center (i.e. c_u is the same as c_w) and $d(w, c_u) = d(w, c_w)$.

If the current phase is -3, the final distance of c_u after w opens is

$$\begin{aligned}
g(w, c_u) &= d(w, c_u) + 2m(w) && \\
&\leq d(w,u) + d_u^* + f_w + \tfrac{\lambda-3}{6}\,d(w,u) && \text{as } m(w) \leq f_w/2 + \tfrac{\lambda-3}{12}\,d(w,u) \\
&\leq d_u^* + \tfrac{\lambda+3}{6}\,[f_w + d(w,u)] && \text{for every } \lambda \geq 3 \\
&\leq d_u^* + \tfrac{\lambda+3}{6}\,[f_{c_u} + d_u^*] && \text{as } f_w + d(w,u) \leq f_{c_u} + d_u^* \\
&< [\tfrac{8}{\lambda} + \tfrac{\lambda+3}{6}\,\tfrac{\lambda+8}{\lambda}]\, f_{c_u} && \text{as } d_u^* < (f_{c_u}/2^{-3})/\lambda = 8f_{c_u}/\lambda \\
&\leq 8 f_{c_u} && \text{for every } \lambda \leq 34
\end{aligned}$$

Hence, $g(w, c_u) < 8f_{c_u}$ and phase -3 ends as soon as w opens.

For every phase $j \geq -2$, $g(F_u, c_u) < 2r_{c_u}(j)$. After w opens, the final distance of c_u is

$$\begin{aligned} g(w, c_u) &= d(w, c_u) + 2m(w) \\ &\leq d(w, c_u) + \tfrac{1}{3}\,[g(F_u, c_u) + d(w, c_u)] && \text{as } m(w) \leq g(F_u, w)/6 \\ &< (\tfrac{4}{3}\,\tfrac{2}{\lambda-1} + \tfrac{1}{3})\, g(F_u, c_u) && \text{as } d(w, c_u) < \tfrac{2}{\lambda-1}\, g(F_u, c_u) \\ &< \tfrac{2(\lambda+7)}{3(\lambda-1)}\, r_{c_u}(j) && \text{as } g(F_u, c_u) < 2r_{c_u}(j) \\ &\leq r_{c_u}(j) && \text{for every } \lambda \geq 17 \end{aligned}$$

Therefore, $g(w, c_u) < r_{c_u}(j)$ and the j-th phase ends as soon as w opens. □

Then, we obtain a lower bound on the probability that an inner demand arriving in phase j opens a new facility and concludes the current phase.

Proposition 7. *Every inner demand arriving in phase j opens a new facility with probability at least $\min\{\frac{\lambda-1}{\alpha(\lambda+8)}\, 2^{-j}, 1\}$.*

Proof. Let j be the phase of c_u in which an inner demand u arrives. Since u is inner, $d^*_u < r_{c_u}(j)/\lambda \leq g(F_u, c_u)/\lambda$. Since we only consider phases $j \geq -3$, $r_{c_u}(j) \leq 8f_{c_u}$ and $d^*_u < 8f_{c_u}/\lambda$.

Let w be the facility in $B_u(\frac{1}{\lambda-1}\, g(F_u, u))$ of minimum $f_w + d(w, u)$. By Proposition 5.b,

$$f_w \leq f_w + d(w, u) \leq f_{c_u} + d^*_u < \tfrac{\lambda+8}{\lambda} f_{c_u} \tag{6}$$

The demand u opens a new facility with probability $\min\{g(F_u, u)/(\alpha f_w), 1\}$. The proposition follows from

$$\frac{g(F_u, u)}{\alpha f_w} \geq \frac{g(F_u, c_u) - d^*_u}{\alpha f_w} > \frac{\frac{\lambda-1}{\lambda}\, r_{c_u}(j)}{\alpha f_w} > \frac{\frac{\lambda-1}{\lambda}\, r_{c_u}(j)}{\frac{\alpha(\lambda+8)}{\lambda}\, f_{c_u}} = \tfrac{\lambda-1}{\alpha(\lambda+8)}\, 2^{-j}$$

The second inequality holds because $g(F_u, c_u) \geq r_{c_u}(j)$ and $d^*_u < r_{c_u}(j)/\lambda$. The third inequality follows from (6). For the ultimate equality, we use $r_{c_u}(j) = f_{c_u}/2^j$. □

We are now ready to bound the expected final assignment cost of inner demands. The proof of the following lemma is similar to that of Lemma 3.

Lemma 7. *For every optimal center c, the expected final assignment cost of the inner demands mapped to c and considered up to the end of the current phase is at most $\frac{4\alpha(\lambda+8)}{\lambda-1} f_c$ plus their optimal assignment cost.*

Proof. We have shown that every inner demand arriving in phase j opens a new facility and concludes the current phase with probability at least $\min\{\frac{\lambda-1}{\alpha(\lambda+8)}\, 2^{-j}, 1\}$.

Let $\alpha \leq \frac{8(\lambda-1)}{\lambda+8}$. If the first inner demand of c arrives in phase -3, it opens a new facility and concludes phase -3 with certainty. Consequently, at most one inner demand of c arrives before phase -2 and its final assignment cost is considered after the beginning of phase -2.

For every phase $j \geq -2$, $\frac{\lambda-1}{\alpha(\lambda+8)}\, 2^{-j} \leq 1$, provided that $\alpha \geq \frac{4(\lambda-1)}{\lambda+8}$. Therefore, every inner demand of c arriving in phase j opens a new facility and concludes phase j with probability at least $\frac{\lambda-1}{\alpha(\lambda+8)}\, 2^{-j}$. Hence the expected number of inner demands arriving in phase j is at most of $\frac{\alpha(\lambda+8)}{\lambda-1}\, 2^j$.

Let $j \geq -2$ be the current phase of c. By linearity of expectation, the expected number of inner demands arriving up to the end of phase j is at most $\frac{\alpha(\lambda+8)}{\lambda-1} 2^{j+1}$. After c enter phase j, the final assignment cost of a demand u mapped to c (with respect to any configuration in phase j) is at most $f_c/2^{j-1} + d_u^*$. Since the expected number of inner demands mapped to c and considered up to the end of phase j is at most $\frac{\alpha(\lambda+8)}{\lambda-1} 2^{j+1}$, their expected final assignment cost (with respect to any configuration in phase j) never exceeds $\frac{4\alpha(\lambda+8)}{\lambda-1} f_c$ plus their optimal assignment cost. □

Outer Demands. Then, we prove that the final assignment cost of every outer demand u is at most $(2\lambda+1)d_u^*$. The proof is similar to the proof of Lemma 4 except for the case where u arrives in phase -3 and $d_u^* < g(F_u, c_u)/\lambda$. In this case, we prove that u opens a new facility with certainty. After the new facility opens, the final assignment cost of u is at most $(2\lambda+1)d_u^*$.

Lemma 8. *For every outer demand u, the final assignment cost of u is at most $(2\lambda+1)d_u^*$.*

Proof. Let j be the phase of c_u in which u arrives as an outer demand. If $j \geq -2$, then $g(F_u, c_u) < 2r_{c_u}(j) \leq 2\lambda d_u^*$ and the final assignment cost of u is $g(F_u, u) \leq (2\lambda+1)d_u^*$.

We have also to consider the case where u arrives in phase -3. If $d_u^* \geq g(F_u, c_u)/\lambda$, the final assignment cost of u is $g(F_u, u) \leq (\lambda+1)d_u^*$.

The most interesting case is when u arrives in phase -3 and $d_u^* < g(F_u, c_u)/\lambda$. Then,

$$\alpha(f_{c_u} + d_u^*) \leq \frac{\alpha(\lambda+8)}{8} d_u^* < \frac{\alpha(\lambda+8)}{8} \frac{1}{\lambda-1} g(F_u, u) \leq g(F_u, u) \tag{7}$$

The first inequality follows from $d_u^* \geq 8f_{c_u}/\lambda$ because u is an outer demand for phase -3. The second inequality follows from $d_u^* < g(F_u, c_u)/\lambda$, which implies that $d_u^* < g(F_u, u)/(\lambda-1)$ (see e.g. (4) in the proof of Proposition 5). The last inequality holds for all $\alpha \leq \frac{8(\lambda-1)}{\lambda+8}$.

Since $d_u^* < g(F_u, c_u)/\lambda$, Proposition 5.b implies that the facility $w \in B_u(\frac{1}{\lambda-1} g(F_u, u))$ that minimizes $f_w + d(w, u)$ has an opening cost of $f_w \leq f_w + d(w, u) \leq f_{c_u} + d_u^*$. Using (7), we obtain that $\alpha f_w \leq g(F_u, u)$. Hence the facility w opens with certainty. After w opens, the final assignment cost of u is at most

$$\begin{aligned}
g(w, u) &= d(w, u) + 2m(w) \\
&\leq d(w, u) + f_w + \tfrac{\lambda-3}{6} d(w, u) && \text{as } m(w) \leq f_w/2 + \tfrac{\lambda-3}{12} d(w, u) \\
&\leq \tfrac{\lambda+3}{6} [f_w + d(w, u)] && \text{for every } \lambda \geq 3 \\
&\leq \tfrac{\lambda+3}{6} [f_{c_u} + d_u^*] && \text{as } f_w + d(w, u) \leq f_{c_u} + d_u^* \\
&\leq \tfrac{\lambda+3}{6} \tfrac{\lambda+8}{8} d_u^* && \text{as } d_u^* \geq 8f_{c_u}/\lambda \\
&\leq (2\lambda+1)d_u^* && \text{for every } \lambda \leq 85
\end{aligned}$$

Consequently, the final assignment cost of an outer demand u (with respect the algorithm's configuration just after u is considered) is no greater than $(2\lambda+1)d_u^*$. □

Combining the upper bounds on the final assignment cost of inner and outer demands, we obtain that the expected assignment cost of FNFL is at most $\frac{4\alpha(\lambda+8)}{\lambda-1} \mathrm{Fac}^* + (2\lambda+1)\mathrm{Asg}^*$.

3.4 Concluding the Proof of Theorem 2

Putting the facility cost and the assignment cost together, we obtain that the expected cost of FNFL is at most

$$\left(\tfrac{4\alpha(\lambda+8)}{\lambda-1}+1\right)\mathrm{Fac}^* + \left(1+\tfrac{1}{\alpha}\right)(2\lambda+1)\,\mathrm{Asg}^*$$

for every $\lambda \in [20, 34]$ and $\alpha \in \left[\frac{4(\lambda-1)}{\lambda+8}, \min\left\{\frac{6(\lambda+1)}{\lambda-3}, \frac{8(\lambda-1)}{\lambda+8}\right\}\right]$. For $\lambda = 20$ and $\alpha = \frac{38}{7}$, the competitive ratio of FNFL is less than 48.6. The competitive ratio can be improved significantly by a careful analysis.

Remark. For a slightly worse competitive ratio, we can use $\lambda = 22$ and $\alpha = \frac{28}{5}$ and set $m(w)$ to $\min\{g(F, u), 3.5 f_w + \frac{\lambda-3}{2}\, d(w, u)\}/7$. This implementation does not compute the final distance of w and is significantly faster in practice than the implementation in Fig. 4.

4 Conclusions

In this work, we present the first one-pass memoryless algorithm for metric Facility Location which maintains a set of facilities approximating the optimal facility configuration within a constant factor. The algorithm considers the demand points one-by-one in arbitrary order, keeps in memory only the facility locations currently open, is randomized and extremely simple to state and implement.

In addition to opening new facilities as in the online setting, e.g. [12, 6, 1], the algorithm can replace existing facilities by new ones. We introduce a simple and intuitive facility replacement rule that works in a memoryless fashion. In particular, a replacement ball around each facility is determined at the moment when the facility opens. Then the facility is replaced by the first new facility in its replacement ball. The idea is that if the replacement radius is set carefully, the replacement rule provides every point p with an upper bound on its distance to the nearest facility in the future. We call this bound the final distance of p. Both the formulation and the analysis of the algorithm are based on the notion of final distance.

Our algorithm maintains an $\mathrm{O}(1)$-competitive facility configuration but it does not maintain the assignment cost of the demands considered so far with respect to it. In Section 1.2, we give some intuition about the difficulties of obtaining a one-pass memoryless algorithm that achieves a constant competitive ratio and maintains both a competitive facility configuration and its assignment cost. An intriguing research direction is to turn that intuition into a formal lower bound.

On the positive side, the best streaming algorithms for k-Median are based on memoryless algorithms for Online Facility Location. In particular, the algorithm of [5] is based on the online algorithm of Meyerson [12], and the algorithm of [8] is closely related to the online algorithm of [1]. An interesting direction for further research is to investigate whether the ideas and techniques presented in this paper can lead to an improved streaming algorithm for k-Median.

References

1. A. Anagnostopoulos, R. Bent, E. Upfal, and P. Van Hentenryck. A Simple and Deterministic Competitive Algorithm for Online Facility Location. *Information and Computation*, 194:175–202, 2004.
2. A. Borodin and R. El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

3. M. Bădoiu, A. Czumaj, P. Indyk, and C. Sohler. Facility Location in Sublinear Time. *ICALP '05*, LNCS 3580, pp. 866–877, 2005.
4. M. Charikar, C. Chekuri, T. Feder, and R. Motwani. Incremental Clustering and Dynamic Information Retrieval. *STOC '97*, pp. 626–635, 1997.
5. M. Charikar, L. O'Callaghan, and R. Panigrahy. Better Streaming Algorithms for Clustering Problems. *STOC '03*, pp. 30–39, 2003.
6. D. Fotakis. On the Competitive Ratio for Online Facility Location. *ICALP '03*, LNCS 2719, pp. 637–652, 2003.
7. D. Fotakis. Incremental Algorithms for Facility Location and k-Median. *ESA '04*, LNCS 3221, pp. 347–321, 2004.
8. G. Frahling and C. Sohler. Coresets in Dynamic Geometric Data Streams. *STOC '05*, pp. 209–217, 2005.
9. S. Guha, A. Meyerson, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering Data Streams: Theory and Practice. *IEEE Transactions on Knowledge and Data Engineering*, 15(3):515–528, 2003.
10. S. Guha, N. Mishra, R. Motwani, and L. O'Callaghan. Clustering Data Streams. *FOCS '00*, pp. 359–366, 2000.
11. P. Indyk. Algorithms for Dynamic Geometric Problems over Data Streams. *STOC '04*, pp. 373–380, 2004.
12. A. Meyerson. Online Facility Location. *FOCS '01*, pp. 426–431, 2001.
13. S. Muthukrishnan. Data Streams: Algorithms and Applications. *SODA '03*, invited talk. Available at `http://athos.rutgers.edu/~muthu/stream-1-1.ps`, 2003.
14. M. Thorup. Quick k-Median, k-Center, and Facility Location for Sparse Graphs. *SIAM J. on Computing*, 34(2):405–432, 2005.