APPROXIMATION SCHEMES FOR COVERING AND PACKING PROBLEMS IN ROBOTICS AND VLSI

Extended Abstract

Dorit S. Hochbaum and Wolfgang Maass
350 Barrows Hall Department of Computer Science
University of California, Berkeley
Berkeley, California 94720

1. Introduction

Polynomial approximation schemes for strongly NP-complete problems have rarely been reported in the literature. We describe in this paper a powerful technique and illustrate its use in deriving polynomial approximation schemes for a variety of strongly NP-complete geometric problems. The problems that we consider are defined in a Euclidean space given a specific geometric object. The input is a set of points distributed in some region in the space and the problem is to find the minimum number of objects needed to cover all points. Another type of problem is the packing (with no overlap) of maximum number of objects in a region of space specified by an input of points.

Such problems often occur in practical applications. We illustrate the technique for several applications that were discussed in the literature. For none of these problems has there been an approximation algorithm with guaranteed worst case bound previously reported (with the exception of the scheduling problem in [1] for which 100% error bound is guaranteed).

One of the problems is the square packing problem. This problem comes up in the attempt to increase yield in VLSI chip manufacture. For example, 64K RAM chips, some of which may be defective, are available on a rectilinear grid placed on a silicon wafer. 2x2 arrays of such nondefective chips could be wired together to produce 256K RAM chips. In order to maximize yield, we want to pack a maximal number of such 2x2 arrays into the array of working chips on a wafer. (See Berman, Leighton and Snyder's result [2] reviewed by Johnson [7] and NP-completeness result due to Fowler, Paterson and Tanimoto [3].)

Another problem is covering with disks. Given points in the plane, the problem is to identify a minimally sized set of disks (of prescribed radius), covering all points. One of the applications of the problem is in the area of locating emergency facilities such that all potential customers will be within a reasonably small radius away from the facility. (The complexity results for this problem are reviewed in [7].)

(*)This research was supported in part by the National Science Foundation under grant ECS-8204695.

(**)Address after Fall 1984: Department of Mathematics, Statistics and Computer Science, University of Illinois at Chicago, Chicago, Illinois 60680.

A third problem considered is covering with squares (or rectangles). This problem has an important application to image processing discussed in Tanimoto and Fowler [9]. Frequently most of the areas of images do not include any interesting information (these are the "background" pixels). The problem is then to cover all nontrivial information in minimum number of square "patches" (the position and size of which are constrained by a specified grid) which are then stored in a database.

The problem of covering with rings (referred to as the ring cover problem in this paper) is a generic example of covering with nonconvex objects. The covering objects are the regions bounded between two concentric spheres which we call rings. This problem has an interesting application in the area of planning motion and positions of robots. Many robot constructions are such that any object that is placed between the minimum and maximum range of the robot arm can be reached by the robot. The problem is to identify a minimum number of robot positions such that all points (objects to be handled) are accessible. When the minimum range of the robot arm is zero, the reachability region forms a disk (or a ball), in which case the problem falls in the previously discussed category of covering with disks or balls.

Nonconvex covering problems also arise in the context of scheduling theory. Realistic models take into account resources that are only intermittently available. This may be due to, say, lunch breaks for workers or preventive maintenance for machines (see Bartholdi [1]).

We shall call an algorithm a δ-approximation, $\delta > 0$, for a certain problem if the error of the value of the solution delivered by the algorithm divided by the value of the optimal solution does not exceed δ. Obviously, we would like to identify a δ-approximation algorithm such that δ is as small as possible. In some cases one can specify a family of algorithms such that for each $\varepsilon > 0$ there is an ε-approximation algorithm in the family that solves a given problem instance within relative error ε. Such a family is called an approximation scheme. The running time of an ε-approximation algorithm will be monotonically increasing with $\frac{1}{\varepsilon}$. If the functional dependence of the running time on the size of the input and $\frac{1}{\varepsilon}$ is polynomial, then the scheme is said to be fully polynomial; if, on the other hand, it is polynomial only in the input size, the scheme is called polynomial.

All the problems described and consequently their extensions are NP-complete in the strong sense (the reader is referred to Garey and Johnson's [4] comprehensive review of this concept). As such, there are no fully polynomial approximation schemes for these problems, unless NP=P ([4]). This negative result does not exclude, however, the existence of a polynomial approximation scheme for these problems, i.e., a family of algorithms such that for any specified relative error $\varepsilon > 0$, there is an ε-approximation algorithm in the scheme that is polynomial. Though such schemes are conceptually feasible, their existence has been rarely reported.

Our main results are the construction of polynomial approximation schemes for the above problems which, given the negative result above, are the best possible results of this type. We present a unified methodology that is helpful for numerous geometric covering and packing problems, and could potentially be applicable to problems beyond this context. We call this fundamental technique the "shifting strategy" and outline the necessary conditions for its applicability.

Throughout the paper the following notation will be used. Z^A denotes the value of the solution delivered by algorithm A. An optimal solution set is denoted by OPT and its size by $|OPT|$. We define a d-dimensional ring to be the volume enclosed between two concentric d-dimensional spheres and we say that such a ring is of size $\langle r,w \rangle$ if the difference between the outer and inner radius of the ring is w and the inner radius is equal to r. The diameter of such ring is denoted by D, where $D = 2r + 2w$.

In this paper we omitted many of the details and proofs. For a complete description and extensions see [5], [6] and [8].

2. The "Shifting Strategy"

We illustrate the use of the shifting strategy via the problem of covering with disks. The shifting strategy allows us to bound the error of the simple divide-and-conquer approach by repetitive applications of it, followed by the selection of the single most favorable resulting solution.

Let the set N of the n given points in the plane be enclosed in an area I. The goal is to cover these points with a minimal number of disks of diameter D. Let the shifting parameter be ℓ. In the first phase the area I is subdivided into vertical strips of width D. Each such strip will be considered left closed and right open. These strips are then considered in groups of ℓ consecutive strips resulting in strips of width $\ell \cdot D$ each. For any fixed subdivision of I into strips of width D, there are ℓ different ways of partitioning I into strips of width $\ell \cdot D$. These partitions can be ordered such that each can be derived from the previous one by shifting it to the right over distance D. Repeating the shift ℓ times we end up with the same partition we started from. We denote such ℓ distinguished shift partitions by $S_1, S_2, \ldots, S_\ell$.

Let A be any algorithm that delivers a solution in any strip of width $\ell \cdot D$ (or less). For a given partition S_i, let $A(S_i)$ be the algorithm that applies algorithm A to each strip in the partition S_i, and outputs the union of all disks used. Such set of disks is clearly a feasible solution to the global problem defined on I. This process of finding a global solution is repeated for each partition $S_i, i=1,2,\ldots,\ell$. The shift algorithm S_A, defined for a given local algorithm A, delivers the set of disks of minimum cardinality among the ℓ sets delivered by $A(S_1),\ldots,A(S_\ell)$.

Let the performance ratio of an algorithm B be denoted by r_B, i.e., r_B is defined as the supremum of $Z^B/|OPT|$ over all problem instances.

Lemma 2.1 (the shifting lemma)

Let A be a local algorithm and ℓ be the shifting parameter then

$$r_{S_A} \leq r_A\left(1 + \frac{1}{\ell}\right) . \tag{2.1}$$

Proof: We produce an upper bound on the sum of errors caused by all algorithms $A(S_i)$.

By the definition of r_A we have

$$Z^{A(S_i)} \le r_A \cdot \sum_{J \text{ in } S_i} |OPT_J|, \tag{2.2}$$

where J runs over all strips in partition S_i and $|OPT_J|$ is the number of disks in an optimal cover of the points in strip J.

Let OPT be the set of rings in an optimal solution and $OPT^{(1)},\ldots,OPT^{(\ell)}$ the set of disks, in OPT, covering points in two adjacent $\ell \cdot D$ strips in the $1,2,\ldots,\ell$ shifts, respectively. It can easily be seen that

$$\sum_{J \text{ in } S_i} |OPT_J| \le |OPT| + |OPT^{(i)}|. \tag{2.3}$$

There can be no disk in the set OPT that covers points in two adjacent strips in more than one shift partition. Therefore, the sets $OPT^{(1)},\ldots,OPT^{(\ell)}$ are disjoint. It follows that

$$\sum_{i=1}^{\ell} (|OPT| + |OPT^{(i)}|) \le (\ell+1) \cdot |OPT|. \tag{2.4}$$

(2.3) and (2.4) imply

$$\min_{i=1,..,\ell} \sum_{J \text{ in } S_i} |OPT_J| \le \frac{1}{\ell} \sum_{i=1}^{\ell} \left(\sum_{J \text{ in } S_i} |OPT_J| \right) \le \left(1 + \frac{1}{\ell}\right) |OPT|. \tag{2.5}$$

Joining the inequality (2.5) with (2.2) we derive that

$$Z^{S_A} = \min_{i=1,..,\ell} Z^{A(S_i)} \le r_A \cdot \left(1 + \frac{1}{\ell}\right) \cdot |OPT|, \tag{2.6}$$

which establishes (2.1).

Q.E.D.

The local algorithm A may itself be derived from an application of the shifting strategy in lower dimensional space. Repetitive applications of this type yield an approximation scheme as described in the following section.

3. Polynomial Approximation Schemes for Covering Problems in Arbitrary Dimensions

We first illustrate in Theorem 3.1 the method of repetitive applications of the shifting for the problem of covering with balls in a d-dimensional space. The remainder of this section consists of a generalization of this concept for other objects.

Theorem 3.1 Let $d \ge 1$ be some finite dimension, then there is a polynomial time approximation scheme H^d s.t. for every given natural number $\ell \ge 1$, the algorithm H^d_ℓ delivers a cover of n given points in a d-dimensional Euclidean space by d-dimensional balls of given diameter D in $O((\ell \cdot \sqrt{d})^d \cdot (2n)^{d\lceil \ell\sqrt{d}\rceil^d+1})$ steps with performance ratio $\le (1 + \frac{1}{\ell})^d$.

Proof: The considered problem is NP-complete only for $d > 1$. For $d = 1$ one can actually compute an optimal solution in linear time with the following algorithm: we always place the next interval (= 1-dimensional ball) with its left end at the leftmost point that is not yet covered.

For $d = 2$ and fixed $\ell > 1$ we use two nested applications of the shifting strategy from section 2. We first cut the plane into vertical strips of width $\ell \cdot D$. Then in order to cover the points in such a strip we apply again the shifting strategy to the other dimension. Thus, we cut the considered strip into squares of side length $\ell \cdot D$. We then find optimal coverings of points in such a square by exhaustive search. With $\lceil \ell \cdot \sqrt{2} \rceil^2$ disks of diameter D we can cover an $\ell \cdot D \times \ell \cdot D$ square completely, thus we never need to consider more disks for one square. Further, we can assume that any disk that covers at least two of the given points has two of these points on its border. (For disks that cover only one point the following estimate holds trivially.) Since there are only two ways to draw a circle of given diameter through two given points, we only have to consider $2 \cdot \binom{\tilde{n}}{2}$ possible disk positions--where $\tilde{n}$ is the number of given points in the considered square. Thus we have to check at most $O(\tilde{n}^{\,2\lceil \ell \cdot \sqrt{2} \rceil^2})$ arrangements of disks.

The two nested applications of the shifting stragegy add another factor ℓ^2 to our global time bound.

For $d > 2$ one proceeds analogously with d nested applications of the shifting strategy.

Q.E.D.

We have considered in Theorem 3.1 the problem of covering given points with a minimal number of balls of given size. The method of Theorem 3.1 can easily be generalized to yield approximation schemes for problems where one covers with other objects than balls. For a fixed type of object (of arbitrary fixed shape) we define D as the maximum diameter of such an object. In a manner similar to Theorem 3.1 we cut the considered d-dimensional space in a number of different ways ("shifting") into d-dimensional cubes with sides of length $\ell \cdot D$. One can always find a local algorithm that proceeds by enumeration in the same way as the one for balls in Theorem 3.1. But now the number of objects of the considered type that are needed to cover a d-dimensional cube with sides of length $\ell \cdot D$ will depend on the ratio between D and the maximal $\tilde{D}$ s.t. a d-dimensional cube with sides of length $\tilde{D}$ is contained in a covering object of the considered type. The running time of the resulting approximation algorithm H_ℓ^d will depend exponentially on this ratio $D/\tilde{D}$. We will show in Theorem 5.3 that at least in one important case one can eliminate the ratio $D/\tilde{D}$ from the exponent of the running time by replacing the local enumeration algorithm by another approximation scheme.

In certain applications the covering problem is defined in terms of objects with fixed orientation. This is the case, for instance, with the covering with squares problem in the context of image processing [9]. This additional constraint simplifies the problem in that the following trick often suffices to eliminate $D/\tilde{D}$ from the exponent of the running time. An example is the following corollary.

Corollary 3.2

Consider the problem of covering n given points in d-space with a minimal number of rectilinear blocks (the sides of which have given lengths $D_1,\ldots,D_d$) oriented with sides parallel to the axes. There is a polynomial time approximation scheme H^d such that for every given integer $\ell > 1$, the algorithm H^d_ℓ delivers a cover in $O(\ell^d \cdot n^{2\ell^d+1})$ steps with performance ratio $\leq (1+\frac{1}{\ell})^d$.

This corollary is proved in the same way as Theorem 3.1, except that the cuts orthogonal to the i^{th} axis are introduced at a distance $\ell \cdot D_i$ from each other.

4. Application of the Technique to Packing Problems

In a packing problem one wants to place without overlap a maximal number of objects of given shape within a given area. Since the error analysis of the shifting strategy remains true for such problems, we can use similar algorithms as in section 3. We consider as an example the problem of packing with squares discussed in the introduction. The squares in this case have to be placed such that their sides coincide with lines of an overlaying rectilinear grid. The following theorem can also be generalized to packing problems without such a restriction.

Theorem 4.1 There is a polynomial time approximation scheme for the problem of packing a maximal number of $k \times k$ squares (for a natural number k) into an area that is given by n squares of unit size on a rectilinear grid. The approximation algorithm with parameter ℓ has an error ratio $\leq (1+\frac{1}{\ell})^2$ and runs in time $O(k^2 \cdot \ell^2 \cdot n^{\ell^2})$.

The idea of the proof is to reduce the problem via two nested applications of the shifting strategy to a local packing problem in an $\ell \cdot k \times \ell \cdot k$ square.

Remark: We also get polynomial approximation schemes for many packing problems in higher dimensions, with arbitrary orientations and with objects other than squares.

5. Covering with Nonconvex Objects in One Dimension: The Ring Cover Problem

In one dimension a ring of size $\langle r,w \rangle$ is simply a pair of closed intervals of length w with distance $2r$ in between. Unlike most other geometric location problems, the ring cover problem does not become easy in one dimension (in one dimension one has to cover n given points on a line by a minimal number of pairs of intervals as above).

Theorem 5.1 (Maass [8]) The ring cover problem in one dimension is strongly NP-complete.

On the other hand, we have noted in the proof of Theorem 3.1 that one can solve (optimally) this problem for $\frac{r}{w} = 0$ in linear time. A natural question is then, how does the running time of an optimal algorithm depend on the "nonconvexity measure" $\frac{r}{w}$ for the covering rings of size $\langle r,w \rangle$.

Theorem 5.2 (Maass [8]) The ring cover problem in one dimension for n given points and rings of size <r,w> can be solved optimally in time $O(\lceil\frac{r}{w}\rceil^2 \cdot \log n \cdot n^{16\cdot\lceil\frac{r}{w}\rceil+18})$. In particular for every fixed bound on $\frac{r}{w}$ the problem is in P.

Concerning approximation algorithms two issues arise. The large degree of the polynomial time bound in Theorem 5.2 leads to the question, whether at least for common values of the parameters one can find reasonably fast approximation algorithms. We consider this topic in section 6. Second, from a more theoretical point of view one would like to know whether for unbounded values of $\frac{r}{w}$ (i.e., the strongly NP-complete problem of Theorem 5.1) a polynomial time approximation scheme exists. We give a positive answer in Theorem 5.3. According to the general discussion in section 3 the parameter $D/\tilde{D}$ --which is equal to $2\frac{r}{w}+2$ in the present situation--appears in the exponent of the time bound for the approximation algorithms of section 3. Thus, the methods of section 3 do not suffice.

The following result relies on a more subtle method where one employs a nested application of the shifting strategy even in the considered one-dimensional case (see [5] for details).

Theorem 5.3 There is a polynomial time approximation scheme C s.t. for every given natural number $\ell \geq 1$ the algorithm C_ℓ computes for any given <r,w> and any given n points on a line in time $O(\ell^4(4n)^{\ell^2})$ an approximate solution to the one-dimensional ring cover problem with performance ratio $\leq (1+\frac{1}{\ell})^2$.

6. Fast Approximation Algorithms

The time bounds of many approximation algorithms in this paper can be improved considerably by using more efficient algorithms for the computation of optimal solutions in combination with the shifting strategy. We mention in this section two examples in the case of the one-dimensional ring cover problem: a nearly quadratic time algorithm with at most 50% error and, further, for rings with $\frac{r}{w} < \frac{1}{2}$ (this appears to be the more typical case in robotics) for every $\varepsilon > 0$ an approximation algorithm that is linear in n. Both results require a more detailed analysis of the mathematical structure of optimal local coverings (see [4]). On the basis of such insight one can drastically shorten the exhaustive search for an optimal local solution: we show that it is enough to try out only a few representative types of local solutions.

Theorem 6.1 There is an algorithm that computes a $\frac{1}{2}$-approximate solution of the one-dimensional ring cover problem in $O(\lceil\frac{r}{w}\rceil \cdot n^2 \cdot \log n)$ steps.

Theorem 6.2 For rings of size <r,w> with $\frac{r}{w} < \frac{1}{2}$ there is a polynomial time approximation scheme for the one-dimensional ring cover problem that covers n given points on the line for any $\varepsilon > 0$ in $O(\lceil\frac{1}{\varepsilon}\rceil \cdot 2^{2\lceil\frac{1}{\varepsilon}\rceil} \cdot n \cdot \log n)$ steps with ratio $\leq 1+\varepsilon$.

7. Summary

The approach described in this paper, the shifting strategy, has proved useful in a large variety of contexts. Via the use of this approach we were able to derive algorithms that are the best possible in the sense that the exponential dependence on $\frac{1}{\varepsilon}$ cannot be removed unless NP=P. We also note that all other polynomial approximation schemes that we are familiar with rely on dynamic programming. The technique we introduced is an alternative to dynamic programming for the construction of polynomial approximation schemes for strongly NP-complete problems.

References

[1] John J. Bartholdi III, "A Guaranteed-Accuracy Round-Off Algorithm for Cyclic Scheduling and Set Covering," Operations Research, Vol. 29, No. 3 (1981).

[2] F. D. Berman, F. T. Leighton and L. Snyder, "Optimal Tile Salvage," unpublished manuscript (1982).

[3] R. J. Fowler, M. S. Paterson, and S. L. Tanimoto, "Optimal Packing and Covering in the Plane Are NP-Complete," Inform. Process. Lett., 12 (1981), 133-137.

[4] M. R. Gary and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman (1978).

[5] D. S. Hochbaum and W. Maass, "Fast Approximation Algorithms for the Robot Placement Problem," manuscript, U.C. Berkeley (1983), submitted.

[6] D. S. Hochbaum and W. Maass, "Approximation Algorithms for Covering and Packing Problems in Image Processing and VLSI," manuscript, U.C. Berkeley (May 1983), submitted.

[7] D. S. Johnson, "The NP-Completeness Column: An Ongoing Guide," Journal of Algorithms, 3, 182-195 (1982).

[8] W. Maass, "On the Complexity of Computing Optimal Positions for Industrial Robots," U.C. Berkeley, manuscript, submitted.

[9] S. L. Tanimoto and R. J. Fowler, "Covering Image Subsets with Patches," Proc. 5th Inter. Conf. on Pattern Recognition, (1980) 835-839.