2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

Heterogeneous Sensor Network Deployment Using Circle Packings†

Miu-ling Lam   Yun-hui Liu

Department of Automation and Mechanical Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong

mllam, yhliu@mae.cuhk.edu.hk

Abstract **– This paper addresses the problem of deploying a set of mobile sensor nodes of heterogeneous sensing ranges to give a large and connected coverage. A novel deployment algorithm based on the circle packing technique is given. It iteratively enhances the coverage area of the sensor network from an initial random deployment, while guarantees the absence of coverage hole and obstacle avoidance. The sensing area of each node is modeled as a circular disc while its radius is bounded by the corresponding sensing range limit. The problem of placing these circular discs to cover a field is intuitively transformed to the circle packing problem: given the specified combinatorics of tangency patterns of *n* circles, find the label *R* denoting the radii of these circles. Since a unique packing exists for any given set of triangulations and boundary conditions, we can always find the minimum sensing range required for every interior node to satisfy such packing conditions. Though an extension from tangency packing to overlap packing, the interstices among triples (which represent coverage holes) can be eliminated. We have proven that the maximum global scaling factor to vanish all possible interstices is *α=3^(1/2)/2*. Based on a number of numerical simulations, we have verified that the proposed algorithm always yields sensor deployments of wide coverage and minimize the sensing ranges required for every interior sensing node to satisfy the packing and boundary conditions.**

Index Terms – Circle packing, deployment, mobile sensors, sensing coverage, wireless sensor network.

I. Introduction

Mobility enables a number of important functionality in sensor networks such as coverage maximization, adaptive sampling, network repair, localization and energy harvesting. An effective way to deploy a large set of sensor nodes is important yet difficult. Existing works dealing with sensor node deployment can be generally classified into two categories: physics-based and geometric (Voronoi-based) approaches.

The work in [1] adopts a potential-field-based approach to spread sensor nodes throughout the target environment from a compact initial configuration. However, it does not consider some crucial problems like connectivity maintenance and topology control. The potential-field-based algorithm and the virtual force algorithm (VFA) presented in [2] work in a similar fashion, in that they increase sensor coverage by considering the virtual attractive and repulsive forces exerted on each sensor node by neighbor nodes and/or obstacles (if any). However, these works only consider homogeneous sensing models (i.e. sensors need to have an identical sensing capability), while in this paper, we address the problem of deploying heterogeneous sensor networks. Besides, VFA assumes all sensor nodes are able to communicate with their cluster head which is responsible for calculating sensor movement and the target location.

In [3], three protocols are proposed to enlarge sensor coverage in a target area. The first protocol, namely VEC, is similar to the VFA presented in [2]. The remaining two are based on the Voronoi diagram. In [4], we have presented an *ISOGRID* algorithm for autonomous deployment of mobile sensor networks. The principle is to redeploy the sensor nodes such that the communication graph approximates the layout of an isometric grid. Upon an initial random placement of sensor nodes, the algorithm iteratively computes node movements to enhance sensing coverage and avoid obstacles while ensuring sensor connectivity.

In this paper, the sensing regions are modeled as circular discs of variable sensing ranges. The problem of placing these circular discs on a field is intuitively transformed to the circle packing problem. A circle packing is a configuration of circles with specified patterns of tangency [5]. The central issues of the topic concern connections between the combinatorics of packings and their geometries, the variety among packings sharing combinatorics, computational methods, and connections with analytic function theory and conformal geometry. The study of circle packings was started by William Thurston in his famous notes [6]. Maps between circle packings which preserve tangency and orientation act in many ways as discrete analogues of analytic functions. Moreover, work flowing from a 1985 conjecture of Thurston, proven by [7], shows that classical analytic functions and more general classical conformal objects can be approximated using circle packings.

† This work is supported in part by Hong Kong RGC under grant 414505. It is also affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies.

In this paper, we adopt a circle packing algorithm to solve the sensor network coverage problem. For simplicity, we assume all nodes initially form a connected communication graph. We further assume that the communication graph can be transformed to some triangulations representing the geometric relation among the sensor nodes. Our problem is: given a set of sensors and their maximum sensing ranges, find a deployment and the required sensing range of each associated sensor to give a large and connected coverage region. The central existence result derives from circle packings with certain extremal properties and these packings are called *maximal packings*. Given a set of combinatorics and boundary conditions, the maximal packing is univalent and essentially unique. Therefore, we can always find the minimum sensing range required for every interior node satisfying the boundary and packing conditions. Though an extension from tangency packing to overlap packing, the interstices of triples (which represent coverage holes) can be eliminated. As the circle packing algorithm utilizes only the local information about a sensor node and its neighbors, this module can be executed in a decentralized framework, and thus it is computationally efficient and scalable. Based on a number of simulation experiments, we have verified that the proposed algorithm always yields sensor deployments of wide coverage and desired topologies while setting all interior sensor nodes to their minimum required ranges.

The remaining of this paper is organized as follows. Section II presents the preliminaries and techniques in circle packing. In section III, we will study the extension of tangency packing to overlap packing and apply it to our sensor node deployment problem, such that no coverage hole exists in the resultant deployment. In Section IV, based on a number of simulation results under various situations, we evaluate the performance of our deployment algorithm. Concluding remarks are given in Section V.

II. Circle Packing

In this section, we give an overview of the circle packing problem. A circle packing is a configuration of circles with specified patterns of tangency [5]. We will first give a set of notations and preliminaries of the circle packing problem, and the corresponding physical meanings in our sensor deployment application.

A. Preliminaries and Notations

We will adopt some notations in [5] to make the statements coherent. A hierarchy of circle packing structure consists of several levels of components, namely *circles*, *triples*, *flowers* and *packings*. The coordinates of circle centers refer to sensor node positions and radii refer to the corresponding sensing ranges. Here, all tangencies are referred as the external ones, each circle lying outside the disc bounded by the other. The number of petals defines the *degree k* of the central circle. The condition that every circle has such a flower is a local planarity condition that we will enforce on all our packings.

Triangulation complex K: The tangency patterns for circle packings are encoded as abstract complexes *K*, which represent the triangulations of oriented topological surfaces. *K* is a combinatorial object, with no metric and no geometry. A sensor network topology is generally not a triangulated planar mesh. However, we adopt Delaunay triangulation to define a triangulation of communication graph. It is a simple and well-known triangulation method. The Delaunay triangulation of a discrete point set $\{v\}$ is the geometric dual of the Voronoi tessellation for $\{v\}$. It is the triangulation of the convex hull of $\{v\}$ in which every circumcircle of a triangle is an empty circle. It always gives unique triangulations and maximizes the minimum angles. Compared to any other triangulation of the points, the smallest angle in the Delaunay triangulation is at least as large as the smallest angle in any other. As it is desirable to avoid narrow triangles, we consider it as the best choice among all conventional triangulation approaches. In Section III, we will discuss an *Obtuse-Angle Pruning* to revise the complex *K* by trimming some boundary triangles of the Delaunay triangulation to improve the deployment result.

Circle packing P: A collection $P=\{c_v\}$ of circles is said to be a *circle packing* for a complex *K* if i) *P* has a circle c_v associated with each vertex *v* of *K*, *ii)* two circles c_u, c_v are externally tangent whenever $<u,v>$ is an edge of *K*, and *iii)* three circles c_u, c_v, c_w form a positively oriented triple whenever $<u,v,w>$ forms a positively oriented face of *K*.

Radius label R: *R* is a collection $\{R(v)\}$ of positive numbers associated with vertices *v* of *K*, where $R(v)$ represents the radius of circle c_v. It refers to the assigned values of sensing ranges over the set of sensor nodes. We refer to $K(R)$ as a labeled complex.

Angle sums $\theta_R(v)$: For each triple of radii r_i, r_j and r_k, the Law of Cosines gives the angle α in a corresponding triple of circles, as in Fig. 1.

$$\alpha(r;r_j,r_{j+1}) = \arccos\left[\frac{(r+r_j)^2+(r+r_{j+1})^2-(r_j+r_{j+1})^2}{2(r+r_j)(r+r_{j+1})}\right] \quad (1)$$

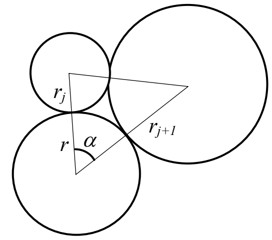


Fig. 1 The Law of Cosines of a triple.

If we add these individual angles over the *k* triples involved, we get the angle sum $\theta_R(v)$ for this label at *v*. Suppose $F_v=\{v;v_1,v_2,\cdots,v_k\}$ is the flower for *v* in *K*. Vertex *v* belongs to *m* faces, where *m=k* if *v* is interior and *m=k-1* if *v* is boundary. In a flower having central label *r* and petal labels $\{r_1,r_2,\cdots,r_k\}$, the angle sum is given by the following summation formula, where *m=k* and $r_{k+1}=r_1$ if the flower is closed, and *m=k-1* otherwise:

$$\theta(r;r_1,r_2,\cdots,r_k)=\sum_{j=1}^{m}\alpha(r;r_j,r_{j+1}) \tag{2}$$

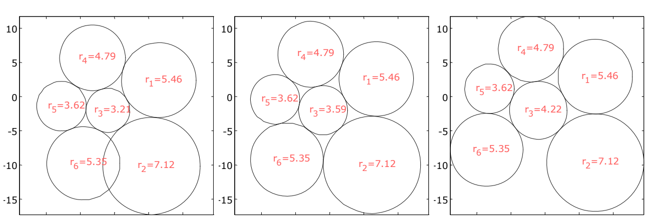
Packing condition: The flower of an interior vertex v can be realized as an actual geometric flower of circles with labels from R if and only if $\theta_R(v)=2\pi n_p$ for some integer $n_p \geq 1$. The integer n is the number of times the petals wrap around the center circle, so it must be 1 for local univalence, i.e. the faces formed by the flower of any interior circle are nonoverlapping. As for sensor coverage application, we are only interested to the univalence case. The branched cases ($n_p > 1$) will not be covered in this paper, yet it has a potential application on *k-coverage* problem. A label R is termed a *packing label* for K if the angle sum $\theta_R(v)$ equals to 2π for every interior vertex v.

If K triangulates a simply connected surface, then a label R for K represents the radii for a circle packing P of K if and only if R is a packing label. [5] has done comprehensive studies of the circle packing problem. The existence and uniqueness of a maximal circle packing rely on a theorem given in [5]. Moreover, the packings we intend to compute are guaranteed by the fundamental existence and uniqueness result: given a set of fixed radii of all boundary vertices of K, there exists a unique circle packing (and the corresponding label R) such that the angle sum θ equals to 2π for every interior vertex of K. Then, we can define the circle packing problem as follows:

PROBLEM DEFINITION *Circle packing problem*: Given a complex K and the radii of all boundary vertices, compute the radii of interior vertices of the corresponding circle packing for K.

B. *Circle Packing Technique*

The central issue of our circle packing problem is to adjust the radii of interior circles until all their angle sums approach 2π, which is the packing condition. One important observation inspires how we should adjust the radii of the central circles to achieve the packing condition: *the angle sum* $\theta(r;r_1,r_2,\cdots,r_k)$ *is strictly decreasing in* r. As illustrated in Fig. 2, the radii of the five petals are fixed. When r_3 increases from (a) to (c), the angle sum θ decreases. This monotonicity suggests that a strategy to adjust $R(v)$ to decrease the difference $|2\pi-\theta(v)|$ for circle c_v: decrease $R(v)$ if $\theta(v)<2\pi$; increase $R(v)$ if $\theta(v)>2\pi$.



(a) $\theta > 2\pi$ (b) $\theta = 2\pi$ (c) $\theta < 2\pi$

Fig. 2 Monotonicity of angle sum and radius of central circle.

The strategy can be implemented in an iterative fashion. Given an interior vertex v, we would ideally replace its current label r with that unique label $\bar{r}$ which gives angle sum 2π at v. However, we cannot yield a guaranteed stability without a proper choice of step size for the adjustment of $R(v)$. [5] gave another geometric monotonicity which suggests a very efficient estimation of unique label $\bar{r}$.

III. ACTIVE SENSOR NETWORK DEPLOYMENT USING CIRCLE PACKINGS

A. *Obtuse-Angle Pruning*

The uniqueness of a certain circle packing result depends on the associated triangulation of the network. However, in circle packing, boundary condition plays an important role in controlling to resultant coverage size and shape. The Delaunay triangulation always yields a convex mesh with the vertices of convex hull as boundary and this highly constrains the circle coverage size. Therefore, we propose an *Obtuse-Angle Pruning* method to reform the triangulation and increase the number of boundary vertices. The idea is simple but efficient: prune boundary edges if the associate triangle is obtuse at the opposite angle (angle opposite to the candidate boundary edge). Whenever an eligible edge is trimmed, the third vertex of the associated triangle turns to a boundary one, and so the number of boundary vertices is increased by one. However, there are a few points to notice. First, the pruning process should be done one by one on the boundary vertices. Different choices of the starting vertex would sometimes lead to different resultant boundaries. However, it does not affect the performance of the algorithm because our aim is merely to enlarge the boundary set. Second, to make sure the resultant complex K is a simply connected triangulated mesh, we should not prune an edge with a vertex of degree<2. Moreover, we should not prune a candidate boundary edge if its third vertex in the associated triangle is already a boundary vertex. This can ensure the boundary edges always form a single enclosing loop with no crossover. The *Obtuse-Angle Pruning* is described as follows:

Start from any boundary vertex $v_{current}$.

Step 1. v_{next} denotes the next boundary vertex in anticlockwise sense and v_{middle} denotes the third vertex in the associated triangle of boundary edge <$v_{current}$, v_{next}>. If *i)* deg($v_{current}$)>2, *ii)* deg(v_{next})>2, *iii)* triangle <$v_{current}$, v_{middle}, v_{next}> is obtuse at v_{middle} (i.e. $\angle v_{current}v_{middle}v_{next} \geq 90^\circ$) and *iv)* v_{middle} is not a boundary vertex, then prune boundary edge <$v_{current}$, v_{next}> (i.e. delete triangle <$v_{current}$, v_{middle}, v_{next}>). Set v_{middle} as the next boundary vertex v_{next} and repeat this step again. Otherwise, go to Step 2

Step 2. Set v_{next} as the current boundary vertex $v_{current}$. If $v_{current}$ equals the starting vertex, then end. Otherwise, go to Step 1.

Fig. 3 shows an example of the pruning process. Originally, the boundary vertices are the six vertices of the convex hull of the point set. We start to prune obtuse triangles from vertex no. 3, v_3, in anticlockwise alone the boundary. Finally, the number of boundary vertices increases to 33. Notice that although $\angle v_{42}v_9v_{21}$ is obtuse, edge $<v_{42}, v_{21}>$ is not pruned since v_9 is already a boundary vertex. The blue marks are the boundary vertices and the yellow ones are the interiors.

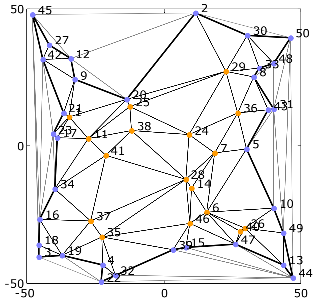


Fig. 3 An example triangulation after *Obtuse-Angle Pruning*.

B. *Overlap packings*

The central idea of circle packing problem is to adjust the radii of all interior circles to achieve the packing condition. Circle centers turn out to be secondary data. The geometric realization of a labelled complex *K(R)* can be done by fixing the coordinates of one circle and one of its neighbour, then the rest are consequently defined by the tangency relationships (Fig. 7(a)).

The circle packing problem deals with tangency of triples and so interstices always exist. However, in the sensing coverage problem, interstices are referred to as coverage holes which are undesirable. Thus, we now discuss the overlap packing problem to eliminate the interstices. The notion of *overlap angle* (Fig. 4) $\phi_{ij} = \phi(c_i, c_j)$ suggests an index to measure the extent of overlap between two circles of euclidean centers z_i, z_j and radii r_i, r_j, and its formula is

$$\phi_{ij} = \cos^{-1} \frac{(r_i^2 + r_j^2) - \left|z_i - z_j\right|^2}{2r_i r_j} . \qquad (3)$$

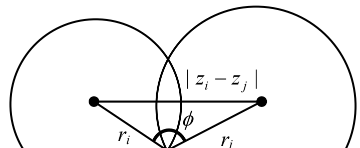


Fig. 4 Overlap angle ϕ.

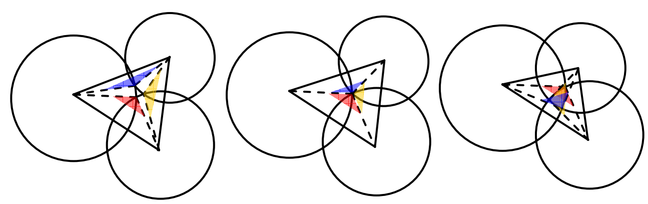
In the tangency case, the sides of the triangle of any triple is equal to the sum of two radii, i.e. $\left|z_i - z_j\right| = r_i + r_j$ for all $v_i \sim v_j$, and therefore the overlap angle always equals π. In the overlap packing case, we reduce the relative distances among the centers (which in turns scale down the sizes of triangles) by a scaling factor so that the interstices disappear. Let $\alpha \in (0,1)$ be a scaling factor for a triple $<c_1, c_2, c_3>$ with packing label r_1, r_2, r_3 to form its geometric realization, i.e.

$$\left|z_i - z_j\right| = \alpha(r_i + r_j) \text{ , } i, j = 1,2,3 \text{ and } i \neq j . \qquad (4)$$

Denote the three overlap angles of triple $<c_1, c_2, c_3>$ as $\phi_{12} = \phi(c_1, c_2)$, $\phi_{23} = \phi(c_2, c_3)$ and $\phi_{13} = \phi(c_1, c_3)$. As shown in Fig. 6, the interstice vanishes if and only if the *summation* $\sum \phi$ of these three overlap angles is smaller than or equal to 2π, i.e.

$$\sum \phi = \phi_{12} + \phi_{23} + \phi_{13} \leq 2\pi \text{ .} \qquad (5)$$

The optimal scaling factor for a triple to precisely vanish its interstice ($\sum \phi = 2\pi$ Fig. 5(b)) depends on the ratios among the three radii. Moreover, it is impractical to impose different scaling factors for different triples, because that would cause incoherence in overall circle placement. Therefore, we should apply a proper scaling factor $\alpha \in (0,1)$ globally over the entire mesh and make sure the selected α is small enough to eliminate all interstices. Fig. 6(a)-(c) shows an example triple with various scaling factors and Fig. 6(d) shows its plot of the summation of overlap angles against scaling factor α.



(a) (b) (c)

Fig. 5 Inversive distance triples: (a) Interstice exists $\sum\phi > 2\pi$ (b) Interstice just vanishes $\sum\phi = 2\pi$ (c) Interstice does not exist $\sum\phi < 2\pi$

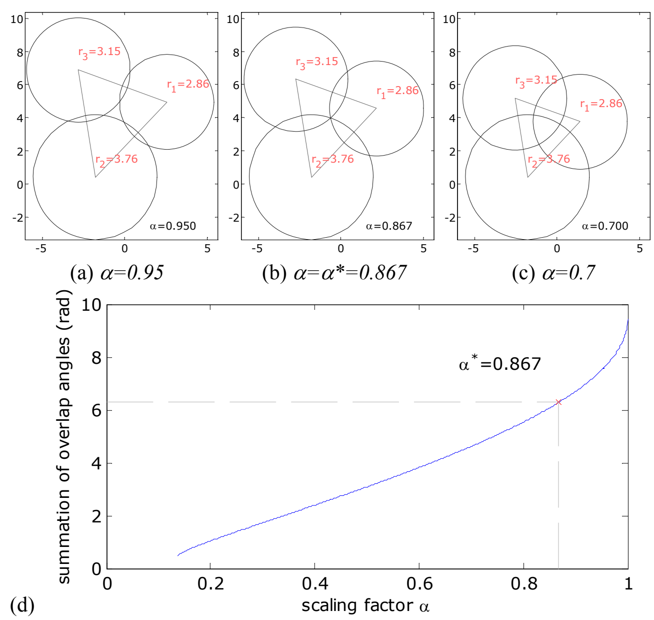


Fig. 6 (a)-(c) A comparison of different scaling factor on the same packing triple. (d) Plot of summation of overlap angles against scaling factor α.

Theorem 3.1 The smallest scaling factor required to vanish the interstice of any triple is $\frac{\sqrt{3}}{2}$.

Proof

From (3)-(5), the interstice of triple $< c_1, c_2, c_3 >$ vanishes if and only if

$$\sum \phi = \phi_{12}(\alpha) + \phi_{23}(\alpha) + \phi_{13}(\alpha)$$
$$= \cos^{-1}\left[\frac{(r_1+r_2)^2(1-\alpha^2)}{2r_1r_2} - 1\right] + \cos^{-1}\left[\frac{(r_2+r_3)^2(1-\alpha^2)}{2r_2r_3} - 1\right]$$
$$+ \cos^{-1}\left[\frac{(r_1+r_3)^2(1-\alpha^2)}{2r_1r_3} - 1\right] \le 2\pi .$$

First, we study how ϕ_{12} is influenced by the ratio of r_1 to r_2. Let $r_1 = r$, $r_2 = \tau r_1 = \tau r$ for $\tau > 0$ and $\sigma_{\alpha,r}(\tau) = \frac{(r+\tau r)^2(1-\alpha^2)}{2r\cdot\tau r} - 1$ (i.e. $\phi_{12} = \cos^{-1}\sigma$, $\sigma \in [-1,1]$) where $\sigma_{\alpha,r}(\tau)$ is the inverse distance function for certain fixed $\alpha \in (0,1)$ and r. Then, we have

$$\sigma_\alpha(\tau) = \frac{1+\tau^2 - \alpha^2(1+\tau)^2}{2\tau}$$

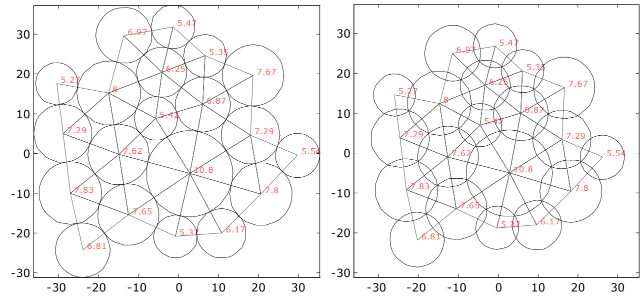
and
$$\frac{d}{d\tau}\sigma_\alpha(\tau) = \frac{(1-\tau^2)(\alpha^2-1)}{2\tau^2} .$$

Therefore, σ is independent of r. σ is strictly decreasing in $\tau \in (0,1)$ ($r_1 > r_2$) and is strictly increasing in $\tau \in (1,+\infty)$ ($r_1 < r_2$). It reaches its minimum $\sigma^* = 1 - 2\alpha^2$ when $\tau^*=1$ ($r_1 = r_2$). Moreover, for real value $\phi_{12} = \cos^{-1}\sigma$, we have $\frac{|\tau - 1|}{\tau + 1} \le \alpha \le 1$. Since $\cos^{-1}\sigma \ge 0$ strictly decreases in $\sigma \in [-1,1]$, without loss of generality, $\tau^*=1$ yields the largest ϕ_{12}, ϕ_{22}, ϕ_{13} and thus summation of overlap angles $\sum\phi$ for a certain fixed $\alpha \in [\frac{|\tau-1|}{\tau+1}, 1]$ among all the choices of τ. When the three circles have an identical radius $r_1 = r_2 = r_3 = r$,

$$\sum \phi = 3\cos^{-1}\left[\frac{(2r)^2(1-\alpha^2)}{2r^2} - 1\right] \le 2\pi .$$

Thus, $\alpha = \frac{\sqrt{3}}{2}$ is the minimum scaling factor required to vanish the interstice of any triple. ■

Hence, taking $\alpha = \frac{\sqrt{3}}{2}$ globally over the entire mesh can yield a set of sensor node positions for complex $K(R)$ while eliminating all interstices. Fig. 7 shows an example of geometric realizations obtained from conventional tangency packing and its overlap packing. Note that overlap angle ϕ_{ij} is undefined for $\alpha < \frac{|\tau-1|}{\tau+1}$. It refers to the case when the smaller circle c_i (assume $r_i < r_j$) lies strictly inside the bigger one c_j. Therefore, if the ratio of sensing radii of two adjacent nodes $\tau > \frac{2+\sqrt{3}}{2-\sqrt{3}} = 13.93$, the sensor node with the smaller radius is idle as its coverage disc lies strictly inside the bigger one and it should be redeployed.



(a) (b)

Fig. 7 (a) Tangency circle packing and (b) overlap packing with $\alpha = \frac{\sqrt{3}}{2}$

C. *Circle packing deployment algorithm*

In this subsection, we will summarize the circle packing and overlap packing strategies and describe the detail implementation of these approaches on the sensor network deployment problem. We first give the underlying assumptions:

1) At the beginning, the sensor nodes are randomly located (on a target region if applicable) while the communication graph remains connected.

2) Each sensor node is capable of broadcasting its position and obtaining the relative distances and orientations and sensing range of its neighbors.

3) Each sensor node has a upper limit of sensing range and is capable of adjusting its sensing range power.

Our central idea is to employ the circle packing result to calculate the required sensing ranges of each associated sensor node. A triangulated mesh describing the communication graph is generated upon a given initial deployment using Delaunay triangulation. Then, we use the *Obtuse-Angle Pruning* to increase the boundary size of the combinatorics complex *K*. The radii of the boundary circles will first be assigned as half of the maximum sensing ranges of the corresponding sensor nodes and gradually increase them to the maximum level. The radii of the interior sensing circles will be recursively calculated using the circle packing algorithm such that the packing condition is achieved while the communication graph is preserved as *K*. Note that if any calculated sensing circle radius exceeds the maximum range of the sensor, it means the required radius for that particular sensor cannot be achieved by its sensor node and the radius should be bounded to its maximum range. Next, the new position of each sensor node will be defined by the geometric realization of labeled complex $K(R)$ with overlap scaling factor $\alpha = \frac{\sqrt{3}}{2}$. A new complex *K* will be defined by the newly obtained deployment and the whole process should be executed iterative until equilibrium state is reached. The *Circle Packing Deployment Algorithm* is summarized as follows:

Step 1: Set maximum number of iterations and threshold to determine if deployment has reached equilibrium.

Step 2. Define the triangulation complex K base on current sensor node positions.

Step 3. Increase the number of boundary vertices using the *Obtuse-Angle Pruning* method.

Step 4. Set the sensing radii of boundary vertices to a certain level in the first iteration. Then gradually increase them to their upper limits in successive iterations.

Step 5. Execute *Circle Packing* algorithm on each interior node in a distributed sense to find its sensing range radius. Bound all sensing ranges to their upper limits.

Step 6. Deploy all sensor nodes based on the circle packing result and take $\alpha = \frac{\sqrt{3}}{2}$ to ensure no coverage hole exists in the connected coverage region.

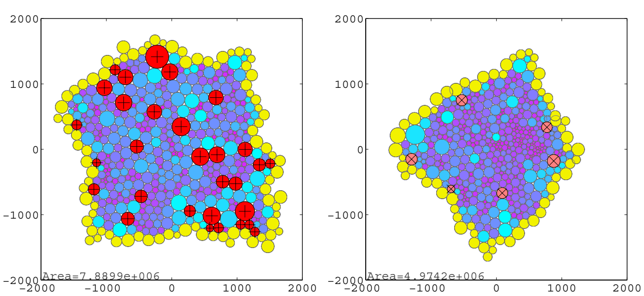
Step 7. Go to Step 2 if movement of sensor nodes is greater than the threshold (i.e. equilibrium not reached) and maximum number of iterations is not reached. Otherwise, end.

IV. SIMULATION EXAMPLES

We have implemented the proposed algorithm in Matlab to verify the approach and demonstrate its performance. Extensive simulations performed show that the circle packing approach can always give deployments of large sensing coverage. Two examples are given below.

A. *Example 1*

Our circle packing deployment algorithm is capable for constraining the sensing ranges. Fig. 8 shows the circle packing deployment results of 500 sensor nodes with different sensing range limits. If we do not limit the sizes of the interior circles, the algorithm may yield a circle packing result that some circles are bigger than their maximum sensing ranges, as indicated by the red circles with '+' mark at center in Fig. 8(a). When we limit the sensing range, another packing result will be obtained as shown in Fig. 8(b), where the six pink circles represent the nodes that have reached their maximum sensing limits.

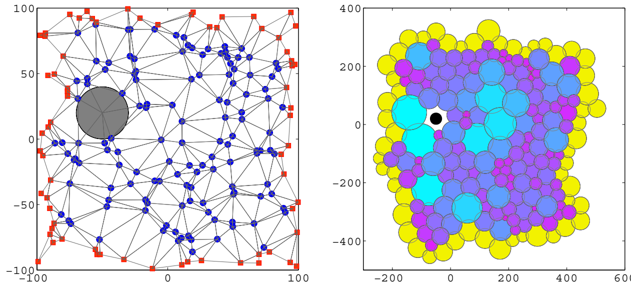


(a) (b)

Fig. 8 Circle packing: (a) without and (b) with sensing range limits

B. *Example 2*

In this example, we illustrate the capability of the proposed algorithm to deploy a sensor network in the presence of obstacle. In Fig. 9, a stationary circular obstacle of radius $r_{ob} = 20$ is located in coordinates (-50,20). In Fig. 9(a), 200 sensor nodes are randomly located in the field. The obstacle is considered as a virtual interior node with 7 neighbors and of radius no less than r_{ob}. The new positions of all sensor nodes are defined by fixing the virtual obstacle circle. Fig. 9(b) shows the final deployment obtained by overlap circle packing.



(a) (b)

Fig. 9 Obstacle avoidance: (a) initial and (b) final deployment.

V. CONCLUDING REMARKS

This paper addresses the problem of autonomous deployment of active sensor networks. Upon an initial random placement, a triangulated mesh describing the neighboring relationship among the nodes is generated and refined by *Obtuse-Angle Pruning* to increase the number of boundary nodes. We have employed the circle packing technique to find the radius of each interior sensing circle. Then, the geometric realization of the circle packing result is done by fixing one node and one of its neighbor via overlap packing. We have proven that a global scaling factor of $\alpha = \frac{\sqrt{3}}{2}$ can always vanish interstices of any triple which represent coverage holes. We have implemented the algorithm in Matlab and demonstrate its performance with numerical examples. We have verified that the proposed algorithm always yields sensor deployments of wide coverage, avoids obstacle and minimizes the sensing range required for every interior sensing node for achieving the packing condition.

REFERENCES

[1] A. Howard, M. J. Mataric' and G. S. Sukhatme, "Mobile Sensor Network Deployment using Potential Fields: A Distributed, Scalable Solution to the Area Coverage Problem," in *Distributed Autonomous Robotic Systems*, pp. 299-308, Springer, 2002.

[2] Y. Zou and K. Chakrabarty, "Sensor Deployment and Target Localization based on Virtual Forces," *INFOCOM*, April 2003.

[3] G.-L. Wang, G. Cao and T. L. Porta, "Movement-Assisted Sensor Deployment," *INFOCOM*, March 2004.

[4] M.-L. Lam and Y.-H. Liu, "ISOGRID: an Efficient Algorithm for Coverage Enhancement in Mobile Sensor Networks," *IROS*, Oct. 2006.

[5] K. Stephenson, Introduction to Circle Packing : The Theory of Discrete Analytic Functions. New York: Cambridge University Press, 2005.

[6] W. Thurston, *The finite Riemann mapping theorem*, Invited talk, An International Symposium at Purdue University in celebrations of de Branges' proof of the Bieberbach conjecture, March 1985.

[7] B. Rodin and D. Sullivan. "The Convergence of Circle Packings to Riemann mapping," *J. of Differential Geometry* vol. 26, p.349-360, 1987.