Algorithmica (1993) 9: 1–22

Algorithmica

The Slab Dividing Approach To Solve the Euclidean *P*-Center Problem

R. Z. Hwang,[1] R. C. T. Lee,[1,2] and R. C. Chang[2,3]

Abstract. Given n demand points on the plane, the Euclidean P-Center problem is to find P supply points, such that the longest distance between each demand point and its closest supply point is minimized. The time complexity of the most efficient algorithm, up to now, is $O(n^{2P-1} \cdot \log n)$. In this paper, we present an algorithm with time complexity $O(n^{O(\sqrt{P})})$.

Key Words. Computational geometry, NP-completeness.

1. Preliminaries. *The Euclidean P-Center* (EPC) *problem* is defined as follows. Given a set D of n demand points on the plane, find a set S of P supply points such that the furthest distance between demand points and their closest supply points is as close as possible. There are many applications in the real world for this problem. One of them is to find P positions to set up fire departments such that the longest distance between each house and its closest fire department is minimized. The EPC problem can be formally formulated as follows:

> Given a set of n demand points $D = \{d_1, d_2, \ldots, d_n\}$, find a set of P supply points $S = \{s_1, s_2, \ldots, s_P\}$, such that
>
> $$\max_{1 \le i \le n} \left\{ \min_{1 \le j \le P} \{\text{dist}(d_i, s_j)\} \right\} \text{ is minimized,}$$
>
> where $\text{dist}(d_i, s_j)$ is the Euclidean distance between d_i and s_j.

Megiddo and Supowit (1984) proved that the EPC problem is NP-hard. Drezner (1984) proposed an algorithm with time $O(n^{2P+1} \cdot \log n)$ for this problem, and it can be revised to $O(n^{2P-1} \cdot \log n)$ by combining it with the result that the Euclidean 1-Center problem can be solved in time $O(n)$ (Megiddo, 1983). This combining method is similar to that in Drezner (1987) which solved some center problems corresponding to the rectilinear distance.

In this paper we propose a new technique, the slab dividing method, to solve the EPC problem with time $O(n^{O(\sqrt{P})})$. In the next section, we review the paper

[1] Institute of Computer Science, National Tsing Hua University, Hsinchu, Taiwan, Republic of China.
[2] Academia Sinica, Taipei, Taiwan, Republic of China.
[3] Institute of Computer Science, National Chiao Tung University, Hsinchu, Taiwan, Republic of China.

Received May 20, 1989; revised March 15, 1990. Communicated by Kurt Mehlhorn

proposed by Drezner (1984). In Section 3, we state the major idea of the slab dividing method. The detail steps and proofs are described in Sections 4–7.

2. Previous Results. In this section, we shall briefly discuss the method proposed by Drezner (1984) from the geometric viewpoint. (The method introduced in this section is the same as that in Drezner (1984); only the form of presentation is different.) First, we shall define another problem.

DEFINITION (The P-Circle Covering (PCC) Problem). Given n demand points on the plane, find the smallest radius r and a set S of P points, such that the circles centered at the points in S with radius r can cover all demand points.

Given a set D of n demand points on the plane and an optimal solution of the EPC problem, let S be the set of the supply points and let r be the longest distance between each demand point and its closest supply point in this solution. It is obvious that r and S form an optimal solution of the PCC problem for the input D. Also, if we obtain an optimal solution of the PCC problem, we have found an optimal solution of the EPC problem. In this paper, we call the set S of points in an optimal solution of the PCC problem *the solution centers* and we call the radius r *the solution radius*. In the following paragraphs, we describe the EPC problem in terms of the PCC problem.

There are two major results shown in Drezner (1984) for the PCC problem. One is that there are $O(n^3)$ *possible radii*. This means that we can find $O(n^3)$ radii and one of them is the solution radius. The other is that there are $O(n^2)$ *possible circle centers* for a given radius r. This means that given a radius r, we can find a set S^* of $O(n^2)$ points, such that if r is the solution radius, then there exist P circles of radius r centered at S', $S' \subset S^*$, which can cover all demand points. From the above two results, we can solve the PCC problem by the following way: Sort the possible radii. Then we choose one of them, say r', and we ask the following question: Can P circles of radius r' cover the n demand points? To answer this question, we first find the set S^* of possible circle centers for the radius r' and draw circles with radius r' centered at the centers in S''. Select any combinations of P circles and then check whether these P circles cover all the n demand points. If there exist P circles which can cover all the n demand points, we choose another radius r'', $r'' < r'$, from the possible radii, otherwise we choose r'', $r'' > r'$. (Do the binary search on the sorted possible radii.) Then repeat the above steps again, until the optimal radius is found.

Since there are $O(n^2)$ possible circle centers, we have $C_P^{O(n^2)}$ selections, and it takes $O(n)$ time to check whether these P circles cover all points, and $O(\log n^3) = O(\log n)$ to do the binary search on the possible radii. So the time complexity is $O(n^{2P+1} \cdot \log n)$.

Now let us see how Drezner (1984) showed that there are at most $O(n^3)$ possible radii. Drezner pointed out that given a set of points, the smallest circle covering all these points must be defined by one, two, or three points. For the circle defined by three points, these three points define the boundary of a smallest circle enclosing all three of them. For the case defined by two points, they form the diameter of this circle. A circle defined by only one point is a degenerated case, where the radius of this circle can be considered as zero and the entire circle contracts to one point. Thus it is obvious that for the P circles in an optimal solution of the PCC problem, at least one circle is defined by one of the above cases, or else we can contract all circles and find another radius which is smaller than the solution radius.

It is obvious that there are C_1^n, C_2^n, and C_3^n circles defined by one, two, and three points, respectively. We call these circles *the bounding circles*. Then the solution radius must be equal to one of the radii of the bounding circles.

Next, let us see how Drezner (1984, 1981) showed that there are at most $O(n^2)$ possible circle centers for a given radius r. First draw circles with radius r centered at all demand points. Let S^* be the set of all intersection points of these circles. For any circle centered at the points in S covering a set D' of demand points, it is obvious that we can move this circle (without changing the radius) such that at least two points, denoted by d_1 and d_2, $d_1, d_2 \in D'$, are on the circle boundary and this circle also covers all points in D'. Let c' be the new circle center. We know that $\text{dist}(d_1, c') = \text{dist}(d_2, c') = r$, so the new circle center c' must belong to S^*. Therefore given an optimal solution of the PCC problem, we can move all circles, such that these new circles are centered at the points in S^* and also cover all demand points. Thus if there is a set S of P circles with radius r covering all demand points then there are P circles with radius r centered at S'', $S'' \subset S^*$, which also cover all demand points. Since the number of the points in S^* is $O(n^2)$, we conclude that there are $O(n^2)$ possible circle centers for a given radius r.

In our algorithm, we also use the above two results and apply the binary search approach. Our basic problem is: Given a set D of n demand points and two parameters P and r, determine whether there exist P circles of radius r which can cover all demand points. We call this problem *the* (P, r) *circle covering problem* (*the* (P, r) *CC problem*). In the next section, we propose a procedure, called Procedure CIRCLE_COVER, which can be used to solve the (P, r) CC problem. With this procedure, we have the following algorithm to solve the PCC problem (the EPC problem).

Algorithm P-Center(D, P, r, S)

Input: A set D of n demand points and a number P.

Output: Return a minimum radius r, and P circles centered at the points in S which can cover all demand points in D.

Step 1. Generate a set of possible radii by using the algorithm in Drezner (1984).
Step 2. Sort the above radii in increasing order, and name them as $r_1, r_2, \ldots, r_k$.
Step 3. Let Low := 1 and High := k.
(From this step, we begin a binary search.)
Step 4. Let Med := $\lceil (\text{Low} + \text{High})/2 \rceil$.
Step 5. If CIRCLE_COVER$(D, P, r_{\text{Med}}, S)$ = "FALSE,"
then Low := Med, else High := Med.
Step 6. If High $\neq$ Low
then go to step 4,
else return S and $r = r_{\text{Med}}$.

Now let us show the time complexity of the above algorithm. From Drezner (1984), we know that steps 1 and 2 need $O(n^3 \cdot \log n)$ steps. Steps 3–7 perform a binary search on $O(n^3)$ radii, so the time needed is also $O(n^3 \cdot \log n)$. In the next section, we will show that Procedure CIRCLE_COVER can be solved in time $O(n^{O(\sqrt{P})})$. Therefore the time complexity of the above algorithm is $O(n^{O(\sqrt{P})}) \cdot \log n = O(n^{O(\sqrt{P})})$.

In the next section, we state the major idea about how to solve the (P, r) CC problem in time $O(n^{O(\sqrt{P})})$.

3. The Slab Dividing Method. In this section, we shall introduce our slab dividing method to solve the (P, r) CC problem. Note that the (P, r) CC problem is an NP-hard problem. Therefore we do not expect that this problem can be solved by the traditional divide-and-conquer method (Horowitz and Sahni, 1978; Aho *et al.*, 1974; Bentley, 1980). Yet, we shall show later that once an optimal solution of a (P, r) CC problem instance is given, we can use part of this solution to divide the input data into two subsets D_a and D_c, such that the (P, r) CC problem can be solved by first solving the (P, r) CC problems defined on D_a and D_c, respectively, and then merging the sub-solutions. Consider Figure 1(a), which contains 54

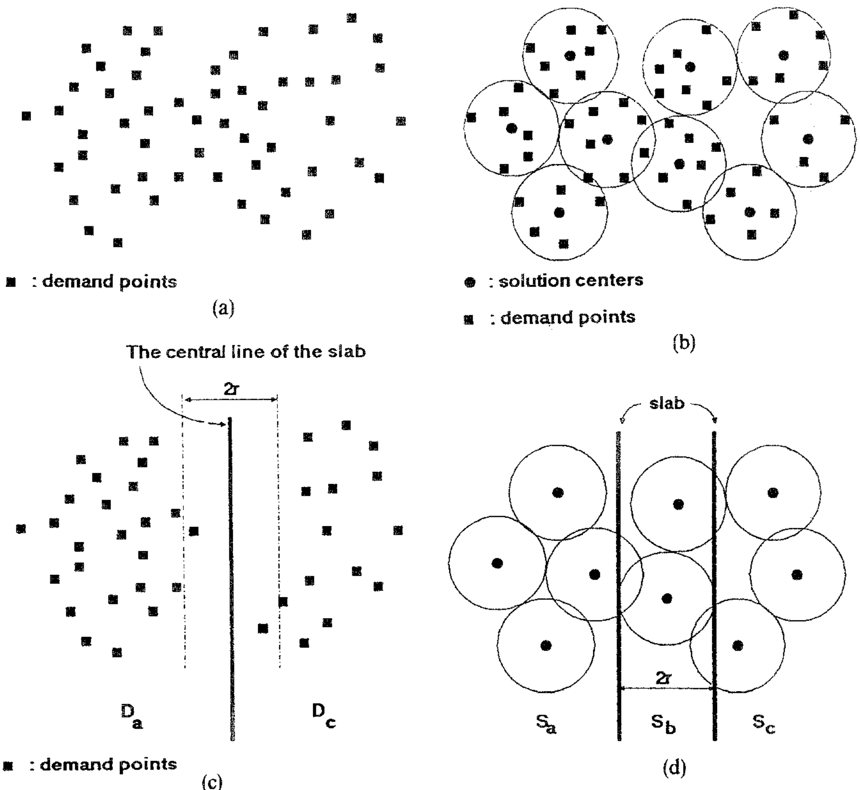


Fig. 1

points. Figure 1(b) shows an optimal solution of a $(9, r)$ CC problem defined on this set of data. Next, we draw a slab with width $2r$ which divides the solution centers into three subsets S_a, S_b, and S_c, as shown in Figure 1(c). Note that the solution centers on the boundaries of this slab are assigned to S_b.

If we remove the demand points covered by S_b and divide the remaining demand points into D_a and D_c, by the central line of this slab, as shown in Figure 1(d), we can see that because of the width $2r$, circles of radius r centered at the solution centers in S_a (resp. S_c) cannot cover the demand points in D_c (resp. D_a). This property guarantees that the two subproblem instances are independent and we call this *the independent property of the slab.*

Because of the independent property, we can see that given an instance of a (P, r) CC problem and a slab with width $2r$, if we know the corresponding S_b and the numbers of points in S_a and S_c in advance, then we can divide the problem into two independent subproblems. One is the $(|S_a|, r)$ CC problem with D_a as input; another is the $(|S_c|, r)$ CC problem with D_c as input. An optimal solution can be obtained by merging S_b and the two solutions in the two subproblems.

In Sections 4 and 5, we will show that in an optimal solution of a (P, r) CC problem instance, there exists a slab with width $2r$ which divides the solution centers into three subsets, S_a, S_b, and S_c, where S_b is the set of solution centers in the median part (including the centers on the two boundary lines), and S_a (resp. S_c) is the set of the solution centers to the left (resp. right) of the slab, such that S_a, S_b, and S_c satisfy the following properties:

(1) the number of points in S_b is no more than $K_s = O(\sqrt{P})$;
(2) the number of points in both S_a and S_c is no more than $\lceil 2P/3 \rceil$.

The slab which satisfies the above properties is called *the dividing slab* of this optimal solution and the above properties are called *the dividing slab properties.*

Our algorithm is based upon two procedures. One is called Procedure GEN_SLABS which can generate a set L_s of slabs and one of them will be the dividing slab of an optimal solution. The other is called Procedure GEN_SUPPNTS, with the set D of all demand points and a slab as inputs. Its output is a set of partial solutions. Each partial solution is a set of circle centers. Furthermore, if the slab is a dividing slab of an optimal solution, then one of the partial solutions will be S_b. Therefore, we may call these partial solutions *the candidates of* S_b. We also guarantee that the size of each candidate, produced by Procedure GEN_SUPPNTS, is no more than $O(\sqrt{P})$.

According to the above properties and the procedures, we can solve the (P, r) CC problem by the following way. First, we call Procedure GEN_SLABS to generate a set L_s of slabs. Then for each slab l, we generate a set S'' of candidates of S_b by calling Procedure GEN_SUPPNTS. For each candidate S'_b of S_b in S'', we draw circles centered at the points in S'_b. Next remove the demand points in D which are covered by these circles and divide the remaining uncovered points into two subsets, D_a and D_c. If l is a dividing slab, from the second property of the dividing slab, we know that there exist no more than $\lceil 2P/3 \rceil$ circles of radius r which can cover all demand points in $D_a(D_c)$. Thus, we recursively call Procedure

CIRCLE_COVER to find $i \le \lceil 2P/3 \rceil$ circles of radius r to cover all points in D_a. Since the total number of circles in an optimal solution is no more than P, we check whether $P - i - |S'_b|$ circles of radius r can cover all demand points in D_c. According to the second property, the number $P - i - |S'_b|$ will be no more than $\lceil 2P/3 \rceil$. If we can find the solutions of the two subproblems, we merge back the solutions; otherwise we try next instance. The detail steps are stated in the following procedure.

Procedure CIRCLE_COVER(D, P, r, S)

Input: A set D of n demand points, a number P, and a radius r.

Output. Return "TRUE," and a set of S of solution centers, if P circles of radius r can cover all points in D; otherwise return "FALSE."

Step 1. If $P < 3$, then use Drezner's algorithm (Drezner, 1984) to solve this problem, else do the following steps.
Step 2. Generate a set L_s of candidates of the dividing slab by using Procedure GEN_SLABS.
Step 3. For each slab l in L_s do:
Step 4. Call Procedure GEN_SUPPNTS to generate a set S'' of sets of candidates of S_b.
Step 5. For each set S'_b in S'' do:
Step 6. Draw the circles of radius r centered at the points in S'_b. Let D' be the set of points in D which are not covered by these circles. Let D_a (resp. D_c) be the set of points to the right (resp. left) of the central line of l.
Step 7. For $i = 0$ to $\lceil 2P/3 \rceil$ do:
Step 8. If $(P - i - |S'_b|) \le \lceil 2P/3 \rceil$ do:
Step 9. Call T_1 = CIRCLE_COVER(D_a, j, r, S_1).
Step 10. Call T_2 = CIRCLE_COVER(D_c, $P - i - |S'_b|$, r, S_2).
Step 11. If $T_1 = T_2 =$ "TRUE," then return "TRUE" and $S = S_1 \cup S_2 \cup S'_b$.
Step 12. Return "False."

Now let us analyze the time complexity of the above procedure. Let $T(P)$ be the time complexity of this procedure. The time complexity needed between step 7 and step 11 can be formulated by using the term $T(P)$ as follows:

$$\begin{aligned} O\left(\sum_{i=0}^{\lceil 2P/3 \rceil} (T(i) + T(P - |S'_b| - i))\right) &\le O\left(\sum_{i=0}^{\lceil 2P/3 \rceil} T(i) + T(P - i)\right) \\ &\le O(2 \cdot (\lceil 2P/3 \rceil) \cdot T(\lceil 2P/3 \rceil)) \\ &= O(P \cdot T(\lceil 2P/3 \rceil)). \end{aligned}$$

Because $|S'_b|$ is bounded by $O(\sqrt{P})$, Step 6 needs $O(\sqrt{P})$ steps to draw the circles, $O(n \cdot \sqrt{P})$ steps to remove the uncovered demand points, and $O(n)$ steps to divide the uncovered points for each slab. Therefore the time needed between step 6 to step 11 is $O(n \cdot \sqrt{P}) + O(P \cdot T(\lceil 2P/3 \rceil))$.

Because steps 2–5 are concerned with two unknown procedures, for analyzing, we will define some notations. Let $T_1(P)$ (resp. $T_2(P)$) be the time complexity of Procedure GEN_SLABS (resp. Procedure GEN_SUPPNTS), and let $N_1(P)$ (resp. $N_2(P)$) be the number of slabs (resp. the candidates of S_b) generated in Procedure GEN_SLABS (resp. Procedure GEN_SUPPNTS). Now we can see that the time needed in this procedure is

$$T(P) = T_1(P) + N_1(P) \cdot (T_2(P) + N_2(P) \cdot (O(n \cdot \sqrt{P}) + O(P \cdot T(\lceil 2P/3 \rceil)))).$$

Later, in Sections 4 and 7, we will show that

$$(1) \quad T_1(P) = O(T(P/3) + P^{3/2} \cdot n), \qquad N_1(P) = O(\sqrt{P}),$$

and

$$(2) \quad T_2(P) = O(n^{O(\sqrt{P})}), \qquad N_2(P) = O(n^{O(\sqrt{P})}).$$

Based upon the above results, the time complexity becomes

$$T(P) = O(T(P/3) + P^{3/2} \cdot n) + O(\sqrt{P}) \cdot (O(n^{O(\sqrt{P})}) \cdot (O(n \cdot \sqrt{P}) + O(P \cdot T(\lceil 2P/3 \rceil)))),$$

$$T(P) = O(\sqrt{P} \cdot (n^{O(\sqrt{P})} \cdot (P \cdot T(\lceil 2P/3 \rceil)))),$$

$$T(P) = O(n^{O(\sqrt{P})} \cdot T(\lceil 2P/3 \rceil)),$$

$$T(P) = O(n^{O(\sqrt{P})}).$$

In the next section, we shall discuss Procedure GEN_SUPPNTS and its relative complexities $T_2(P)$ and $N_2(P)$. The first and second properties of the dividing slab are discussed in Sections 5 and 6, respectively. $T_1(P)$, $N_1(P)$, and Procedure GEN_SLABS are discussed in Section 7.

4. Generating the Candidates of S_b. In this section, we shall discuss the details about Procedure GEN_SUPPNTS.

From Drezner (1984), we can generate a set of $O(n^2)$ possible solution centers for a given radius (see Section 2). From the first property of the dividing slab, we know that the number of points in S_b is no more than K_s. Combining these two results, we can select any i points from the set of possible solution centers as the candidate of S_b, where i ranges from 0 to K_s. The following procedure states the detailed steps.

Procedure GEN_SUPPNTS(r, l, S'')
Input: A radius r and a slab l.
Output: A set S'' of sets of candidates of S_b. If l is a dividing slab, then $S_b \in S''$.

Step 1. Generate the set S^* of possible solution centers, for the given radius r, by using the method in Drezner (1984) (see Section 2). Let S_b'' be the set of points which belong to S^* and are in the slab l.
Step 2. For $i = 0$ to K_s do:
Step 3. Enumerate all subsets of i points from S_b''. Add these subsets into S''.

It is obvious that if l is a dividing slab, then S_b must belong to S''. Now let us analyze the time complexity of the above procedure. From the discussion in Section 2, we know that $|S^*| = O(n^2)$ and step 1 takes $O(n^2)$ steps. Because $K_s = O(\sqrt{P})$, steps 2–3 take $C_0^{O(n^2)} + C_1^{O(n^2)} + \cdots + C_{K_s}^{O(n^2)} = O(n^{O(\sqrt{P})})$ and the number of candidates generated in Step 3 is also $O(n^{O(\sqrt{P})})$. Therefore $T_2(P) = O(n^{O(\sqrt{P})})$ and $N_2(P) = O(n^{O(\sqrt{P})})$.

5. The First Property of the Dividing Slab. To prove the properties of the dividing slab, we arrange the slabs in such a way that their central lines intersect in a common reference point and if there are L slabs, the angle between two consecutive slabs is π/L, as shown in Figure 2. In this section, we would show that if L is large enough, one of these slabs satisfies the first property of the dividing slab. In the next section, we show that we can determine a reference point, such that all the L slabs can satisfy the second property.

Before the formal discussion, let us see the example in Figure 3. Assume that the right circles in Figure 3 constitute an optimal solution of some $(8, r)$ CC problem. We draw four slabs in this figure. We can see that some slabs contain more solution centers and some contain less. Here slab 2 contains the smallest number of supply points. Later we shall show that if the number of slabs is large enough, then one of the slabs will contain no more than $K_s = O(\sqrt{P})$ solution centers.

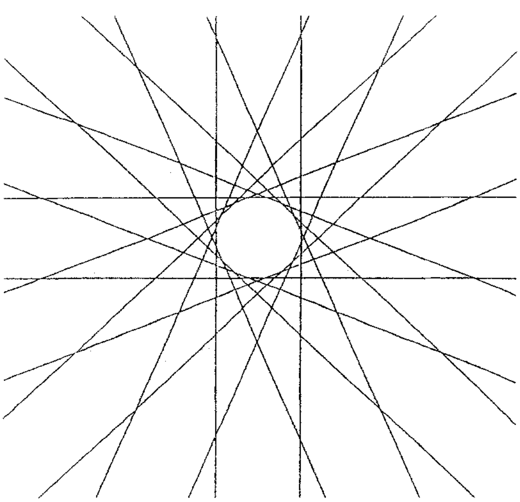


Fig. 2

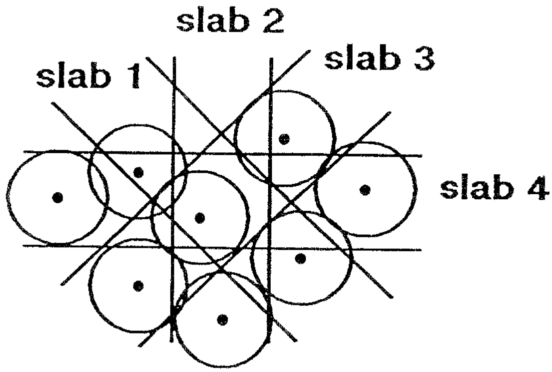


Fig. 3

Let the number of solution centers inside slab i be denoted as g_i and $g = \min_{1 \leq i \leq L}\{g_i\}$. The first question to ask is: How large should L (L is the number of slabs regularly surrounding a reference point) be? Obviously, the larger L is, the smaller g is, because a larger L indicates that we are examining a larger number of slabs. Consequently, we shall not miss any slab in which only a small number of slabs needs be examined. However, it takes time to examine slabs. Therefore, we can hardly afford examining too many slabs. In the rest of this section, we shall show that an upper bound of g is $O(\sqrt{P})$, when L is large enough.

Consider Figure 4. There are many concentric circles. Each circle has radius $i \cdot r$, $i = 1, 2, 3, \ldots$. Let $A(i)$ denote the region between two concentric circles of radius $i \cdot r$ and $(i-1) \cdot r$. Let m_i denote the number of solution centers in $A(i)$. Note that inside $A(i)$, a solution center may be covered by more than one slab. Let x_i denote the solution center which is covered by the largest number of slabs in $A(i)$. Let V_i denote the number of slabs covering x_i.

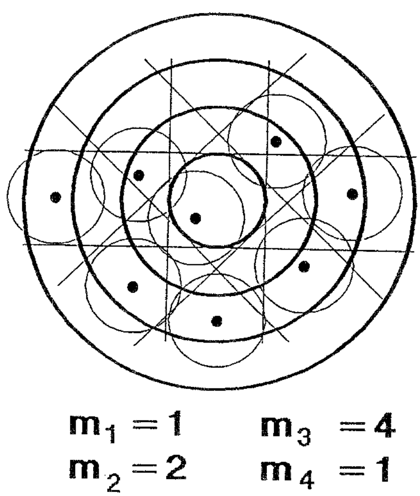


Fig. 4

Let $V(i)$ be an upper bound of V_i, let $U(i)$ be an upper bound of $\sum_{k=1}^{i} m_i$, and let K be a number such that for any $k \geq K$, $U(k) = P$. The rest of this section is organized as follows. In Lemma 1, we show that $g \leq \sum_{i=1}^{K}(U(i) - U(i-1)) \cdot V(i)/L$. In Lemma 2, we show that a proper value of $V(i)$ can be determined. Later values of $U(i)$ and K are determined in Lemma 5. Finally, in Theorem 1, we combine the above results, and conclude that g can be bounded by a number $K_s = O(\sqrt{P})$, when $L = K$.

Now we propose the following lemma.

LEMMA 1.

$$g \leq \sum_{i=1}^{K}(U(i) - U(i-1)) \cdot V(i)/L.$$

PROOF.

$$\begin{aligned} g \cdot L &\leq \sum_{i=1}^{L} g_i \qquad \left(\text{Since } g = \min_{1 \leq i \leq L} \{g_i\}.\right) \\ &\leq \sum_{i=1} m_i V_i. \end{aligned}$$

There must exist a number C, such that for any $i > C$, $m_i = 0$. Therefore

$$\begin{aligned} g \cdot L &\leq \sum_{i=1} m_i V_i = \sum_{i=1}^{C} m_i \cdot V_i \\ &= \sum_{i=1}^{C-1} m_i \cdot V(i) + m_C \cdot V(C) \\ &\leq \sum_{i=1}^{C-1} m_i \cdot V(i) + \left(U(C) - \sum_{i=1}^{C-1} m_i\right) \cdot V(C) \\ &\quad \left(\text{Note that } U(C) \geq \sum_{i=1}^{C} m_i = m_C + \sum_{i=1}^{C-1} m_i.\right) \\ &= \sum_{i=1}^{C-1} m_i \cdot (V(i) - V(C)) + U(C) \cdot V(C) \\ &= \sum_{i=1}^{C-2} m_i \cdot (V(i) - V(C)) + m_{C-1}(V(C-1) - V(C)) + U(C) \cdot V(C) \\ &\leq \sum_{i=1}^{C-2} m_i(V(i) - V(C)) + \left(U(C-1) - \sum_{i=1}^{C-2} m_i\right) \\ &\quad \cdot (V(C-1) - V(C)) + U(C) \cdot V(C) \\ &= \sum_{i=1}^{C-2} m_i(V(i) - V(C) - V(C-1) + V(C)) \\ &\quad + U(C-1) \cdot (V(C-1) - V(C)) + U(C) \cdot V(C) \end{aligned}$$

$$= \sum_{i=1}^{C-2} m_i(V(i) - V(C-1)) + U(C-1)\cdot(V(C-1) - V(C)) + U(C)\cdot V(C)$$

$$\vdots$$

$$\le \sum_{i=1}^{C-3} m_i(V(i) - V(C-2)) + U(C-2)\cdot(V(C-2) - V(C-1))$$
$$+ U(C-1)\cdot(V(C-1) - V(C)) + U(C)\cdot V(C)$$

$$\vdots$$

$$\le \sum_{i=1}^{C-1} U(i)\cdot(V(i) - V(i+1)) + U(C)\cdot V(C).$$

Since $U(i)$ is defined on $i = 1, 2, \ldots$, we may conveniently set $U(0) = 0$. Therefore

$$g\cdot L \le \sum_{i=1}^{C-1} U(i)\cdot V(i) + U(C)\cdot V(C) - \sum_{i=1}^{C-1} U(i)\cdot V(i+1) - U(0)\cdot V(1)$$
$$= \sum_{i=1}^{C} U(i)\cdot V(i) - \sum_{i=0}^{C-1} U(i)\cdot V(i+1)$$
$$= \sum_{i=1}^{C} U(i)\cdot V(i) - \sum_{i=1}^{C} U(i-1)\cdot V(i)$$
$$= \sum_{i=1}^{C} (U(i) - U(i-1))\cdot V(i).$$

Therefore

$$g\cdot L \le \sum_{i=1}^{C} (U(i) - U(i-1))\cdot V(i).$$

If $K \ge C$,

$$g\cdot L \le \sum_{i=1}^{C} (U(i) - U(i-1))\cdot V(i) \le \sum_{i=1}^{K} (U(i) - U(i-1))\cdot V(i).$$

If $K < C$,

$$g\cdot L \le \sum_{i=1}^{C} (U(i) - U(i-1))\cdot V(i)$$
$$= \sum_{i=1}^{K} (U(i) - U(i-1))\cdot V(i) + \sum_{i=K+1}^{C} (U(i) - U(i-1))\cdot V(i)$$
$$= \sum_{i=1}^{K} (U(i) - U(i-1))\cdot V(i).$$

(Since $U(i) = U(i-1) = P$, for any $i - 1 \ge K$.)

Therefore

$$g \leq \sum_{i=1}^{K} (U(i) - U(i-1)) \cdot V(i)/L. \qquad \square$$

In the next lemma, we determine a value of $V(i)$. We first define another term B_j, which is like V_i, except that it is defined at the circle of radius $j \cdot r$ (where j is any positive real number) instead of a range. Therefore we can see that the upper bound of $\max\{B_j | i-1 \leq j \leq i\}$ is also an upper bound of V_i. Now we prove Lemma 2.

LEMMA 2. *If we choose* $V(1) = V(2) = L$, *and* $V(i) = \lceil (2L/\pi) \times (1/(i-2)) \rceil$, *for* $i \geq 3$, *then* $V(k)$ *is an upper bound of* V_k, *for any positive integer* k.

PROOF. It is easy to see that $V_1 \leq L$ and $V_2 \leq L$. Therefore we only consider the case when $i \geq 3$. Let y_j be a point on the boundary of a circle of radius $j \cdot r$ and covered by the largest number of slabs. Let B_j be the number of slabs covering y_j. (Note that j is any positive number, and i is a positive integer.) Therefore, from the definitions, we know that

$$(1) \qquad V_i = \max\{B_j | i-1 \leq j \leq i\}.$$

Draw a line through y_j and the origin. Let θ be the largest angle as shown in Figure 5, such that any slab which has width $2r$ and whose central line lies inside θ will cover y_j. It is obvious that $0 < \theta/2 < \pi/2$, when $j \geq 3$.

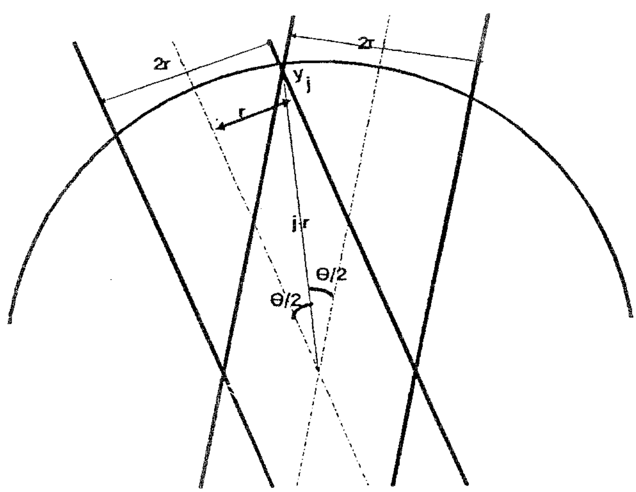


Fig. 5

An upper bound of θ can be found in the following derivations (when $j \geq 3$):

$$\tan(\theta/2) = \frac{r}{\sqrt{(j\cdot r)^2 - r^2}} = \frac{1}{\sqrt{j^2 - 1}}.$$

It is well known that

$$\tan(x) = x + \frac{x^3}{3} + \frac{2x^5}{15} + \frac{17x^7}{315} + \cdots + \frac{2^{2m}\cdot(2^{2m}-1)\cdot B_m x^{2m-1}}{(2m)!} + \cdots,$$

$-\pi/2 < x < \pi/2$, where B_m is the Bernoulli number of m

$$\tan(x) - x = \frac{x^3}{3} + \frac{2x^5}{15} + \frac{17x^7}{315} + \cdots + \frac{2^{2m}\cdot(2^{2m}-1)\cdot B_m x^{2m-1}}{(2m)!} + \cdots.$$

Since $B_m \geq 0$, we know that $\tan(x) - x > 0$, when $-\pi/2 < x < \pi/2$.
In our case, $0 < \theta/2 < \pi/2$. Therefore $\theta/2 < \tan(\theta/2)$

$$\begin{aligned}\theta/2 &< \tan(\theta/2) < \frac{1}{\sqrt{j^2-1}}\\ &= \frac{1}{\sqrt{(j-1)\cdot(j+1)}} < \frac{1}{\sqrt{(j-1)^2}} < \frac{1}{j-1},\\ \theta &< \frac{2}{j-1}.\end{aligned}$$

An upper bound of θ has been found. Then we ask another question: How many slabs among the L slabs can be put into angle θ? Because there are only L slabs, the angle between any two consecutive slabs is π/L. Therefore the largest number of slabs which can be put into angle θ is

$$B_j \leq \left\lceil \frac{\theta}{(\pi/L)} \right\rceil < \lceil (2L/\pi)\cdot(1/(j-1)) \rceil. \tag{2}$$

From (1) and (2), we derive that $V_i = B_{i-1}$. Therefore,

$$V_i = B_{i-1} < \lceil (2L/\pi)\cdot(1/(i-2)) \rceil,$$

when $i \geq 3$. □

In the next step, we shall derive a value of $U(i)$. The way to determine $U(i)$ is somewhat complicated. We first present a lemma to show that if a large circle of radius $(k+1)\cdot r$ can be fully covered by i small circles of radius r (that means any area in the large circle is covered by at least one small circle), there are at most i solution centers in the area of a circle of radius $k \cdot r$.

LEMMA 3. *Let C_1 and C_2 be two concentric circles of radii $k \cdot r$ and $(k + 1) \cdot r$, respectively. If C_2 can be fully covered by i circles of radius r, then there are at most i solution centers of the optimal solution in C_1*

PROOF. First, we assume that there are j solution centers of the optimal solution in C_1, where $j > i$. We now show that these j solution centers cannot constitute an optimal solution.

The area which is covered by the circles centered at the solution centers inside C_1 (radius $k \cdot r$) is restricted in C_2 (radius $(k + 1) \cdot r$), for the radii of these circles are only r. Therefore those demand points covered by these circles centered at the j solution centers must be distributed inside C_2.

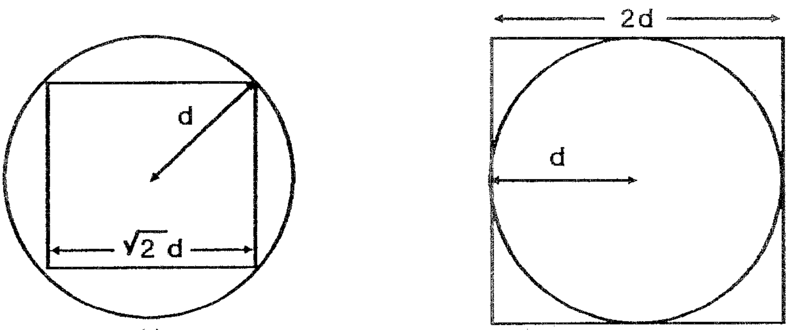
Because i circles are sufficient to cover all the area of C_2, these i circles must also cover all demand points inside this circle. Therefore we can choose the centers of these i circles as the new supply points. This shows that these j solution centers cannot be an optimal solution, because there is a better solution which needs only i $(i < j)$ supply points. □

Now another problem arises. What is the relation between i and k in the above lemma? To find the relation directly is difficult. Therefore we use an indirect method to solve this problem. We know that a circle of radius r can fully cover a square of side length $\sqrt{2} \cdot r$. Therefore if an area can be fully covered by i squares of side length $\sqrt{2} \cdot r$, this area can also be covered by i circles of radius r. In the following property and the lemma, we discuss the relation between the squares and the circle.

PROPERTY A. A circle of radius r can cover a square of side length $\sqrt{2} \cdot r$, and a square of side length $2 \cdot r$ can cover a circle of radius r.

Figure 6 illustrates this property.

Now we use the above property to derive the next lemma.



(a) (b)

Fig. 6

LEMMA 4. *A circle of radius $k \cdot r$ can be covered by $(\sqrt{2}k + 1)^2$ squares of side length $\sqrt{2} \cdot r$.*

PROOF. From property A, we know that a circle of radius $k \cdot r$ can be covered by a square of side length $2 \cdot k \cdot r$. Because $(2 \cdot k \cdot r)/(\sqrt{2} \cdot r) = \sqrt{2} \cdot k$, $\lceil \sqrt{2} \cdot k \rceil^2$ squares of side length $\sqrt{2} \cdot r$ are sufficient to cover a square of side length $2 \cdot k \cdot r$. Therefore $(\sqrt{2}k + 1)^2 \geq \lceil \sqrt{2} \cdot k \rceil^2$ squares of side length $\sqrt{2} \cdot r$ can cover a circle of radius $k \cdot r$. □

Now we want to derive a value of $U(i)$. From Lemma 3, we know that if we want to find an upper bound of solution centers in $A(k)$, we should first calculate the number of circles which can cover a circle of radius $(k + 1) \cdot r$. Since Lemma 4 shows that $(\sqrt{2}k + 1)^2$ squares of side length $\sqrt{2} \cdot r$ are sufficient to cover a circle of radius $k \cdot r$, we conclude that there must exist $(\sqrt{2}k + 1)^2$ circles of radius r which can cover this circle, for a circle of radius r can cover a square of side length $\sqrt{2} \cdot r$. Therefore we can now determine the value of $U(i)$.

LEMMA 5. *If we choose*

$$U(k) = \min\{(\sqrt{2} \cdot k + \sqrt{2} + 1)^2, P\}$$

and

$$K = \lceil \sqrt{P/2} - (1 + 1/\sqrt{2}) \rceil,$$

then $U(k)$ is an upper bound of $\sum_{i=1}^{k} m_k$, where k is any positive integer, and for any $i \geq K$, $U(i) = P$.

PROOF. We know that, for Lemma 4, $(\sqrt{2}k + 1)^2$ squares of side length $\sqrt{2} \cdot r$ can cover a circle of radius $k \cdot r$, and a circle of radius r can cover a square of side length $\sqrt{2} \cdot r$. Therefore $(\sqrt{2}k + 1)^2$ circles of radius r must be able to cover a circle of radius $k \cdot r$, and $(\sqrt{2}(k + 1) + 1)^2 = (\sqrt{2} \cdot k + \sqrt{2} + 1)^2$ circles of radius r must be able to cover a circle of radius $(k + 1) \cdot r$.

Considering Lemma 3 and the above result, we conclude that there are at most $(\sqrt{2} \cdot k + \sqrt{2} + 1)^2$ solution centers in the circle of radius $k \cdot r$. Since the total number of solution centers in the optimal solution is P,

$$\sum_{i=1}^{k} m_k \leq \min\{(\sqrt{2} \cdot k + \sqrt{2} + 1)^2, P\}.$$

Therefore if $U(k) = \min\{(\sqrt{2} \cdot k + \sqrt{2} + 1)^2, P\}$, then $\sum_{i=1}^{k} m_k \leq U(k)$. Since $U(k)$ is known, it is easy to derive that $K = \lceil \sqrt{P/2} - (1 + 1/\sqrt{2}) \rceil$. □

Now we have already found the values of $U(i)$, $V(i)$, and K. At the beginning of this section, we have defined L, which is the number of slabs which we should examine. In the next theorem, we combine all the above results together and show that when $L = K$ we can derive that an upper bound of g is $O(\sqrt{P})$.

THEOREM 1. *If $L = K$, an upper bound of g is $K_s = O(\sqrt{P})$.*

PROOF.

$$g \le \left(\sum_{i=1}^{K} (U(i) - U(i-1)) \cdot V(i) \right) \Big/ L \qquad \text{(from Lemma 1)}$$

$$= \left(\sum_{i=3}^{K} (U(i) - U(i-1)) \cdot V(i) \right) \Big/ L + (U(2) - U(1)) \cdot V(2)/L + (U(1) - U(0)) \cdot V(1)/L$$

$$\le \left(\sum_{i=3}^{K} (U(i) - U(i-1)) \cdot V(i) \right) \Big/ L + U(2) \qquad (\text{for } U(0) = 0 \text{ and } V(1) = V(2) = L)$$

$$\le \sum_{i=3}^{K} ((\sqrt{2} \cdot i + \sqrt{2} + 1)^2 - (\sqrt{2} \cdot i + 1)^2) \cdot V(i)/L + U(2)$$

$$= \sum_{i=3}^{K} (4 \cdot i + 2 + 2\sqrt{2}) \cdot \left(\left\lceil \frac{2L}{\pi(i-2)} \right\rceil \right) \Big/ L + U(2)$$

$$\le \sum_{i=3}^{K} (4 \cdot i + 2 + 2\sqrt{2}) \cdot \left(\frac{2L}{\pi(i-2)} + 1 \right) \Big/ L + U(2)$$

$$= \sum_{i=3}^{K} (4 \cdot i + 2 + 2\sqrt{2}) \cdot \left(\frac{2}{\pi(i-2)} + \frac{1}{L} \right) + U(2).$$

Let $j = i - 2$

$$g \le \sum_{j=1}^{K-2} (4 \cdot j + 10 + 2\sqrt{2}) \cdot \left(\frac{2}{\pi \cdot j} + \frac{1}{L} \right) + U(2)$$

$$\le \sum_{j=1}^{K-2} (4 \cdot j + 10 + 2\sqrt{2}) \left(\frac{2}{\pi \cdot j} \right) + \sum_{j=1}^{K-2} (4 \cdot j + 10 + 2\sqrt{2}) \left(\frac{1}{L} \right) + U(2)$$

$$\le \sum_{j=1}^{K-2} (4 \cdot j + 10 + 2\sqrt{2}) \left(\frac{2}{\pi} \right) + \sum_{j=1}^{K-2} (4 \cdot j + 10 + 2\sqrt{2}) \left(\frac{1}{L} \right) + U(2)$$

$$\le (K-2) \cdot (8 + 20 + 4\sqrt{2}) \left(\frac{1}{\pi} \right) + \sum_{j=1}^{K-2} \left((4 \cdot j/L) + (K-2) \cdot (10 + 2\sqrt{2}) \cdot \left(\frac{1}{L} \right) \right)$$

$$+ U(2).$$

From Lemma 5, $U(2) = 19 + 6\sqrt{2}$. Let $c1 = U(2)$, $c2 = (8 + 20 + 4\sqrt{2})(1/\pi)$,

$c3 = (10 + 2\sqrt{2})$, and $L = K$. Then we derive that

$$g \leq (K-2)\cdot c2 + \sum_{j=1}^{K-2}\left((4\cdot j/K) + (K-2)\cdot c3\cdot\left(\frac{1}{K}\right)\right) + c1,$$

$$g \leq (K-2)\cdot c2 + (K-2)\cdot c3\cdot\left(\frac{1}{K}\right) + 4\cdot(K-2)\cdot(K-1)/2K + c1.$$

From the above equation, we can see that when $L = K$, there is an upper bound of g which is $O(K)$.

Let

$$K_s = (K-2)\cdot c2 + (K-2)\cdot c3\cdot\left(\frac{1}{K}\right) + 4\cdot(K-2)\cdot(K-1)/2K + c1.$$

Since $K = \lceil\sqrt{P/2} - (1 + 1/\sqrt{2})\rceil$, we conclude that $g \leq K_s = O(K) = O(\sqrt{P})$. □

Now we have shown the first property of the dividing slab. In the next section, we will show the second property of the dividing slab.

6. The Second Property of the Dividing Slab. In the above section, we deliberately avoided discussing the problem of determining a proper location of the L slabs. We shall now proceed to discuss the problem. Consider the case when $L = 1$, as shown in Figure 7. Assume that an optimal solution of some (P, r) CC problem instance is known. We want to find a position to put this slab, such that the number of solution centers to the left and to the right of the slab are both no more than $\lceil 2P/3\rceil$.

To achieve this, we observe that because of the width, the circles of radius r centered to the left of the central line of the slab cannot cover the demand points to the right of the slab, and vice versa. Therefore, if the number of solution centers to the left of the central line of the slabs is not less than $\lfloor P/3\rfloor$, then the number of solution centers to the right of the slab must be no more than $\lceil 2P/3\rceil$, for the total number of solution centers is no more than P, as shown in Figure 8.

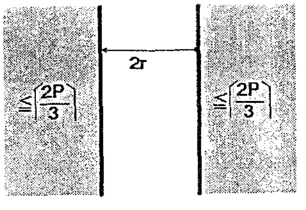


Fig. 7

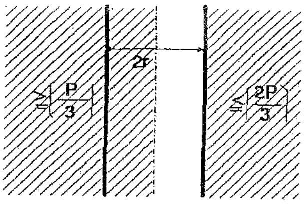


Fig. 8

To find the position of a central line, we may conduct a linear scan and recursively call Procedure CIRCLE_COVER (see Section 3) to determine whether a set of points can be covered by $\lfloor P/3 \rfloor$ circles with radius r. Consider Figure 9.

We may conduct the linear scanning from both the leftmost and the rightmost direcctions toward their opposite directions, and ask whether the set of demand points to the left of the left scan line and to the right of the right scan line can be covered by exactly $\lfloor P/3 \rfloor$ circles with radius r. Then we can find two scan lines, as shown in Figure 10. We call the middle empty area the gap. It can be easily seen that if we place a slab centered at any position of the gap, we obtain a slab such that the numbers of solution centers to the left and to the right of this slab are both no more than $\lceil 2P/3 \rceil$, as shown in Figure 10.

In the above discussion we assumed that the direction of the slab is determined; we only have to find the proper location. It is easy to see that as long as the direction of the slab does not coincide with the direction of a line linking two demand points, this location can always be found. If the direction does coincide with the direction of a line linking two demand points, we may tilt this direction slightly to overcome the trouble.

Let us now consider the case when $L = 2$. Using similar reasoning techniques, we can determine two gaps perpendicular to each other as shown in Figure 11. There exists a common intersection and we can place the two slabs centered at this intersection area, as shown in Figure 12. Then our problem is solved.

For any number of L, we can perform the same operations. But we can see that there is no guarantee that the gaps between all pairs of scan lines will intersect at a common area. It is interesting that we can use Helly's theorem (Edelsbrunner, 1987) to solve this problem. Helly's theorem is presented as follows:

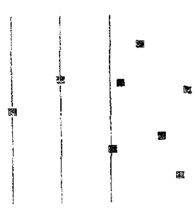


Fig. 9

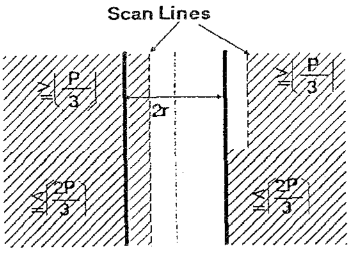


Fig. 10

THEOREM 2 (Helly's Theorem) (a simplified version) (Edelsbrunner, 1987; Helly, 1923). *If there are n convex sets in a plane, and any three of them have a common intersection, then all n convex sets have a common intersection.*

Because a half-plane is also a convex set, Helly's theorem can also be stated as follows:

If n half-planes do not have a common intersection, then there exist three half-planes such that they have no common intersection.

Now let us see how we can apply Helly's theorem. As shown in Figure 11, the intersection of the two gaps can be expressed by the following formula:

$$(h_1 \cup h_2 \cup h_3 \cup h_4)^c,$$

where h_1, h_2, h_3, and h_4 are half-planes and h_i^c denotes the complement of h_i.

In general, we are given $2 \cdot L$ half-planes $h_1, h_2, \ldots, h_{2 \cdot L}$ and we are interested in knowing whether $(h_1 \cup h_2 \cup \cdots \cup h_{2 \cdot L})^c$ is empty or not. We shall prove the following theorem. (Let $\mathbb{R}^2$ denote the entire plane.)

THEOREM 3. *If* $(h_1 \cup h_2 \cup \cdots \cup h_{2 \cdot L})^c = \varnothing$, *then there exist three half-planes* h_{i1}, h_{i2}, $h_{i3} \in \{h_1, h_2, \ldots, h_{2 \cdot L}\}$, *such that* $(h_{i1} \cup h_{i2} \cup h_{i3}) = \mathbb{R}^2$.

PROOF. Since $(h_1 \cup h_2 \cup \cdots \cup h_{2 \cdot L})^c = \varnothing$, we have $(h_1^c \cap h_2^c \cap \cdots \cap h_{2 \cdot L}^c) = \varnothing$. According to Helly's theorem, there exist three half-planes, h_{i1}, h_{i2},

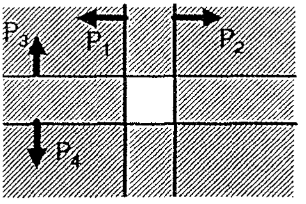


Fig. 11

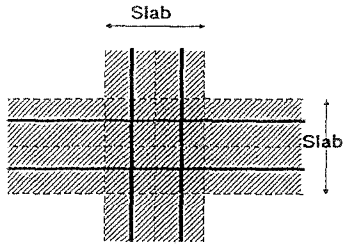


Fig. 12

$h_{i3} \in \{h_1, h_2, \ldots, h_{2 \cdot L}\}$, such that $(h_{i1}^c \cap h_{i2}^c \cap h_{i3}^c) = \varnothing$. This implies that $(h_{i1} \cup h_{i2} \cup h_{i3}) = \mathbb{R}^2$. □

Theorem 3 shows that if L gaps do not have a common intersection, then there must exist three half-planes such that they comprise the entire plane. But each half-plane defines a region in which the demand points can be covered by $\lfloor P/3 \rfloor$ circles. This means that the entire set of demand points can be covered by at most P circles with radius r. We can find these circles by adding up all circles found in these three half-planes by calling Procedure CIRCLE_COVER.

We may conclude that either we have a reference location to put all the L slabs, such that each slab satisfies the second property of the dividing slab, or we have found no more than P circles which can cover all demand points in D. The detailed algorithm about how to draw these slabs and find the reference location is described in Procedure GEN_SLABS discussed in the next section.

7. Procedure GEN_SLABS and Its Relative Complexities. We now present Procedure GEN_SLABS, which corresponds to the method described in the above two sections. In this procedure, we generate a set of slabs and one of them is the dividing slab. Or we will find no more than P circles with radius r which can cover all the demand points in D.

Procedure GEN_SLABS(D, P, r, L_s, or S)

Input: A set D of n demand points, a number P, and a radius r.

Output: Return: A set L_s of slabs and one of them is the dividing slab, or K' circles of radius r, $K' \leq P$, which can cover all points in D.

Step 1. Draw a set L_s of L slabs regularly surrounding a point in the plane.

Step 2. Rotate the L slabs, such that no lines connecting any two demand points are parallel to any slab. (First, we choose any direction as the L slabs direction and test whether there exists any line, which connects any two demand points, parallel to any slab direction or not. If such a line exists, we then find all angles between slabs and lines. Let the smallest nonzero angle be δ. It is obvious that if we rotate all the slabs by angle $\delta/2$, there will exist no line connecting any two demand points parallel to any slab direction.)

Step 3. For each slab l_i in L_s do:

Step 4. Find a line L'_i which is perpendicular to l_i.
Step 5. Sort all demand points in D by the sequence that they map into L'_i from left to right, denoted as $d'_1, d'_2, \ldots, d'_n$.
Step 6. For $j := 1$ to n do:
Step 7. Let $m := 2 \cdot L$.
Step 8. $T :=$ CIRCLE_COVER($\{d'_1, \ldots, d'_j\}, \lfloor P/3 \rfloor, r, S_{m-1}$).
Step 9. If $T =$ "FALSE," then $j := j - 1$ and go to next step.
Step 10. For $k := 1$ to n do:
Step 11 $T :=$ CIRCLE_COVER($\{d'_n, d'_{n-1}, \ldots, d'_k\}, \lfloor P/3 \rfloor, r, S_m$).
Step 12. If $T =$ "FALSE," then $k := k + 1$ and go to next step.
Step 13. Draw the line l''_{m-1} (resp. l''_m) which is parallel to l_i and passes through the point d'_j (resp. d'_k).
Step 14. Let h_{m-1} (resp. h_m) be the half-plane to the left (resp. right) of l''_{m-1} (resp. l''_m) including the line l''_{m-1} (resp. l''_m).
Step 15. If $(h_1 \cup h_2 \cup h_3 \cup \cdots \cup h_{2 \cdot L}) = \mathbb{R}^2$, then do:
Step 16. Find $m1$, $m2$, and $m3$, such that $(h_{m1} \cup h_{m2} \cup h_{m3}) = \mathbb{R}^2$.
Step 17. Return $S = S_{m1} \cup S_{m2} \cup S_{m3}$ as a solution of the (P, r) CC problem.
Step 18. Else do:
Step 19. Find a point p not covered by any of $h_1, h_2, \ldots, h_{2 \cdot L}$.
Step 20. Move the set L_s of L slabs, such that their central lines intersect at point p. Return L_s.

The time complexities of the above steps are as follows: The time complexity needed in steps 1 and 2 is $O(n^2 \cdot L)$. In Steps 3–14, it takes $O(n \cdot L \cdot T(\lfloor P/3 \rfloor))$, where $T(\lfloor P/3 \rfloor)$ is the time needed by recursively calling Procedure CIRCLE_COVER. In Steps 15–17, it takes $O(L^3 \cdot n)$ and in Steps 18–20, it takes $O(L)$ time. Therefore $T_1(P)$ is $O(n^2 \cdot L + n \cdot L \cdot T(\lfloor P/3 \rfloor) + L^3 \cdot n) = O(T(\lfloor P/3 \rfloor) + n \cdot P^{3/2})$.

Recall that, for the EPC problem (also the PCC problem), we perform the binary search on all possible radii before calling Procedure CIRCLE_COVER. Therefore the total time complexity of the corresponding algorithm for the EPC problem (also the PCC problem) is $O(T(P)) \cdot O(\log n) = O(n^{O(\sqrt{P})})$.

8. Conclusions. In this paper, we propose the slab-dividing method solving the Euclidean P-Center problem in time $O(n^{O(\sqrt{P})})$. Lipton and Tarjan (1979, 1980) and Mehlhorn (1984) also proposed an algorithm which solved some planar NP-hard problems in time $O(n^{O(\sqrt{n})})$. We believe that there are still many famous NP-hard problems defined on the planar graph or the geometry plane which can be solved more efficiently as these cases.

References

Aho, A. V., Hopcroft, J. E., and Ullman, J. D., *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

Bentley, J. L., Multidimensional divide and conquer, *Comm. ACM*, Vol. 23, No. 4, 1980, pp. 214–229.

Drezner, Z., On a modified one-center model, *Management Sci.*, Vol. 27, 1981, pp. 848–851.

Drezner, Z., The *P*-center problem.—Heuristics and optimal algorithms, *J. Oper. Res. Soc.*, Vol. 35, No. 8, 1984, pp. 741–748.

Drezner, Z., On the rectangular *P*-center problem, *Naval Res. Logist. Quart.*, Vol. 34, 1987, pp. 229–234.

Edelsbrunner, H., *Algorithms in Combinatorial Geometry*, Springer-Verlag, New York, 1987.

Helly, E., Über Mengen konvexer Körper mit gemeinschaftlichen Punkten, *Jahresber. Deutsch. Math.-Verein.*, Vol. 32, 1923, pp. 175–176.

Horowitz, E. and Sahni, S., *Fundamentals of Computer Algorithms*, Computer Science Press, Rockville, MD, 1978.

Lipton, R. and Tarjan, R. E., A separator theorem for planar graphs, *SIAM J. Appl. Math.*, Vol. 36, No. 2, 1979, pp. 177–189.

Lipton, R. and Tarjan, R. E., Applications of a planar separator theorem, *SIAM J. Comput.*, Vol. 9, No. 3, 1980, pp. 615–627.

Megiddo, N., Linear-Time algorithms for linear programming in R^3 and related problems, *SIAM J. Comput.*, Vol. 12, No. 4, 1983, pp. 759–776.

Megiddo, N. and Supowit, K. J., On the complexity of some common geometric location problems, *SIAM J. Comput.*, Vol. 13, No. 1, 1984, pp. 1182–1196.

Mehlhorn, K., *Data Structures and Algorithms 2: Graph Algorithms and NP-Completeness*, Springer-Verlag, Berlin, 1984.