Variable-Size Rectangle Covering

Francis Y.L. Chin⋆, Hing-Fung Ting⋆⋆ and Yong Zhang

Department of Computer Science, The University of Hong Kong, Hong Kong
{chin, hfting, yzhang}@cs.hku.hk

Abstract. In wireless communication networks, optimal use of the directional antenna is very important. The *directional antenna coverage (DAC) problem* is to cover all clients with the smallest number of directional antennas. In this paper, we consider the *variable-size rectangle covering (VSRC) problem*, which is a transformation of the DAC problem. There are n points above the base line $y = 0$ of the two-dimensional plane. The target is to cover all these points by minimum number of rectangles, such that the dimension of each rectangle is not fixed but the area is at most 1, and the bottom edge of each rectangle is on the base line $y = 0$. In some applications, the number of rectangles covering any position in the two-dimensional plane is bounded, so we also consider the variation when each position in the plane is covered by no more than two rectangles. We give two polynomial time algorithms for finding the optimal covering. Further, we propose two 2-approximation algorithms that use less running time ($O(n \log n)$ and $O(n)$).

1 Introduction

Let R be a region above the *base*, i.e., $y = 0$, of a two-dimensional plane. An h-rectangle is a rectangle with its lower edge touching the base, and with its height $h > 0$ and width w, such that $w \cdot h \leq 1$, i.e., the area of an h-rectangle is bounded by 1. An *h-rectangle (h, x^ℓ, x^r)* is defined by its height, the position of its left edge and right edge. We say *an h-rectangle is at q* if its left edge is at $x^\ell = q$.

The *Variable-Size Rectangle Covering (VSRC) problem* is to cover a given set of points with the minimum number of h-rectangles. Note that this VSRC problem differs from the traditional set covering problem in several ways. Besides the points in a two-dimensional region to be covered by h-rectangles, the dimensions of the h-rectangles can vary and the lower sides of the h-rectangles have to be grounded (i.e., touching the base $y = 0$). For example, consider the set of points $P = \{(0, 0.05), (3, 0.3), (4, 0.35), (5, 0.45), (6, 0.1), (7, 0.12), (8, 0.09)\}$. P can be covered by the following three h-rectangles (as shown in Figure 1(a)): $(0.3, 0, 3)$, $(0.45, 4, 6)$ and $(0.12, 7, 8)$, or alternatively, by two h-rectangles (as shown in Figure 1(b)): $(0.12, 0, 8)$ and $(0.45, 3, 5)$. The formulation of the problem has some immediate applications; e.g., points can be assumed as dirty stains on a
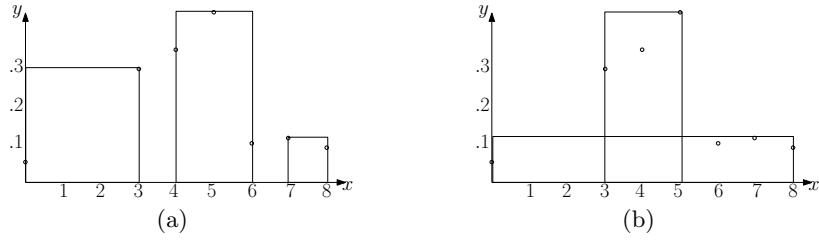
Fig. 1. Two coverings for $P = \{(0, 0.05), (3, 0.3), (4, 0.35), (5, 0.45), (6, 0.1), (7, 0.12),$ $(8, 0.09)\}$

wall, the VSRC problem is equivalent to covering these dirty stains with the minimum number of rectangular cardboards standing on the floor and leaning against the wall, and whose dimensions are not fixed but the area of each cardboard is bounded.

An important application of the VSRC problem is to optimize the use of a directional antenna in a wireless network [11–13]. Traditionally, a wireless network uses an antenna that is *omnidirectional*, i.e., the signals are sent and received in all directions. In recent years, the use of directional antennas has become more common. A *directional antenna* is one whose signal is concentrated in a certain direction. Comparing with omnidirectional ones, it is far more efficient in terms of the frequency bandwidth and energy. In some antenna designs, multiple beams pointing at different directions can be used simultaneously, e.g., the *multi-beam adaptive array* (MBAA) [1]. For example, consider a set of clients on a two-dimensional plane. There is a base station which provides wireless coverage to these clients. The base station uses a directional antenna to send "beams" to cover the clients so as to provide wireless coverage (as shown in Figure 2).
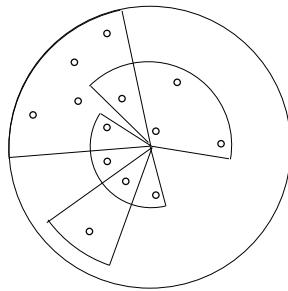

Fig. 2. Clients covered by beams from directional antennas

In an abstract model, the coverage area of a beam can be represented by a circular sector with angle θ and radius r. Typically the covered area of a beam

is bounded since the transmission energy of each base station is bounded, which means the wider the beam angle, the shorter the range [2]. For efficiency, antennas may dynamically adjust their orientation, and/or beam angle (and hence the range) [9]. The *directional antennas coverage (DAC) problem* is to cover all clients with the smallest number of beams.

The DAC problem can be transformed to the VSRC problem as follows. Each client at position (r_i, θ_i) can be converted to point $p_i = (x_i, y_i)$, where $x_i = k\theta_i$ and $y_i = r_i$. Similarly each beam with radius r and angle $\theta(r)$ at a direction θ_0 is equivalent to an h-rectangle (h, x^ℓ, x^r), where $h = r$, $x^\ell = k\theta_0$ and $x^r = k(\theta_0 + \theta(r))$. The bounded area of each beam can be converted to the bounded area of each h-rectangle. Furthermore, since the beam is sent by an antenna at the origin which corresponds to $r = 0$, this would imply that each h-rectangle would have to touch the base $y = 0$ when capturing the points (clients). Let the *wrapped around line* for the VSRC problem be the line $x = 2k\pi$. In this way, the DAC problem can be reduced to the VSRC problem with the points in the region being wrapped around and minimizing the number of beams used would be equivalent to minimizing the number of h-rectangles. From Lemma 1 in Section 2.1, we have the good property of optimal covering, i.e., any two h-rectangles are either disjoint or nested. If we have an algorithm for the VSRC problem without wrapped around point, we can modify the algorithm to deal with wrapped around case (DAC) as follows:

– For each point, cut the region at its x-coordinate then attach the right part to the left of the left part (as shown in Figure 3) and apply the algorithm for finding the optimal covering;
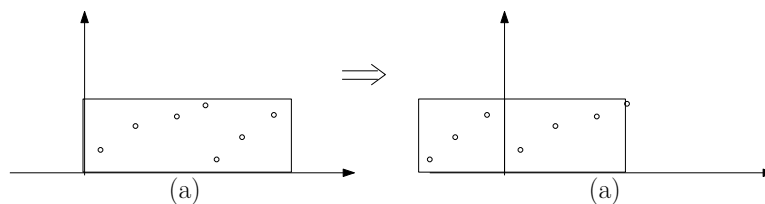
Fig. 3. Cut at the 4th point then attach the right part to the left of the left part

– Return the one with minimal number of h-rectangles.

Thus, we can reduce the algorithm for the VSRC problem without wrapped around point for the DAC problem. On the other hand, given the algorithm for the DAC problem, we can apply it for the VSRC problem without wrapped around point as follows:

– Find the point of the lowest height h;

– Apply the algorithm for VSRC with wrapped around points (DAC) and the wrapped around line is at $x = x_r + 1/h + 1$, where x_r is the x-coordinate of the rightmost point.

In this way, any two points cannot be covered by an h-rectangle crossing the wrapped around line. Thus, the algorithm for the DAC problem can be reduced to the algorithm for the VSRC problem without wrapped around point.

In most applications on wireless communications, to avoid interference, each position in the plane can be covered by a bounded number of antennas. In other applications, this constraint is still reasonable, e.g., in the above example of covering dirty stains on the wall, the number of cardboards covering each position on the wall cannot be too many, that is because the total thickness of cardboards on each point cannot be too thick. In this paper, we consider the variation that any position in two-dimensional plane can be covered by at most two h-rectangles.

The problem of covering a set of points in a two-dimensional plane with the minimum number of unit disks or squares is well studied. It is NP-hard [5] to find the optimal solution, and polynomial time approximation schemes are known [7] (even for any higher but fixed dimensions). Berman et al. [2] used bin packing [8, 10] and other techniques to give a 3-approximation algorithm for the capacitated version of DAC problem, i.e., the number of clients covered by a beam is bounded. They also considered the case where the radii of all antennas are fixed and equal. This essentially reduces the problem to one dimension. They [2] gave a tight 1.5-approximation algorithm for this case.

To find the optimal covering for the VSRC problem, we give two polynomial time algorithms: an $O(n^4)$ time algorithm and an $O(n^3)$ time algorithm for the variations without and with the constraint that each position in the plane is covered by no more than two h-rectangles. With less running time ($O(n \log n)$ and $O(n)$), we propose two 2-approximation algorithms.

In Section 2, we give two optimal algorithms for two variations of the VSRC problem, we also propose two 2-approximation algorithms for this problem by using less running time. The conclusion and future research are discussed in Section 3.

2   Algorithms for the VSRC Problem

In this section, we show that because all h-rectangles have to be grounded, the VSRC problem can be solved in polynomial time by dynamic programming,

2.1 An $O(n^4)$ Algorithm

Firstly, we give a polynomial time algorithm for solving the VSRC problem without the constraint that each position in two-dimensional plane is covered by at most a bounded number of h-rectangles.

Consider any two h-rectangles $r_1 = (h_1, x_1^\ell, x_1^r)$ and $r_2 = (h_2, x_2^\ell, x_2^r)$ in the optimal covering. Assuming $x_1^l \leq x_2^l$, there are at most three cases of the relationship between r_1 and r_2.

- $x_1^l < x_1^r \leq x_2^l < x_2^r$, i.e., these two h-rectangles are *disjoint*, which is shown in Figure 4(a)
- $x_1^l \leq x_2^l < x_1^r \leq x_2^r$, i.e., these two h-rectangles are *overlapped*, as shown in the left part of Figure 4(b). In this case, we may transform these two overlapped h-rectangles to two disjoint h-rectangles, as shown in the right part of Figure 4(b), which still cover the same area.
- $x_1^l \leq x_2^l < x_2^r \leq x_1^r$, i.e., these two h-rectangles are *nested*, as shown in Figure 4(c)

(a)   (b)   (c)
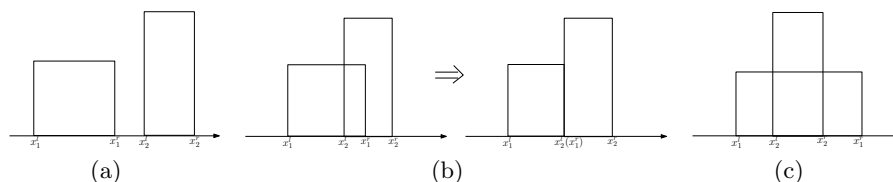
Fig. 4. The relationship between two h-rectangles in the optimal covering.

Thus, we have the following lemma for the optimal covering:

Lemma 1. *In the optimal covering, any two h-rectangles are either disjoint or nested.*

Consider the set of points $P = \{p_1, p_2, ..., p_n\}$, where $p_i = (x_i, y_i)$, and $x_1 \leq x_2 \leq ... \leq x_n$. Define $N(i, j, h)$ be the minimum number of h-rectangles needed to cover the points $Q = \{p_k \in P | i \leq k \leq j \text{ and } y_k > h\}$.

According to Lemma 1, the optimal covering for Q must be in one of these forms (i) disjoint optimal covering for $\{p_i...p_k\} \cap Q$ and $\{p_{k+1}...p_j\} \cap Q$ for some $i \leq k \leq j$; or (ii) nested covering with an h-rectangle R_{ij} with height $h' = 1/(x_j - x_i)$ covering p_i and p_j and some other points in P and with height no more than h', and another optimal covering for $Q' = \{p_k \in Q | i < k < j \text{ and } y_k > h'\}$.

From the above analysis, we have the recursive formula for the minimum number of h-rectangles covering all the points.

$$N(i,j,h) = \begin{cases} 1 & \text{if all points can be covered by one h-rectangle} \\ \min\Big\{ \min_{i \leq k \leq j} \{N(i,k,h)+N(k+1,j,h)\}, N(i,j,h')+1 \Big\} & \text{otherwise} \end{cases}$$

$$(1)$$

As there are $O(n^3)$ terms of $N(i,j,h)$ and from the above formula, it is easy to see that each term takes $O(n)$ time to compute. Thus, we have the following result.

Theorem 1 *The VSRC problem can be solved in $O(n^4)$ time, if each position of the two-dimensional plane can be covered by an unbounded number of h-rectangles.*

This preliminary result shows that the VSRC problem is not NP-hard.

2.2   2-Approximation Algorithms with Less Running Time

In this section, we give two simple algorithms, whose solutions is at most twice of the optimal solution, i.e., 2-approximation. The idea of the first algorithm is from [2].

In any covering strategy, the highest point $p = (x,y)$ must be covered by an h-rectangle with $h \geq y$, and the width of this h-rectangle is at most $1/h$. Note that if $h > y$, the coverage by the h-rectangle higher than y is wasted. Thus, we can assume that $h = y$ and use two disjoint h-rectangles to cover p, on the left and right side of p. After all the covered points are removed, the next highest point will then be selected and processed in the same way until all points are covered. Formally, the algorithm can be described as follows.

Input: $P = \{(x_i, y_i) | 1 \leq i \leq n\}$
Algorithm **VSRC1**

1: **while** $P \neq \phi$ **do**
2:     Find the point (x,y) with the highest height y in P.
3:     Create two h-rectangles $(y, (x-1/y), x)$ and $(y, x, (x+1/y))$
4:     **if** these two h-rectangles overlap with previous h-rectangles **then**
5:         Trim the width of these two h-rectangles until there is no overlap.
6:     **end if**
7:     Remove from P those points covered by these two h-rectangles.
8: **end while**

Theorem 2 *Algorithm VSRC1 is 2-approximation with running time $O(n \log n)$.*

Proof. Let q_i be the highest point selected in round i. We claim that in any covering strategy, any two points q_i and q_j, with $i < j$, must be covered by two disjoint h-rectangles. Otherwise, $|x_i - x_j| \cdot y_i \leq 1$, that means point q_j should be covered and removed in round i, this leads to contradiction.

As the minimum number of h-rectangles needed to cover all the points is no less than the number of highest points selected from each round. Thus, the number of rounds is the lower bound for the optimal covering. Since we create two h-rectangles at each round, VSRC1 is a 2-approximation algorithm.

As updating the set of uncovered points and finding the highest point take $O(\log n)$ time and the number of rounds is at most n, the running time of algorithm VSRC1 is $O(n \log n)$. $\square$

Consider an example with points $p_i = (h, i)$ for $1 \le i \le n$ and $h > 1$. It can be shown easily that VSRC1 outputs $2n$ h-rectangles while the optimal covering needs n h-rectangles.

Can we further improve the running time while having the same approximation ratio? The answer is positive. Algorithm VSRC2 is very simple, just scan all the points from left to right once, and create h-rectangle with area at most 1 to cover the leftmost uncovered points. Formally,

**Input:** $p_i = (x_i, y_i)$ for $1 \le i \le n$ and $x_1 \le x_2 \le ... \le x_n$
Algorithm **VSRC2**

1: $i = 1$
2: **while** $i \le n$ **do**
3:     Find the largest k, s.t. points p_i until p_{i+k} can be covered by one h-rectangle with area no more than 1.
4:     Create an h-rectangle to cover the points $p_i, \cdots, p_{i+k}$
5:     $i = i + k + 1$
6: **end while**

Theorem 3 *Assume $\{p_i\}$ where $x_1 \le x_2 \le ... \le x_n$, Algorithm VSRC2 is 2-approximation with running time $O(n)$.*

Proof. At each round, we create one h-rectangle to cover all those points whose x-coordinates are within an interval, thus, there is no overlap between any two h-rectangles.

Consider the optimal covering OPT, we modify OPT to another covering OPT' without overlapping h-rectangles. For any interval $[l, r]$ covered by more than one h-rectangles in OPT, we use the h-rectangle with the highest height to cover this interval. For example, in the optimal covering, the interval $[l, r]$ is covered by three h-rectangles: (h_1, l_1, r_1), (h_2, l_2, r_2) and (h_3, l, r), where $h_1 \le h_2 \le h_3$, after the modification, the interval $[l, r]$ is only covered by one h-rectangle (h_3, l, r). Note that some h-rectangles with lower height in OPT may be split to several h-rectangles in OPT', as shown in Figure 5. In such modification, if two h-rectangles are nested in the optimal covering, the h-rectangle with lower height will be split to more than one h-rectangles.

Consider the example shown in Figure 5, in the optimal covering as shown in Figure 5(a), there are four h-rectangles 1, 2, 3 and 4, h-rectangle 1 is of the lowest height and overlaps with the other three h-rectangles of higher heights.
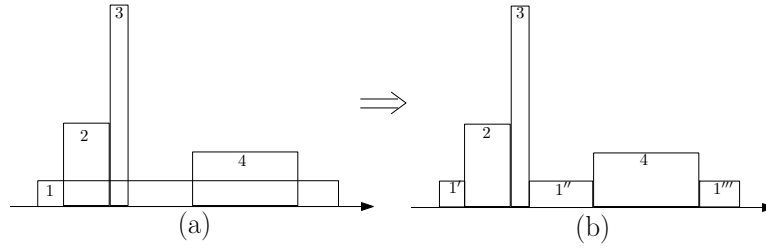
Fig. 5. Example of modify the optimal covering OPT to disjoint covering OPT'

After modification, h-rectangle 1 is split to three h-rectangles, say $1'$, $1''$ and $1'''$, with less widths. Any split occurs from its overlap with an h-rectangle of higher height. We can associate each part, except the leftmost one, of a split to the right edge of the overlapping h-rectangle with higher height. In this example, h-rectangles $1''$ and $1'''$ are associated to the right edges of h-rectangles 3 and 4 respectively. It is easy to see that the right edge of each h-rectangle in OPT can associate with at most one split part in OPT'. Thus, the number of disjoint h-rectangles in OPT' is at most twice to the number of h-rectangles in OPT.

In each round of Algorithm VSRC2, an h-rectangle is created to cover as many points as possible. We can prove by induction from the leftmost point to the rightmost point that the number of h-rectangles created by VSRC2 is no greater than the number of h-rectangles of OPT'. Thus, we can say that VSRC2 is 2-approximation.

At each round, as the points are in order with respect to x_i, finding the largest k such that point p_i until p_{i+k} can be covered by one h-rectangle takes $O(k)$ time. As we can create an h-rectangle in each round in constant time, the total running time of VSRC2 is bounded by $O(n)$. □
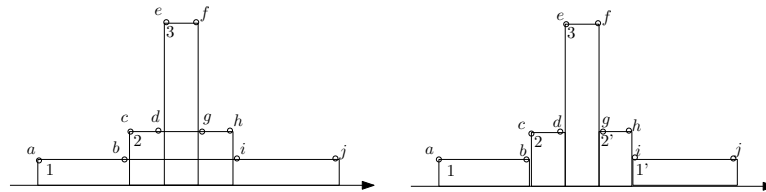
Fig. 6. Tight example of algorithm VSRC2

Consider the example shown in Figure 6, there are 10 points (a to j) to be covered in the plane. The heights of these points are in three levels: a, b, i and j are of the same lowest height; c, d, g and h are of the same middle height; while e and f are of the same highest height. The points of the same height can be covered by an h-rectangle, but any two leftmost points of any two levels

cannot be covered by an h-rectangle. Similarly, any two rightmost points of any two levels cannot be covered by an h-rectangle. For example, points a and c, c and e, f and h, h and j cannot be covered by an h-rectangle. To deal with this example, the optimal covering uses three h-rectangles (1, 2 and 3) while Algorithm VSRC2 uses 5 h-rectangles (1, 2, 3, 2' and 3').

We can generalize this example, force each h-rectangle except the highest one from the optimal covering covers 4 points of the same height, while the highest h-rectangle covers 2 points. When applying VSRC2, all h-rectangles except the one with highest height will be split to two h-rectangles with less width. In this way, we can force the optimal covering uses k h-rectangles while Algorithm VSRC2 uses $2k - 1$ h-rectangles. Thus, the approximation ratio 2 is tight for VSRC2.

2.3 Each Position Covered by at most 2 h-Rectangles

In the previous covering strategy, some positions in two-dimensional plane may be in the overlapped area of more than 2 h-rectangles.

Note that Lemma 1 still holds for the optimal covering if each position can be covered by at most 2 h-rectangles. If two h-rectangles $r_1 = (h_1, x_1^l, x_1^r)$ and $r_2 = (h_2, x_2^l, x_2^r)$ overlap, they must be nested, w.l.o.g., $x_1^l < x_2^l < x_2^r < x_1^r$. We can say that r_2 only overlaps with r_1, otherwise, some position in the plane will be covered by more than two h-rectangles. Therefore, if there are more than one h-rectangles nested in an h-rectangle r, all these h-rectangles except r are disjoint.

Define $N(i, j)$ to be the minimum number of h-rectangles for covering all points between x_i and x_j. Similar to the description in Section 2.1, the optimal covering consists of either two optimal coverings from x_i to x_k and from x_{k+1} to x_j, or an h-rectangle r covering x_i and x_j combined with the optimal covering for those points higher than r. Based on the above analysis, we have the following recurrence formula for the optimal covering of the VSRC problem.

$$N(i, j) = \begin{cases} 1 & \text{if all points can be covered by one h-rectangle} \\ \min \left\{ \min_{i \leq k \leq j} \left\{ N(i, k) + N(k+1, j) \right\}, N^1(i, j) + 1 \right\} & \text{otherwise} \end{cases} \quad (2)$$

where $N^1(i, j)$ is the minimal number of disjoint h-rectangles for covering those points whose x-coordinates are between x_i and x_j, and with their heights higher than $h = 1/(x_j - x_i)$. From the above analysis, as the h-rectangles must be disjoint, we can use Algorithm VSRC2 to find $N^1(i, j)$ as VSRC2 would give the optimal covering if the h-rectangles are not allowed to overlap. Thus, $N^1(i, j)$ can be computed in $O(j-i)$ time, and the computation of $N(i, j)$ can be finished in $O(n)$ time. As there are $O(n^2)$ entries of $N(i, j)$ and $N^1(i, j)$, we have the following theorem.

Theorem 4 *Assume $p_i = (x_i, y_i)$ for $1 \leq i \leq n$ and $x_1 \leq x_2 \leq ... \leq x_n$ and each position can be covered by no more than two h-rectangles, the VSRC problem can be solved in $O(n^3)$ time.*

3 Concluding Remark

We have considered the variable-size rectangle covering problem and proposed several algorithms for finding the optimal covering or approximation covering of some variations of the problem. There are still many covering problems unsolved. In our future research, we will focus on the following directions:

- If the number of points covered by an h-rectangle is bounded and the weight of each point is fractional, it is NP-hard and we can directly use the algorithm from [2] to achieve 3-approximation. But if each point is of unit weight, can we achieve the optimal covering in polynomial time?
- The online version of VSRC problem is a very interesting problem. For one-dimensional case, there are several results on some similar problems [3, 4, 10, 14]. For two-dimensional case, there are many results on covering points by rectangles if the bottom edges of rectangles do not have to be grounded at the base.

References

1. Bao, L., Garia-Luna-Aceves, J.J.: Transmission Scheduling in Ad Hoc Networks with Directional Antennas. In: Proceedings of the 8th International Conference on Mobile computing and networking, pp. 48-58 (2002)
2. Berman, P., Jeong, J., Kasiviswanathan, S., Urgaonkar, B.: Packing to Angles and Sectors. In: Gibbons, P. B., Scheideler, C. (Eds.) SPAA 2007. pp. 171-180 (2007)
3. Charikar, M., Chekuri, C., Feder, T., Motwani, R.: Incremental Clustering and Dynamic Information Retrieval. SIAM J. Comput. 33(6), 1417-1440 (2004)
4. Chan, T.M., Zarrabi-Zadeh, H.: A randomize Algorithm for Online Unit Clustering. In: Erlebach, T., Kaklamanis, C. (Eds.) WAOA 2006. LNCS, vol.4368, pp. 121-131. Springer, Heidelberg (2006)
5. Fowler, R.J., Paterson, M.S., Tanimoto, S.L.: Optimal Packing and Covering in the Plane are NP-Complete. Information Processing Letters 12 (3), 133-137 (1981)
6. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide o the Theory of NP-Completeness. W.H. Freeman (1979)
7. Hochbaum, D.S., Mass, W.: Approximation Scheme for Covering and Packing Problems in Image Processing and VLSI. JACM 32, 130-136 (1985)
8. Karmarkar, N., Karp, R.M.: An Efficient Approximation Scheme for the One-Dimensional Bin-Packing Problem. In: Proceedings of the 23th Annual Symposium on Foundations of Computer Science, pp. 312-320 (1982)
9. Roy, S., Hu, Y., Peroulis D., Li, X.Y.: Minimum-Energy Broadcast Using Practical Directional Antennas in All-Wireless Networks. In: Proc. of the 25th INFORCOM, pp. 1-12 (2006)
10. Seiden, S.: On the Online Bin Packing Problem. JACM 49 (5), 640-671 (2002)
11. Spyropoulos, A., Raghavendra, C.: Energy Efficient Communication in Ad Hac Networks Using Directional Antennas. In: Proc. of the 21th INFORCOM, pp. 220-228 (2002)
12. Voipio V., Vainikainen, P.: Narrowbeam Cylindrical Antenna Array with Sparse Antenna Spacing. In: Proc. of the 48th IEEE Vehicular Technology Conference, vol.1, pp. 465-469 (1998)

13. Yi, S., Pei, Y., Kalyanaraman, S.: On the Capacity Improvement of Ad Hoc Wireless Networks Using Directional Antennas. In: Proceedings of the 9th International Conference on Mobile computing and networking, pp. 108-116 (2003)
14. Zarrabi-Zadeh, H., Chan, T.M.: An Improved Algorithm for Online Unit Clustering. In: Lin, G. (Eds.) COCOON 2007. LNCS, vol. 4598, pp. 383-393. Springer, Heidelberg (2007)