Combinatorial and Computational Geometry
MSRI Publications
Volume **52**, 2005

Geometric Approximation via Coresets

PANKAJ K. AGARWAL, SARIEL HAR-PELED,
AND KASTURI R. VARADARAJAN

ABSTRACT. The paradigm of coresets has recently emerged as a powerful tool for efficiently approximating various extent measures of a point set P. Using this paradigm, one quickly computes a small subset Q of P, called a *coreset*, that approximates the original set P and and then solves the problem on Q using a relatively inefficient algorithm. The solution for Q is then translated to an approximate solution to the original point set P. This paper describes the ways in which this paradigm has been successfully applied to various optimization and extent measure problems.

1. Introduction

One of the classical techniques in developing approximation algorithms is the extraction of "small" amount of "most relevant" information from the given data, and performing the computation on this extracted data. Examples of the use of this technique in a geometric context include random sampling [Chazelle 2000; Mulmuley 1993], convex approximation [Dudley 1974; Bronshteyn and Ivanov 1976], surface simplification [Heckbert and Garland 1997], feature extraction and shape descriptors [Dryden and Mardia 1998; Costa and César 2001]. For geometric problems where the input is a set of points, the question reduces to finding a small subset (a *coreset*) of the points, such that one can perform the desired computation on the coreset.

As a concrete example, consider the problem of computing the diameter of a point set. Here it is clear that, in the worst case, classical sampling techniques like ε-approximation and ε-net would fail to compute a subset of points that contain a good approximation to the diameter [Vapnik and Chervonenkis 1971; Haussler and Welzl 1987]. While in this problem it is clear that convex approximation

Research by the first author is supported by NSF under grants CCR-00-86013, EIA-98-70724, EIA-01-31905, and CCR-02-04118, and by a grant from the U.S.-Israel Binational Science Foundation. Research by the second author is supported by NSF CAREER award CCR-0132901. Research by the third author is supported by NSF CAREER award CCR-0237431.

(i.e., an approximation of the convex hull of the point set) is helpful and provides us with the desired coreset, convex approximation of the point set is not useful for computing the narrowest annulus containing a point set in the plane.

In this paper, we describe several recent results which employ the idea of coresets to develop efficient approximation algorithms for various geometric problems. In particular, motivated by a variety of applications, considerable work has been done on measuring various descriptors of the extent of a set P of n points in $\mathbb{R}^d$. We refer to such measures as *extent measures* of P. Roughly speaking, an extent measure of P either computes certain statistics of P itself or of a (possibly nonconvex) geometric shape (e.g. sphere, box, cylinder, etc.) enclosing P. Examples of the former include computing the k-th largest distance between pairs of points in P, and the examples of the latter include computing the smallest radius of a sphere (or cylinder), the minimum volume (or surface area) of a box, and the smallest width of a slab (or a spherical or cylindrical shell) that contain P. There has also been some recent work on maintaining extent measures of a set of moving points [Agarwal et al. 2001b].

Shape fitting, a fundamental problem in computational geometry, computer vision, machine learning, data mining, and many other areas, is closely related to computing extent measures. The shape fitting problem asks for finding a shape that best fits P under some "fitting" criterion. A typical criterion for measuring how well a shape γ fits P, denoted as $\mu(P, \gamma)$, is the maximum distance between a point of P and its nearest point on γ, i.e., $\mu(P, \gamma) = \max_{p \in P} \min_{q \in \gamma} \|p - q\|$. Then one can define the extent measure of P to be $\mu(P) = \min_{\gamma} \mu(P, \gamma)$, where the minimum is taken over a family of shapes (such as points, lines, hyperplanes, spheres, etc.). For example, the problem of finding the minimum radius sphere (resp. cylinder) enclosing P is the same as finding the point (resp. line) that fits P best, and the problem of finding the smallest width slab (resp. spherical shell, cylindrical shell)[1] is the same as finding the hyperplane (resp. sphere, cylinder) that fits P best.

The exact algorithms for computing extent measures are generally expensive, e.g., the best known algorithms for computing the smallest volume bounding box containing P in $\mathbb{R}^3$ run in $O(n^3)$ time. Consequently, attention has shifted to developing approximation algorithms [Barequet and Har-Peled 2001]. The goal is to compute an $(1+\varepsilon)$-approximation, for some $0 < \varepsilon < 1$, of the extent measure in roughly $O(nf(\varepsilon))$ or even $O(n+f(\varepsilon))$ time, that is, in time near-linear or linear in n. The framework of coresets has recently emerged as a general approach to achieve this goal. For any extent measure μ and an input point set P for which we wish to compute the extent measure, the general idea is to argue that there exists an easily computable subset $Q \subseteq P$, called a *coreset*, of size $1/\varepsilon^{O(1)}$, so

[1] A *slab* is a region lying between two parallel hyperplanes; a *spherical shell* is the region lying between two concentric spheres; a *cylindrical shell* is the region lying between two coaxial cylinders.

that solving the underlying problem on Q gives an approximate solution to the original problem. For example, if $\mu(Q) \geq (1-\varepsilon)\mu(P)$, then this approach gives an approximation to the extent measure of P. In the context of shape fitting, an appropriate property for Q is that for any shape γ from the underlying family, $\mu(Q,\gamma) \geq (1-\varepsilon)\mu(P,\gamma)$. With this property, the approach returns a shape γ^* that is an approximate best fit to P.

Following earlier work [Barequet and Har-Peled 2001; Chan 2002; Zhou and Suri 2002] that hinted at the generality of this approach, [Agarwal et al. 2004] provided a formal framework by introducing the notion of ε-kernel and showing that it yields a coreset for many optimization problems. They also showed that this technique yields approximation algorithms for a wide range of problems. Since the appearance of preliminary versions of their work, many subsequent papers have used a coreset based approach for other geometric optimization problems, including clustering and other extent-measure problems [Agarwal et al. 2002; Bădoiu and Clarkson 2003b; Bădoiu et al. 2002; Har-Peled and Wang 2004; Kumar et al. 2003; Kumar and Yildirim $\geq$ 2005].

In this paper, we have attempted to review coreset based algorithms for approximating extent measure and other optimization problems. Our aim is to communicate the flavor of the techniques involved and a sense of the power of this paradigm by discussing a number of its applications. We begin in Section 2 by describing ε-kernels of point sets and algorithms for constructing them. Section 3 defines the notion of ε-kernel for functions and describes a few of its applications. We then describe in Section 4 a simple incremental algorithm for shape fitting. Section 5 discusses the computation of ε-kernels in the streaming model. Although ε-kernels provide coresets for a variety of extent measures, they do not give coresets for many other problems, including clustering. Section 6 surveys the known results on coresets for clustering. The size of the coresets discussed in these sections increases exponentially with the dimension, so we conclude in Section 7 by discussing coresets for points in very high dimensions whose size depends polynomially on the dimension, or is independent of the dimension altogether.

2. Kernels for Point Sets

Let μ be a measure function (e.g., the width of a point set) from subsets of $\mathbb{R}^d$ to the nonnegative reals $\mathbb{R}^+ \cup \{0\}$ that is monotone, i.e., for $P_1 \subseteq P_2$, $\mu(P_1) \leq \mu(P_2)$. Given a parameter $\varepsilon > 0$, we call a subset $Q \subseteq P$ an *ε-coreset* of P (with respect to μ) if

$$(1-\varepsilon)\mu(P) \leq \mu(Q).$$

Agarwal et al. [2004] introduced the notion of *ε-kernels* and showed that it is an $f(\varepsilon)$-coreset for numerous minimization problems. We begin by defining ε-kernels and related concepts.

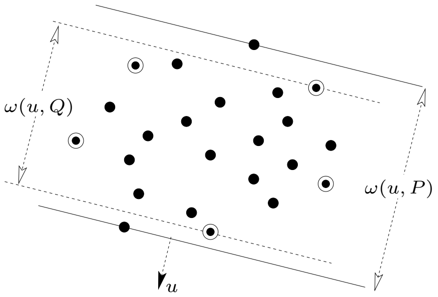

Figure 1. Directional width and ε-kernel.

ε-kernel. Let $\mathbb{S}^{d-1}$ denote the unit sphere centered at the origin in $\mathbb{R}^d$. For any set P of points in $\mathbb{R}^d$ and any direction $u \in \mathbb{S}^{d-1}$, we define the *directional width* of P in direction u, denoted by $\omega(u, P)$, to be

$$\omega(u, P) = \max_{p \in P} \langle u, p \rangle - \min_{p \in P} \langle u, p \rangle ,$$

where $\langle \cdot, \cdot \rangle$ is the standard inner product. Let $\varepsilon > 0$ be a parameter. A subset $Q \subseteq P$ is called an *ε-kernel* of P if for each $u \in \mathbb{S}^{d-1}$,

$$(1 - \varepsilon)\omega(u, P) \leq \omega(u, Q).$$

Clearly, $\omega(u, Q) \leq \omega(u, P)$. Agarwal et al. [2004] call a measure function μ *faithful* if there exists a constant c, depending on μ, so that for any $P \subseteq \mathbb{R}^d$ and for any ε, an ε-kernel of P is a $c\varepsilon$-coreset for P with respect to μ. Examples of faithful measures considered in that reference include diameter, width, radius of the smallest enclosing ball, and volume of the smallest enclosing box. A common property of these measures is that $\mu(P) = \mu(\mathrm{conv}(P))$. We can thus compute an ε-coreset of P with respect to several measures by simply computing an (ε/c)-kernel of P.

Algorithms for computing kernels. An ε-kernel of P is a subset whose convex hull approximates, in a certain sense, the convex hull of P. Other notions of convex hull approximation have been studied and methods have been developed to compute them; see [Bentley et al. 1982; Bronshteyn and Ivanov 1976; Dudley 1974] for a sample. For example, in the first of these articles Bentley, Faust, and Preparata show that for any point set $P \subseteq \mathbb{R}^2$ and $\varepsilon > 0$, a subset Q of P whose size is $O(1/\varepsilon)$ can be computed in $O(|P| + 1/\varepsilon)$ time such that for any $p \in P$, the distance of p to $\mathrm{conv}(Q)$ is at most $\varepsilon \mathrm{diam}(Q)$. Note however that such a guarantee is not enough if we want Q to be a coreset of P with respect to faithful measures. For instance, the width of Q could be arbitrarily small compared to the width of P. The width of an ε-kernel of P, on the other hand, is easily seen to be a good approximation to the width of P. To the best of our knowledge, the first efficient method for computing a small ε-kernel of an arbitrary point set is implicit in [Barequet and Har-Peled 2001].

We call P *α-fat*, for $\alpha \le 1$, if there exists a point $p \in \mathbb{R}^d$ and a hypercube $\overline{\mathbb{C}}$ centered at the origin so that

$$p + \alpha\overline{\mathbb{C}} \subset \operatorname{conv}(P) \subset p + \overline{\mathbb{C}}.$$

A stronger version of the following lemma, which is very useful for constructing an ε-kernel, was proved in [Agarwal et al. 2004] by adapting a scheme from [Barequet and Har-Peled 2001]. Their scheme can be thought of as one that quickly computes an approximation to the Löwner–John Ellipsoid [John 1948].

LEMMA 2.1. *Let P be a set of n points in $\mathbb{R}^d$ such that the volume of* $\operatorname{conv}(P)$ *is nonzero, and let $\mathbb{C} = [-1, 1]^d$. One can compute in $O(n)$ time an affine transform τ so that $\tau(P)$ is an α-fat point set satisfying $\alpha\mathbb{C} \subset \operatorname{conv}(\tau(P)) \subset \mathbb{C}$, where α is a positive constant depending on d, and so that a subset $Q \subseteq P$ is an ε-kernel of P if and only if $\tau(Q)$ is an ε-kernel of $\tau(P)$.*

The importance of Lemma 2.1 is that it allows us to adapt some classical approaches for convex hull approximation [Bentley et al. 1982; Bronshteyn and Ivanov 1976; Dudley 1974] which in fact do compute an ε-kernel when applied to fat point sets.

We now describe algorithms for computing ε-kernels. By Lemma 2.1, we can assume that $P \subseteq [-1, +1]^d$ is α-fat. We begin with a very simple algorithm.

Let δ be the largest value such that $\delta \le (\varepsilon/\sqrt{d})\alpha$ and $1/\delta$ is an integer. We consider the d-dimensional grid $\mathbb{Z}$ of size δ. That is,

$$\mathbb{Z} = \{(\delta i_1, \dots, \delta i_d) \mid i_1, \dots, i_d \in \mathbb{Z}\}.$$

For each column along the x_d-axis in $\mathbb{Z}$, we choose one point from the highest nonempty cell of the column and one point from the lowest nonempty cell of the column; see Figure 2, top left. Let Q be the set of chosen points. Since $P \subseteq [-1, +1]^d$, $|Q| = O(1/(\alpha\varepsilon)^{d-1})$. Moreover Q can be constructed in time $O(n + 1/(\alpha\varepsilon)^{d-1})$ provided that the ceiling operation can be performed in constant time. Agarwal et al. [2004] showed that Q is an ε-kernel of P. Hence, we can compute an ε-kernel of P of size $O(1/\varepsilon^{d-1})$ in time $O(n+1/\varepsilon^{d-1})$. This approach resembles the algorithm of [Bentley et al. 1982].

Next we describe an improved construction, observed independently in [Chan 2004] and [Yu et al. 2004], which is a simplification of an algorithm of [Agarwal et al. 2004], which in turn is an adaptation of a method of Dudley [1974]. Let $\mathcal{S}$ be the sphere of radius $\sqrt{d} + 1$ centered at the origin. Set $\delta = \sqrt{\varepsilon\alpha} \le 1/2$. One can construct a set $\mathcal{I}$ of $O(1/\delta^{d-1}) = O(1/\varepsilon^{(d-1)/2})$ points on the sphere $\mathcal{S}$ so that for any point x on $\mathcal{S}$, there exists a point $y \in \mathcal{I}$ such that $\|x - y\| \le \delta$. We process P into a data structure that can answer ε-approximate nearest-neighbor queries [Arya et al. 1998]. For a query point q, let $\varphi(q)$ be the point of P returned by this data structure. For each point $y \in \mathcal{I}$, we compute $\varphi(y)$ using this data structure. We return the set $Q = \{\varphi(y) \mid y \in \mathcal{I}\}$; see Figure 2, top right.

We now briefly sketch, following the argument in [Yu et al. 2004], why Q is is an ε-kernel of P. For simplicity, we prove the claim under the assumption that $\varphi(y)$ is the *exact* nearest-neighbor of y in P. Fix a direction $u \in \mathbb{S}^{d-1}$. Let $\sigma \in P$ be the point that maximizes $\langle u, p\rangle$ over all $p \in P$. Suppose the ray emanating from σ in direction u hits $\mathcal{S}$ at a point x. We know that there exists a point $y \in \mathcal{I}$ such that $\|x - y\| \le \delta$. If $\varphi(y) = \sigma$, then $\sigma \in Q$ and

$$\max_{p \in P}\langle u, p\rangle - \max_{q \in Q}\langle u, q\rangle = 0.$$

Now suppose $\varphi(y) \neq \sigma$. Let B be the d-dimensional ball of radius $||y - \sigma||$ centered at y. Since $\|y - \varphi(y)\| \le \|y - \sigma\|$, $\varphi(y) \in B$. Let us denote by z the point on the sphere ∂B that is hit by the ray emanating from y in direction $-u$. Let w be the point on zy such that $zy \perp \sigma w$ and h the point on σx such that $yh \perp \sigma x$; see Figure 2, bottom.

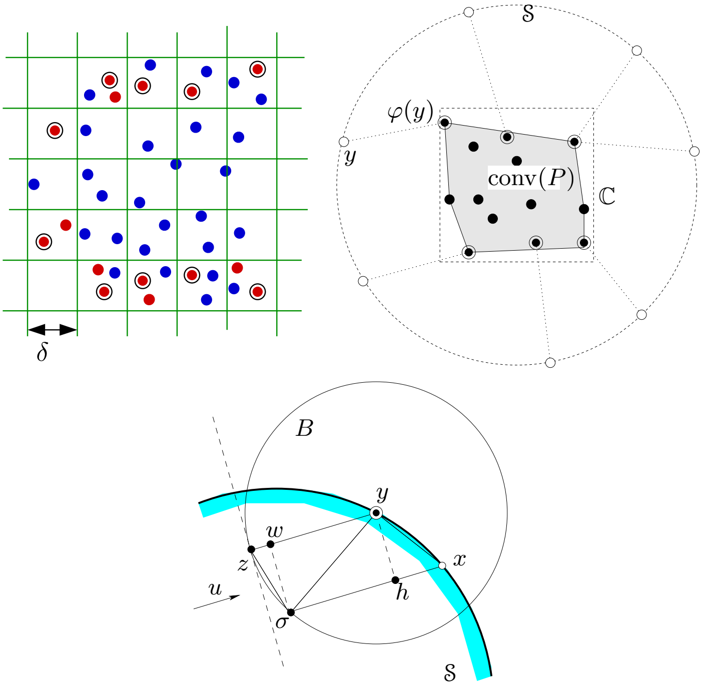

Figure 2. Top left: A grid based algorithm for constructing an ε-kernel. Top right: An improved algorithm. Bottom: Correctness of the improved algorithm.

The hyperplane normal to u and passing through z is tangent to B. Since $\varphi(y)$ lies inside B, $\langle u, \varphi(y)\rangle \ge \langle u, z\rangle$. Moreover, it can be shown that $\langle u, \sigma\rangle - \langle u, \varphi(y)\rangle \le \alpha\varepsilon$. Thus, we can write

$$\max_{p \in P}\langle u, p\rangle - \max_{q \in Q}\langle u, q\rangle \le \langle u, \sigma\rangle - \langle u, \varphi(y)\rangle \le \alpha\varepsilon.$$

Similarly, we have $\min_{p \in P} \langle u, p \rangle - \min_{q \in Q} \langle u, q \rangle \geq -\alpha\varepsilon$.

The above two inequalities together imply that $\omega(u,Q) \geq \omega(u,P) - 2\alpha\varepsilon$. Since $\alpha\mathbb{C} \subset \operatorname{conv}(P)$, $\omega(u,P) \geq 2\alpha$. Hence $\omega(u,Q) \geq (1-\varepsilon)\omega(u,P)$, for any $u \in \mathbb{S}^{d-1}$, thereby implying that Q is an ε-kernel of P.

A straightforward implementation of the above algorithm, i.e., the one that answers a nearest-neighbor query by comparing the distances to all the points, runs in $O(n/\varepsilon^{(d-1)/2})$ time. However, we can first compute an $(\varepsilon/2)$-kernel Q' of P of size $O(1/\varepsilon^{d-1})$ using the simple algorithm and then compute an $(\varepsilon/4)$-kernel using the improved algorithm. Chan [2004] introduced the notion of discrete Voronoi diagrams, which can be used for computing the nearest neighbors of a set of grid points among the sites that are also a subset of a grid. Using this structure Chan showed that $\varphi(y)$, for all $y \in \mathfrak{I}$, can be computed in a total time of $O(n + 1/\varepsilon^{d-1})$ time. Putting everything together, one obtains an algorithm that runs in $O(n + 1/\varepsilon^{d-1})$ time. Chan in fact gives a slightly improved result:

THEOREM 2.2 [Chan 2004]. *Given a set P of n points in $\mathbb{R}^d$ and a parameter $\varepsilon > 0$, one can compute an ε-kernel of P of size $O(1/\varepsilon^{(d-1)/2})$ in time $O(n + 1/\varepsilon^{d-(3/2)})$.*

Experimental results. Yu et al. [2004] implemented their ε-kernel algorithm and tested its performance on a variety of inputs. They measure the quality of an ε-kernel Q of P as the maximum relative error in the directional width of P and Q. Since it is hard to compute the maximum error over all directions, they sampled a set Δ of 1000 directions in $\mathbb{S}^{d-1}$ and computed the maximum relative error with respect to these directions, i.e.,

$$\operatorname{err}(Q,P) = \max_{u \in \Delta} \frac{\omega(u,P) - \omega(u,Q)}{\omega(u,P)}. \tag{2–1}$$

They implemented the constant-factor approximation algorithm of [Barequet and Har-Peled 2001] for computing the minimum-volume bounding box to convert P into an α-fat set, and they used the ANN library [Arya and Mount 1998] for answering approximate nearest-neighbor queries. Table 1 shows the running time of their algorithm for a variety of synthetic inputs: (i) points uniformly distributed on a sphere, (ii) points distributed on a cylinder, and (iii) clustered point sets, consisting of 20 equal sized clusters. The running time is decomposed into two components: (i) preprocessing time that includes the time spent in converting P into a fat set and in preprocessing P for approximate nearest-neighbor queries, and (ii) query time that includes the time spent in computing $\varphi(x)$ for $x \in \mathfrak{I}$. Figure 3 shows how the error $\operatorname{err}(Q,P)$ changes as the function of kernel. These experiments show that their algorithm works extremely well in low dimensions (≤ 4) both in terms of size and running time. See [Yu et al. 2004] for more detailed experiments.

Input Type	*Input Size*	$d=2$		$d=4$		$d=6$		$d=8$	
		Pre	*Que*	*Pre*	*Que*	*Pre*	*Que*	*Pre*	*Que*
sphere	10^4	0.03	0.01	0.06	0.05	0.10	9.40	0.15	52.80
	10^5	0.54	0.01	0.90	0.50	1.38	67.22	1.97	1393.88
	10^6	9.25	0.01	13.08	1.35	19.26	227.20	26.77	5944.89
cylinder	10^4	0.03	0.01	0.06	0.03	0.10	2.46	0.16	17.29
	10^5	0.60	0.01	0.91	0.34	1.39	30.03	1.94	1383.27
	10^6	9.93	0.01	13.09	0.31	18.94	87.29	26.12	5221.13
clustered	10^4	0.03	0.01	0.06	0.01	0.10	0.08	0.15	2.99
	10^5	0.31	0.01	0.63	0.02	1.07	1.34	1.64	18.39
	10^6	5.41	0.01	8.76	0.02	14.75	1.08	22.51	54.12

Table 1. Running time for computing ε-kernels of various synthetic data sets, $\varepsilon < 0.05$. *Prepr* denotes the preprocessing time, including converting P into a fat set and building ANN data structures. *Query* denotes the time for performing approximate nearest-neighbor queries. Running time is measured in seconds. The experiments were conducted on a Dell PowerEdge 650 server with a 3.06GHz Pentium IV processor and 3GB memory, running Linux 2.4.20.

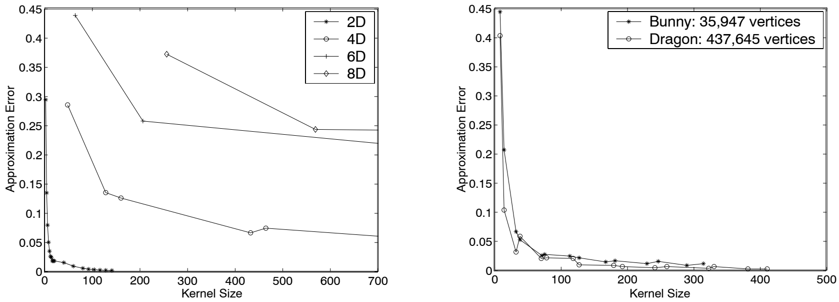

Figure 3. Approximation errors under different sizes of computed ε-kernels. Left: *sphere*. Right: various geometric models. All synthetic inputs had 100,000 points.

Applications. Theorem 2.2 can be used to compute coresets for faithful measures, defined in Section 2. In particular, if we have a faithful measure μ that can be computed in $O(n^\alpha)$ time, then by Theorem 2.2, we can compute a value $\overline{\mu}$, $(1-\varepsilon)\mu(P) \le \overline{\mu} \le \mu(P)$ by first computing an (ε/c)-kernel Q of P and then using an exact algorithm for computing $\mu(Q)$. The total running time of the algorithm is $O(n + 1/\varepsilon^{d-(3/2)} + 1/\varepsilon^{\alpha(d-1)/2})$. For example, a $(1+\varepsilon)$-approximation of the diameter of a point set can be computed in time $O(n + 1/\varepsilon^{d-1})$ since the exact diameter can be computed in quadratic time. By being a little more careful, the running time of the diameter algorithm can be improved to $O(n + 1/\varepsilon^{d-(3/2)})$ [Chan 2004]. Table 2 gives running times for computing an $(1+\varepsilon)$-approximation of a few faithful measures.

We note that ε-kernels in fact guarantee a stronger property for several faithful measures. For instance, if Q is an ε-kernel of P, and C is some cylinder containing

Extent	Time complexity
Diameter	$n + 1/\varepsilon^{d-(3/2)}$
Width	$(n + 1/\varepsilon^{d-2}) \log(1/\varepsilon)$
Minimum enclosing cylinder	$n + 1/\varepsilon^{d-1}$
Minimum enclosing box(3D)	$n + 1/\varepsilon^{3}$

Table 2. Time complexity of computing $(1 + \varepsilon)$-approximations for certain faithful measures.

Q, then a “concentric” scaling of C by a factor of $(1 + c\varepsilon)$, for some constant c, contains P. Thus we can compute not only an approximation to the minimum radius r^* of a cylinder containing P, but also a cylinder of radius at most $(1+\varepsilon)r^*$ that contains P.

The approach described in this section for approximating faithful measures had been used for geometric approximation algorithms before the framework of ε-kernels was introduced; see [Agarwal and Procopiuc 2002; Barequet and Har-Peled 2001; Chan 2002; Zhou and Suri 2002], for example. The framework of ε-kernels, however, provides a unified approach and turns out to be crucial for the approach developed in the next section for approximating measures that are not faithful.

3. Kernels for Sets of Functions

The crucial notion used to derive coresets and efficient approximation algorithms for measures that are not faithful is that of a kernel of a set of functions.

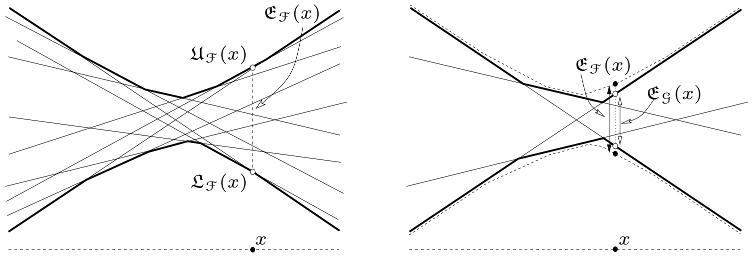
$\mathfrak{E}_{\mathcal{F}}(x)$
$\mathfrak{U}_{\mathcal{F}}(x)$
$\mathfrak{L}_{\mathcal{F}}(x)$
x
$\mathfrak{E}_{\mathcal{F}}(x)$
$\mathfrak{E}_{\mathcal{G}}(x)$
x

Figure 4. Envelopes, extent, and ε-kernel.

Envelopes and extent. Let $\mathcal{F} = \{f_1, \dots, f_n\}$ be a set of n d-variate real-valued functions defined over $x = (x_1, \dots, x_{d-1}, x_d) \in \mathbb{R}^d$. The *lower envelope* of $\mathcal{F}$ is the graph of the function $\mathfrak{L}_{\mathcal{F}} : \mathbb{R}^d \to \mathbb{R}$ defined as $\mathfrak{L}_{\mathcal{F}}(x) = \min_{f \in \mathcal{F}} f(x)$. Similarly, the *upper envelope* of $\mathcal{F}$ is the graph of the function $\mathfrak{U}_{\mathcal{F}} : \mathbb{R}^d \to \mathbb{R}$ defined as $\mathfrak{U}_{\mathcal{F}}(x) = \max_{f \in \mathcal{F}} f(x)$. The *extent* $\mathfrak{E}_{\mathcal{F}} : \mathbb{R}^d \to \mathbb{R}$ of $\mathcal{F}$ is defined as

$$\mathfrak{E}_{\mathcal{F}}(x) = \mathfrak{U}_{\mathcal{F}}(x) - \mathfrak{L}_{\mathcal{F}}(x).$$

Let $\varepsilon > 0$ be a parameter. We say that a subset $\mathcal{G} \subseteq \mathcal{F}$ is an *ε-kernel* of $\mathcal{F}$ if

$$(1-\varepsilon)\mathfrak{E}_{\mathcal{F}}(x) \le \mathfrak{E}_{\mathcal{G}}(x) \qquad \forall x \in \mathbb{R}^d.$$

Obviously, $\mathfrak{E}_{\mathcal{G}}(x) \le \mathfrak{E}_{\mathcal{F}}(x)$, as $\mathcal{G} \subseteq \mathcal{F}$.

Let $\mathcal{H} = \{h_1, \ldots, h_n\}$ be a family of d-variate linear functions and $\varepsilon > 0$ a parameter. We define a *duality* transformation that maps the d-variate function (or a hyperplane in $\mathbb{R}^{d+1}$) $h : x_{d+1} = a_1x_1 + a_2x_2 + \cdots + a_dx_d + a_{d+1}$ to the point $h^\star = (a_1, a_2, \ldots, a_d, a_{d+1})$ in $\mathbb{R}^{d+1}$. Let $\mathcal{H}^\star = \{h^\star \mid h \in \mathcal{H}\}$. It can be proved [Agarwal et al. 2004] that $\mathcal{K} \subseteq \mathcal{H}$ is an ε-kernel of $\mathcal{H}$ if and only if $\mathcal{K}^*$ is an ε-kernel of $\mathcal{H}^*$. Hence, by computing an ε-kernel of $\mathcal{H}^*$ we can also compute an ε-kernel of $\mathcal{H}$. The following is therefore a corollary of Theorem 2.2.

COROLLARY 3.1 [Agarwal et al. 2004; Chan 2004]. *Given a set $\mathcal{F}$ of n d-variate linear functions and a parameter $\varepsilon > 0$, one can compute an ε-kernel of $\mathcal{F}$ of size $O(1/\varepsilon^{d/2})$ in time $O(n + 1/\varepsilon^{d-(1/2)})$.*

We can compute ε-kernels of a set of polynomial functions by using the notion of linearization.

Linearization. Let $f(x,a)$ be a $(d+p)$-variate polynomial, $x \in \mathbb{R}^d$ and $a \in \mathbb{R}^p$. Let $a^1, \ldots, a^n \in \mathbb{R}^p$, and set $\mathcal{F} = \left\{ f_i(x) \equiv f(x, a^i) \mid 1 \le i \le n \right\}$. Suppose we can express $f(x,a)$ in the form

$$f(x,a) = \psi_0(a) + \psi_1(a)\varphi_1(x) + \cdots + \psi_k(a)\varphi_k(x), \tag{3–1}$$

where $\psi_0, \ldots, \psi_k$ are p-variate polynomials and $\varphi_1, \ldots, \varphi_k$ are d-variate polynomials. We define the map $\varphi : \mathbb{R}^d \to \mathbb{R}^k$

$$\varphi(x) = (\varphi_1(x), \ldots, \varphi_k(x)).$$

Then the image $\Gamma = \left\{ \varphi(x) \mid x \in \mathbb{R}^d \right\}$ of $\mathbb{R}^d$ is a d-dimensional surface in $\mathbb{R}^k$ (if $k \ge d$), and for any $a \in \mathbb{R}^p$, $f(x,a)$ maps to a k-variate linear function

$$h_a(y_1, \ldots, y_k) = \psi_0(a) + \psi_1(a)y_1 + \cdots + \psi_k(a)y_k$$

in the sense that for any $x \in \mathbb{R}^d$, $f(x,a) = h_a(\varphi(x))$. We refer to k as the *dimension* of the *linearization* φ, and say that $\mathcal{F}$ admits a linearization of dimension k. The most popular example of linearization is perhaps the so-called lifting transform that maps $\mathbb{R}^d$ to a unit paraboloid in $\mathbb{R}^{d+1}$. For example, let $f(x_1, x_2, a_1, a_2, a_3)$ be the function whose absolute value is some measure of the "distance" between a point $(x_1, x_2) \in \mathbb{R}^2$ and a circle with center (a_1, a_2) and radius a_3, which is the 5-variate polynomial

$$f(x_1, x_2, a_1, a_2, a_3) = a_3^2 - (x_1 - a_1)^2 - (x_2 - a_2)^2 .$$

We can rewrite f in the form

$$f(x_1, x_2, a_1, a_2, a_3) = [a_3^2 - a_1^2 - a_2^2] + [2a_1x_1] + [2a_2x_2] - [x_1^2 + x_2^2] , \tag{3–2}$$