

Approximation Schemes for Covering and Packing Problems in Image Processing and VLSI

DORIT S. HOCHBAUM AND WOLFGANG MAASS

University of California, Berkeley, California

Abstract. A unified and powerful approach is presented for devising polynomial approximation schemes for many strongly NP-complete problems. Such schemes consist of families of approximation algorithms for each desired performance bound on the relative error $\epsilon > 0$, with running time that is polynomial when ϵ is fixed. Though the polynomiality of these algorithms depends on the degree of approximation ϵ being fixed, they cannot be improved, owing to a negative result stating that there are no fully polynomial approximation schemes for strongly NP-complete problems unless $NP = P$.

The unified technique that is introduced here, referred to as the shifting strategy, is applicable to numerous geometric covering and packing problems. The method of using the technique and how it varies with problem parameters are illustrated. A similar technique, independently devised by B. S. Baker, was shown to be applicable for covering and packing problems on planar graphs.

Categories and Subject Descriptors: F.2.2 [Analysis of Algorithms and Problem Complexity]: Nonnumerical Algorithms and Problems—*geometrical problems and computations*

General Terms: Algorithms, Theory

Additional Key Words and Phrases: Covering, packing, covering points in the Euclidean space, image processing, VLSI, shifting strategy, worst case analysis of heuristics, polynomial approximation scheme

1. Introduction

Polynomial approximation schemes are described in this paper for several strongly NP-complete problems that have important applications in the areas of robot motion planning, VLSI design, image processing, and location. These problems appear in the contexts of covering and packing with convex objects. One of them is the *square packing* problem, which 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. 2×2 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 2×2 arrays into the array of working chips on a wafer. (See the result of Berman et al. [2], reviewed by Johnson [6], and the NP-completeness result of Fowler et al. [3].)

The research of the first author was supported in part by the National Science Foundation under grant ECS 82-04695. During the preparation of this paper, the second author was supported by the Heisenberg Programm der Deutschen Forschungsgemeinschaft, West Germany.

Authors' address: D. S. Hochbaum, School of Business Administration, 350 Barrows Hall, University of California at Berkeley, Berkeley, CA 94720; W. Maass, Department of Mathematics, University of Illinois, Chicago, IL 60680.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1985 ACM 0004-5411/85/0100-0130 \$00.75

Another problem is *covering with disks*, that is, given points in the plane, to identify a minimally sized set of disks (of prescribed radius) covering all points. One of its applications is in the area of locating emergency facilities such that all potential customers will be within a reasonably small radius around the facility. (The complexity results for this problem are reviewed in [6].)

A third problem considered is *covering with squares or (rectangles)*, which has an important application to image processing, discussed in Tanimoto and Fowler [7]. Here one wants to store information in square “patches” such that all points with information (“pixels”) are contained in at least one of the patches. The general problem can be described as: Given points in a Euclidean space (in this application, on a grid), find a minimally sized set of squares of prescribed size covering all those points.

None of the above problems was reported to have a bounded error ratio approximation algorithm. 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 $\epsilon > 0$ 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 increase monotonically with $1/\epsilon$. If the functional dependence of the running time on the size of the input and $1/\epsilon$ 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, theorem 6.8]). This negative result, however, does not exclude the existence of a polynomial approximation scheme for these problems, that is, a family of algorithms such that for any specified relative error $\epsilon > 0$ there is an ϵ -approximation algorithm in the scheme that is polynomial. Though such schemes are conceptually feasible, their existence has rarely been 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. A similar technique has been independently discovered by Baker [1]. Her technique is applicable to planar graphs, whereas ours applies to problems defined in Euclidean space; but our concepts of strips of bounded width and shifting are analogous to the concepts of bounded outerplanarity and shifting used by Baker.

The shifting strategy is described and the “shifting lemma” is proved in Section 2. We then proceed in Section 3 with its application to problems of covering with balls and derive a polynomial approximation scheme for this case. We further explain the behavior of such schemes for problems of covering with arbitrarily shaped objects in higher dimensional spaces. In Section 4 we describe the application to packing problems with convex objects, and we summarize our results in Section 5.

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 $|\text{OPT}|$.

2. The Shifting Strategy

The shifting strategy allows us to bound the error of the simple divide-and-conquer approach by applying it repetitively and selecting the single most favorable resulting solution. This approach was successfully used for problems of covering with nonconvex objects as well [5]. In this section, however, we illustrate the application of the shifting strategy for covering with planar balls (disks).

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 , where each strip is left closed and right open. Groups of l consecutive strips, resulting in strips of width $l \cdot D$ each, are considered. For any fixed subdivision of I into strips of width D , there are l different ways of partitioning I into strips of width $l \cdot D$. 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 the l distinct shift partitions that result by S_1, S_2, \dots, S_l .

Let A be any algorithm that delivers a solution in any strip of width $l \cdot D$ (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 a 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 , $i = 1, 2, \dots, l$. 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 $A(S_1), \dots, A(S_l)$.

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

LEMMA 2.1 (THE SHIFTING LEMMA).

$$r_{S_A} \leq r_A \left(1 + \frac{1}{l} \right), \quad (2.1)$$

where A is a local algorithm and l is the shifting parameter.

PROOF. We produce an upper bound on the sum of errors caused by all algorithms $A(S_i)$ for $i = 1, 2, \dots, l$.

By the definition of r_A we have

$$Z^{A(S_i)} \leq r_A \cdot \sum_{J \text{ in } S_i} |\text{OPT}_J|, \quad (2.2)$$

where J runs over all strips in partition S_i and $|\text{OPT}_J|$ is the number of disks in an optimal cover of the points in strip J .

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

$$\sum_{J \text{ in } S_i} |\text{OPT}_J| \leq |\text{OPT}| + |\text{OPT}^{(i)}|. \quad (2.3)$$

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 $\text{OPT}^{(1)}, \dots, \text{OPT}^{(l)}$ are disjoint

and can add up to OPT at most. It follows that

$$\sum_{i=1}^l (|\text{OPT}| + |\text{OPT}^{(i)}|) \leq (l + 1) \cdot |\text{OPT}|. \quad (2.4)$$

Expressions (2.3) and (2.4) imply

$$\min_{i=1, \dots, l} \sum_{J \text{ in } S_i} |\text{OPT}_J| \leq \frac{1}{l} \sum_{i=1}^l \left(\sum_{J \text{ in } S_i} |\text{OPT}_J| \right) \leq \left(1 + \frac{1}{l} \right) |\text{OPT}|. \quad (2.5)$$

Combining inequality (2.5) with (2.2), we obtain

$$Z^{S_A} = \min_{i=1, \dots, l} Z^{A(S_i)} \leq r_A \cdot \left(1 + \frac{1}{l} \right) \cdot |\text{OPT}|, \quad (2.6)$$

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 convex objects.

THEOREM 3.1. *Let $d \geq 1$ be some finite dimension. Then there is a polynomial time approximation scheme H^d such that for every given natural number $l \geq 1$, the algorithm H^d delivers a cover of n given points in a d -dimensional Euclidean space by d -dimensional balls of given diameter D in $O(l^d(l \cdot \sqrt{d})^d \cdot (2n)^{d(l \cdot \sqrt{d})^d + 1})$ steps with performance ratio $\leq (1 + 1/l)^d$.*

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 (i.e., 1-dimensional ball) with its left end at the leftmost point that is not yet covered.

For $d = 2$ and fixed $l \geq 1$ we use two nested applications of the shifting strategy from Section 2. We first cut the plane into vertical strips of width $l \cdot D$. Then, in order to cover the points in such a strip, we apply the shifting strategy to the other dimension. Thus, we cut the considered strip into squares of side length $l \cdot D$. We find optimal coverings of points in such a square by exhaustive search. With $(l \cdot \sqrt{2})^2 = 2l^2$ disks of diameter D we can cover an $l \cdot D \times l \cdot D$ square compactly; 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 $O(\tilde{n}^{2(l \cdot \sqrt{2})^2})$ arrangements of disks. We specify the position of each disk by its center. In order to check whether an arrangement of disks is a feasible cover of the \tilde{n} points in the square, we need to determine for each point whether it is within a distance of at most $D/2$ from one of the centers. Such a check will require

$O(l^2 \cdot \tilde{n})$ steps, with the assumption that we can determine in one step the distance between two points on the plane or the center of a disk with the necessary precision. (Note that it is possible to speed up that process by sorting the disk centers in $O(l^2)$ prespecified subsquares of the region and then determining in which of the subsquares there could be the potential covering disk by binary search in time $O(\log l)$.) The two nested applications of the shifting strategy 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 objects other 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 $l \cdot D$. One can always find a local algorithm that proceeds by enumeration in the same way as the algorithm 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 $l \cdot D$ will depend on the ratio between D and the maximal \tilde{D} such that 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} . For instance, for objects with known orientation the expression $(l \cdot \sqrt{d})^d$ in the exponent will be replaced by $(l \cdot D/\tilde{D})^d$. This ratio D/\tilde{D} is usually of interest also in other contexts. Note that, e.g., for rectangles of size $a \times b$ in two dimensions this ratio is closely related to the "aspect ratio" $\max\{a/b, b/a\}$. We have shown in another paper [5] that in at least 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 [7]. This additional constraint simplifies the problem in that the trick illustrated in the following corollary often suffices to eliminate D/\tilde{D} from the exponent of the running time.

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, \dots, D_d) oriented with sides parallel to the axes. There is a polynomial-time approximation scheme H^d such that for every given integer $l \geq 1$, the algorithm H_l^d delivers a cover in $O(l^d \cdot n^{2l^d+1})$ steps with performance ratio $\leq (1 + 1/l)^d$.*

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 $l \cdot D_i$ 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 with a given area. Since the error analysis of the shifting strategy remains true for such problems, we can use algorithms similar to those 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 so that their sides

coincide with the 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 $\leq (1 + 1/l)^2$ and runs in time $O(k^2 \cdot l^2 \cdot n^2)$.*

PROOF. One reduces the problem via two nested applications of the shifting strategy to a local packing problem in an $l \cdot k \times l \cdot k$ square. The local packing problem can be solved by enumeration in $O(\tilde{n}^l)$ steps, where \tilde{n} is the number of given unit squares that fall into the considered $l \cdot k \times l \cdot k$ square. We assume here that a list of all possible placements of $k \times k$ squares that lie totally inside the area of the n given unit squares has been generated as part of the input.

The shifting strategy that has been described in Section 2 for the covering problem can be translated to packing problems with the help of the following observation. The absolute error that results from solving the packing problem separately in each strip for a given shift can be bounded by the number of $k \times k$ squares in an optimal global solution of the packing problems that include unit squares in two different strips. Q.E.D.

For fixed l one has the option of speeding up the corresponding approximation algorithm by using some faster algorithm for optimal packings in $l \cdot k \times l \cdot k$ squares (see, e.g., the analogous approach in [5, sec. 5]).

Remark. We also get polynomial approximation schemes for many packing problems in higher dimensions, with arbitrary orientations and with other objects than squares—see the analogous discussion in Section 3.

5. Conclusion

All the approximation schemes described in this paper use a local algorithm that solves the covering or packing problem in a cube with sides of length $l \cdot D$. An interesting direction for future research is to find optimal local algorithms that are more efficient than enumeration. Another alternative is to consider for such local problems a heuristic that belongs to some approximation scheme. Together with the shifting strategy defined in this paper such a heuristic, if extant, will produce a faster overall approximation algorithm.

The results of this paper and Baker's paper [1] raise the question of whether there exist polynomial approximation schemes also for nongeometric and nonplanar covering and packing problems. In general, it would be desired to find additional strongly NP-complete problems that have polynomial-time approximation schemes.

REFERENCES

1. BAKER, B. S. Approximation algorithms for NP-complete problems on planar graphs. In *Proceedings of the 24th Annual IEEE Symposium on Foundations of Computer Science* (Tucson, Ariz., Nov. 7–9). IEEE, New York, 1983, pp. 265–273.
2. BERMAN, F. D., LEIGHTON, F. T., AND SNYDER, L. Optimal tile salvage. Unpublished manuscript, 1982.
3. FOWLER, R. J., PATERSON, M. S., AND TANIMOTO, S. L. Optimal packing and covering in the plane are NP-complete. *Inf. Proc. Lett.* 12, 3 (June 1981), 133–137.

4. GAREY, M. R., AND JOHNSON, D. S. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, 1978.
5. HOCHBAUM, D. S., AND MAASS, W. Fast approximation schemes for geometric covering and packing problems in robotics and VLSI. Unpublished manuscript, University of California, Berkeley, Calif., 1983.
6. JOHNSON, D. S. The NP-completeness column: An ongoing guide. *J. Algorithms* 3, 2 (June 1982), 182-195.
7. TANIMOTO, S. L., AND FOWLER, R. J. Covering image subsets with patches. In *Proceedings of the 5th International Conference on Pattern Recognition*, 1980, pp. 835-839.

RECEIVED AUGUST 1983; REVISED AUGUST 1984; ACCEPTED SEPTEMBER 1984