

Processing of Deployment Using an Adaptive Self-Spreading Algorithm Based on Electrostatic Field for Mobile Sensor

<http://dx.doi.org/10.3991/ijoe.v12i01.5131>

Ming Li^{1,2}, HuanYan Qian¹

¹ Nanjing University of Science and Technology, Nanjing, China

² Hohai University, Nanjing, China

Abstract—The deployment of mobile sensor networks is a hotspot issue. This paper is dedicated to proposing a distributed, adaptive, and scalable mobile sensor network algorithm based on the theory in electrostatic field. In the algorithm, the obstacles and nodes in the deployment region were regarded as the charged particles, which were driven by the Coulomb Force exerted by the other particles and obstacles, and eventually all the nodes were distributed throughout the entire network while repelled by the obstacles and other nodes simultaneously. Based on the classical virtual potential field algorithm, the boundary conditions, the static equilibrium conditions and the controllable coverage rate were expanded in the present algorithm, which are also the primary innovation points and contributions of this paper. The efficiency of the proposed algorithm was evaluated by performing simulations under four frequently-used scenarios, namely, deployment under normal conditions, deployment with obstacles, re-deployment due to unpredicted node failures, and deployment with regions of interest. Moreover, the performance of the algorithm was assessed using coverage and uniformity as the criteria. The simulation results indicate that the algorithm runs well under all these four frequently-used scenarios. In a completely unknown deployment area, the designated goal can also be achieved using the proposed algorithm owing to its adaptivity.

Index Terms—Deployment, Mobile Sensor Network, Electrostatic Field, Potential Field.

I. INTRODUCTION

Sensor deployment [1-3] is an urgent issue that needs to be solved in the applications of Wireless Sensor Networks (WSNs). It imposes a determinant effect on the accuracy, completeness, and timeliness of network monitoring. Currently, the studies on sensor deployment in the related field have always focused on static sensor networks [4-6]. Most of the researches are limited to the scenarios that all sensor nodes are stationary, and thus are not applicable to some situations where mobile sensors are required, for instance, monitoring the life of wild animals, tracking patients' heartbeat, etc. In recent years, with the rapid developments of sensors, embedded systems and robotics technologies, mobile sensor networks have attracted increasingly attentions and the related applications also have become popularized [7-8]. The deployment algorithm proposed in this paper is exactly dedicated to solving the problems in mobile sensor networks.

Sensor deployment refers to that, in a certain region, the sensor nodes are distributed under appropriate strategies, in order to meet some specific requirements or perform certain functions. How to optimize the number and distribution strategy of the sensor nodes, make a high-efficiency exploitation of the limited resources in a sensor network, and minimize the energy consumption are all the important tasks to be addressed when deploying sensors. The history of sensor deployment can be traced back to two classic problems in computational geometry: the art gallery problem proposed by O'Rourke, and the perimeter coverage problem proposed by Williams [9-10]. An appropriate approach to deploying sensor nodes can not only maximize network efficiency and optimize network resources, but also can change the number of active sensors according to different requirements in different applications, in order to dynamically adjust the node density of network. Besides, in the cases that some sensor nodes are disabled due to physical damage or energy depletion, the performance of network can maintain uninfluenced by the re-deployment of sensor nodes under certain strategies, i.e., the network has strong robustness.

In the present work, focusing on the self-deployment of mobile sensor nodes, the mobile sensor nodes were considered as the electrically charged particles. These particles have same or opposite charges, suggesting that they are attracted or repelled by each other under the effect of Coulomb force. According to the Newton's second law of motion, these particles under force will move and tend to be accelerated. Consequently, the research emphasis of this paper is how to control and manipulate the trajectory of the moving particle so that they can be deployed in the expected region of interest.

In this paper, the governing equations, which the movement of particles follows, were established strictly based on the Coulomb's law and the Newton's second law of motion. Accordingly, the problem of mobile sensor deployment was transformed into the solution of classic mechanics equations, by which the complexity of the deployment problem was simplified drastically and the difficulties of deployment were reduced. Especially for the deployment in certain unknown regions, owing to the strong adaptivity of the proposed algorithm, sensor nodes can spontaneously move to reach the best coverage without any additional information of the size and environment of the regions.

Additionally, an explanation of the proposed deployment algorithm was provided in detail. In order to verify

the effectiveness and correctness of the algorithm, with the adoption of coverage, uniformity, deployment time and average distance as the assessment criteria, a certain number of simulations under four frequently-used scenarios were conducted, namely, deployment under normal conditions, deployment with obstacles, re-deployment due to unpredicted node failures, and deployment with regions of interest. The results indicate that the algorithm performs well under all the above-mentioned scenarios.

In almost all the literatures regarding virtual potential field algorithm, zone boundaries and barriers were directly assumed as high potential, in order to prevent from border-crossing of nodes. Thus, few studies have investigated the boundary conditions in depth. Aiming at exploring the function of boundary conditions, the conditions and influencing factors for the static equilibrium of network were analyzed in this paper. Furthermore, almost all the previous literatures were oriented on the maximum of coverage rate. On the contrary, the feasibility of controlling coverage rate from another perspective was investigated in this paper. Compared with the previous research achievements, the main contributions and innovation points of the present work are listed below.

- (1) To discuss the role of boundary conditions in node deployment and explain the influence of boundary conditions on the coverage rate of network;
- (2) To explore the conditions which contribute to the static equilibrium of network and the factors which influence the network to achieve a static equilibrium;
- (3) To establish a mathematic model for network coverage so that the network can be deployed precisely according to the pre-set coverage rate;
- (4) To optimize the computational formula for uniformity.

II. ELECTROSTATIC DEPLOYMENT MODEL

A. Electric Field

As a commonly-used and well-understood method in the field of mobile robotics, potential field theory has been widely applied in several tasks such as local navigation and obstacle avoidance. In the present work, potential field theory was employed to solve the deployment problem. Given the following scenario, two point charges with the same sign are stationary in vacuum, with the charges of Q_1 and Q_2 , respectively, and the distance between them is $2a$, as shown in Figure 1.

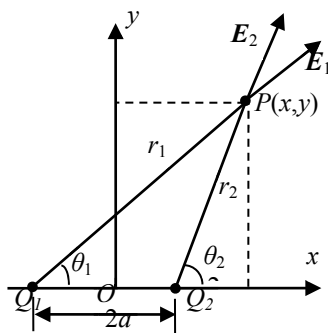


Figure 1. A pair of point charges

The electric potential at point $P(x,y)$ yielded by two point charges with like charges of Q_1 and Q_2 can be written as:

$$U = U_1 + U_2 = \frac{kQ_1}{r_1} + \frac{kQ_2}{r_2} \quad (1)$$

where r_1 denotes the Euclidean distance between P and Q_1 ; r_2 denotes the Euclidean distance between P and Q_2 , and k is the Coulomb's Constant.

Based on the definition in fundamental field theory, the intensity of the electric field generated by point charges can be computed via its potential gradient:

$$E = -\nabla U \quad (2)$$

where ∇ is a gradient operator,

$$\nabla = \frac{\partial}{\partial x} i + \frac{\partial}{\partial y} j + \frac{\partial}{\partial z} k \quad (3)$$

The electric field lines and equipotential lines yielded by two point charges with like charges of Q_1 and Q_2 are shown in the left part of Figure 2. The electric field lines start from the positive charges and extend towards infinity. The electric field lines are perpendicular with the equipotential line, and any two electric field lines are disjointed. The right part of Figure 2 displays the x-component of the field intensity.

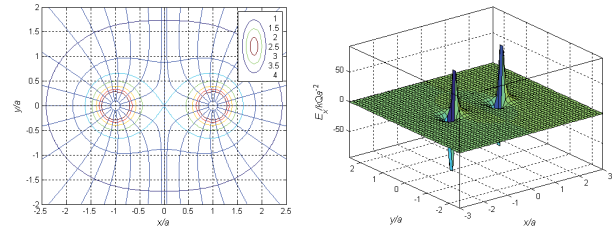


Figure 2. Electric field lines and equipotential lines yielded by two point charges with like charges (left) and the x-component of the field intensity (right).

The situation shown in Figure 1 can be generalized to other cases. Assume that all the sensor nodes and obstacles in the mobile sensor network are the point charges, U_n and U_o denote the superimposed electric potential fields at P induced by all the nodes and obstacles, respectively, and then the potential at P can be expressed as $U_n + U_o$. A single node N_i in the mobile sensor network can be regarded as a point charge with the charge of q , the electric force F_i exerted on this node at point P in Figure 1 can be expressed as:

$$F_i = E_q = -\nabla(U_x + U_o)q \quad (4)$$

It can be easily inferred from the Newton's second law of motion, the node N_i starts moving and accelerating due to the electric force F_i .

B. Governing Equations

The nodes are subject to resistance from the environment in actual deployment process. Suppose that the damping factor is μ , the resistance is proportional to the velocity v_i , and the mass of the node is m , the acceleration of the node can be computed as:

$$a_i = (F_i - \mu v_i) / m \quad (5)$$

Suppose that the initial time of deployment is t_0 , the initial velocity of all nodes is 0, i.e., $v_0=0$, and the initial position of a certain node N_i is $P_i(t_0)$, the electric force exerted on node N_i can be computed using the above-described (4), and therefore the acceleration $a_i(t_0)$ at t_0 can be computed as:

$$a_i(t_0) = F_i(t_0) / m \quad (6)$$

If t is quite short, it can be assumed that N_i moves at an identical acceleration in infinitesimal time interval t , then the velocity $v_i(t_1)$ at t_1 ($t_1 = t_0 + t$) can be expressed as:

$$v_i(t_1) = v_i(t_0) + a_i(t_0)\Delta t = a_i(t_0)\Delta t \quad (7)$$

The new position of node N_i at moment t_1 , $P_i(t_1)$, can be written as:

$$P_i(t_1) = P_i(t_0) + \frac{1}{2} a_i(t_0)\Delta t^2 \quad (8)$$

When the node N_i reaches the new position $P_i(t_1)$, the acceleration $a_i(t_1)$ at t_1 needs to be re-calculated based on electric force and resistance exerted on the node at new position. Therefore, the position, velocity, and acceleration of node N_i at t_1 can be computed. Repeatedly, the position, velocity, and acceleration of node N_i at any moment t_n can be computed. Furthermore, the position, velocity and acceleration of all the nodes at any moment t_n can be computed in the same manner.

Considering that the velocity and acceleration of nodes are finite in practical applications, two threshold values, v_{th} and a_{th} , were introduced to limit the nodes' velocity and acceleration, respectively.

C. Boundary Condition

Generally, the region of deployment is limited and the nodes should be deployed within the boundary of the region. The boundaries can be physical (e.g. mountains, rivers and etc.), or hypothetical (e.g. latitude and longitude lines). When the nodes move towards or approach the boundary, a corresponding approach is required to ensure that the nodes cannot exceed the boundary, also known as the boundary conditions, as shown in Figure 3.

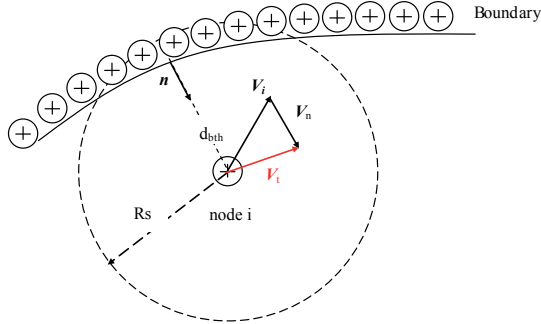


Figure 3. Illustration of the boundary condition

Suppose that the boundary of the deployment region consists of a large number of charges with the same sign, when the distance d between node and boundary is less than the sensing radius of the node R_s , the node will change the direction of net force due to the Coulomb force exerted by the boundary, resulting in a change in velocity due to the superposition of original velocity V_i and vertical boundary velocity V_n . Accordingly, a new velocity V_t is produced, and the change in the direction of V_t will eventually drive the node away from the boundary. Therefore, there is always a safe distance between node and boundary.

Assume that the threshold value of the distance is d_{bth} , the number of electric charges within per unit length along the boundary is equal to the number of electric charges carried by a single node, and the number of electric charg-

es carried by a single node is supposed to be equal to the sensing radius R_s of the node in quantity.

$$F_{ib} = \begin{cases} 0, & d_{ib} \geq d_{bth} \\ \beta \frac{q_i q_b}{d_{ib}^2} = \beta \frac{R_s^2 R_s^2}{d_{ib}^2}, & d_{ib} < d_{bth} \end{cases} \quad (9)$$

In (9), q_i and q_b denote the electric charges carried by node i and point b along the boundary, respectively. The quantity of electric charges carried by point b is equal to the number of electric charges carried by unit length, i.e., R_s in amount. d_{ib} refers to the Euclidean distance between node and point b while d_{bth} denotes the threshold value of distance far away from the boundary. When the distance between node and boundary is larger than the threshold value, no influence will be generated ($F_{ib}=0$). When the distance is smaller than the threshold value, a repulsive force will be generated and then imposed on the node. β is a proportionality coefficient, by which the repulsive force can be adjusted. In (9), the force exerted on the node i by barriers and the unit length of boundary can be calculated. Correspondingly, the resultant force exerted on the node i by the whole barrier can be worked out by performing integral computation on the entire boundary.

D. Static Equilibrium

Within the region of deployment, the nodes are in motion due to the Coulomb force, but the process will not last forever. When all the nodes are distributed all over the region, the Coulomb force among them decreases over time. When the net force on a node is close to 0, it tends to stop motion since the acceleration is lost, and finally the equilibrium is achieved.

Assume that there is a virtual attractive force and a repulsive force between any pair of adjacent nodes in the sensor network, the virtual force exerted on node s_i by s_j , labeled as F_{ij} ($j=1, 2, \dots, k$, while $s_j \in S_{Ni}$, and S_{Ni} is the group of the adjacent nodes of node s_i), can be calculated as:

$$F_{ij} = \begin{cases} 0, & d_{ij} = d_{th} \\ \omega_A \left(\frac{1}{d_{th}} - \frac{1}{d_{ij}} \right), & d_{ij} > d_{th} \\ \omega_R \frac{q_i q_j}{d_{ij}}, & d_{ij} < d_{th} \end{cases} \quad (10)$$

d_{ij} is the Euclidean distance between node s_i and node s_j (s_j and s_i are the adjacent nodes, so that $d_{ij} < R_c$, and R_c refers to the communication distance of nodes), d_{th} represents the threshold value of distance; ω_A and ω_R denote the attractive coefficient and repulsive coefficient between the nodes, respectively. Accordingly, the net force exerted on node s_i by its adjacent nodes can be written as:

$$F_i = \sum_{s_j \in S_{Ni}} F_{ij} \quad (11)$$

Assume that F_a denotes the ultimate net force exerted on the node, F_i denotes the net force exerted on the node by the adjacent nodes and F_b denotes the net force exerted by the boundary, the following expression can be obtained:

$$F_a = F_i + F_b - \mu v \quad (12)$$

In (12), μv refers to the resistance exerted on the node. When node i approaches to a static equilibrium, its speed is zero, i.e., $v = 0$, and thus $\mu v = 0$. It can be concluded that, the net force exerted on the node is $F_a=0$, and moreover, $F_i=0$, $F_b=0$. In accordance with (9), when the distance between node i and the adjacent nodes S_{Ni} follows $d_{ij}=d_{th}$, F_i is zero. According to (10), when the distance between node i and the boundary complies with $d_{ib} \geq d_{bth}$, F_b is zero. Thus, with regard to node i , the following conditions should be satisfied for achieving a static equilibrium: $d_{ij}=d_{th}$ and $d_{ib} \geq d_{bth}$.

According to (5), (7) and (12), it can be concluded that the factors which affect the duration for nodes to achieve static equilibriums include the quantity and electric charge of the node, the damping factor μ of deployment region, as well as the threshold values of distance d_{th} and d_{bth} . Figure 4 shows the influence of distance threshold d_{th} on the duration for nodes to achieve static equilibriums, from which one can find, with the gradual decrease of d_{th} ($0.5R_c \sim 1.5R_c$), the net force exerted on the node grows correspondingly (in the left part of Figure 4). Besides, as the average movement speed of nodes increases constantly (in the right part of Figure 4), the duration for nodes to achieve static equilibriums becomes longer and longer. The influence of boundary threshold d_{bth} on node's static equilibrium is similar to that of d_{th} , which won't be stated repeatedly here. Other influencing factors will be described in the simulation chapter of the paper.

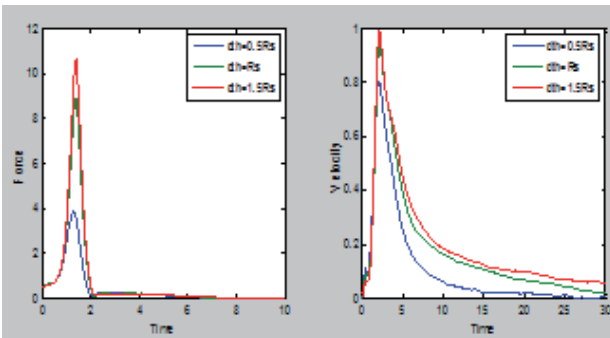


Figure 4. Influence of distance threshold d_{th} on the duration to achieve a static equilibrium

The above-described argument rests on the assumption that the environment is static. When the environment of deployment is changing, the nodes which had already reached equilibriums may need to be re-deployed in order to reach new equilibriums. In other words, the algorithm is adaptive. However, if the environment of deployment is changing, the nodes may only reach temporary equilibriums. The equilibrium will be broken induced by the changes in environment and a further re-balance will be achieved. The energy will be wasted during this re-deployment process.

III. ADAPTIVE SELF-SPREADING ALGORITHM

Suppose that there is a region A in which the sensor nodes need to be deployed in order to monitor the region. How to deploy a certain number of sensor nodes to meet the expected deployment results? A parameter, known as coverage ratio, was introduced hereby. According to the definition in [11], the coverage ratio can be expressed as:

$$C(R_c) = \frac{\pi R_c^2 N}{A} \quad (13)$$

in which, N denotes the total number of nodes, A denotes the area of the deployment region, R_c is the communication radius of sensor nodes, and C is the coverage ratio of the region. As indicated in (13), the coverage ratio C reflects the ratio of the sum of communication coverage area of all the nodes (with a total number of N) to the area of deployment region A . The value of C indicates the communication coverage of the region. A larger coverage ratio can provide a network with better connectivity and higher service quality. Conversely, given an expected value of C , the required minimum number of nodes can be easily calculated, denoted as N , on the condition that the area of the region is known. When deploying sensors using this algorithm in practical applications, an exact size of the region is not necessary. Instead, only an approximate value of area A is needed for calculating an approximate number of nodes required.

When deploying the nodes, the initial positions can be set randomly or artificially, and both two arrangement methods will not affect the process of deployment. Starting from the initial time t_0 , the nodes move under the influence of electric force. To be specific, the electric forces exerted on the nodes are from one or more objects among the following: other nodes, obstacles and boundaries, in addition to viscous resistance F_d . These forces will form a net force exerted on the node, and the node will move along the direction of the net force until reach an equilibrium eventually. When all the nodes reach equilibriums, the deployment process is finished, as shown in Figure 5.

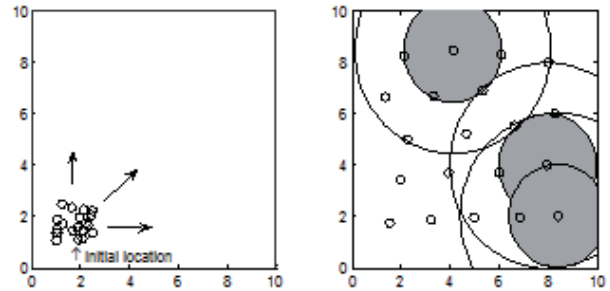


Figure 5. Deployment process

In practical applications, there might be some cases that a certain number of nodes are disabled due to horrible weather conditions, battery depletion or other triggers. Disabled nodes may lead to the loss of monitoring coverage in certain regions. When this situation occurs, the previous equilibrium reached by using adaptive self-spreading (ASS) algorithm will be broken so that all the nodes need to be re-located to cover these 'exposed' areas, finally achieving new equilibriums.

The whole process of the proposed adaptive self-spreading (ASS) algorithm is presented in Figure 6.

Using 3-D topographic maps, how the proposed algorithm is working can be illustratively presented. As shown in Figure 7, this is a computer-generated 3-D topographic map. There are mountains (marked in red), lakes (marked in blue), and relatively flat regions. The initial positions of the nodes are in the lower right corner, marked by the red rectangle, in which the white dots denote the mobile sensor nodes to be deployed.

Algorithm : Adaptive Self-Spreading Algorithm

Requirement: A, N, C, Rc, Rs, m.

- 1: Estimate the number of nodes N based on the expected coverage ratio C and the area of region A
- 2: At time t_0 , all the nodes (with the number of N) are located at their initial positions
- 3: For each node i at time t
- 4: Compute the Coulomb force F_e exerted by the other nodes in the sensing radius R_s
- 5: Compute the Coulomb force F_o exerted by the obstacles in the sensing radius R_s
- 6: Compute the damping force F_d on the node
- 7: Compute the net force on node i : $F_i = F_e + F_o + F_d$
- 8: Compute the acceleration a_i of node i at time t
- 9: Compute the velocity v_i of node i at time t
- 10: Determine the relationship between the threshold values a_{th} & v_{th} and the calculated a_i & v_i respectively, and then proceed accordingly.
- 11: Compute the position $P_i(t)$ of node i at time t , and the node moves to this position
- 12: Determine if node i reaches an equilibrium at time t .; if yes, terminate the node's motions; otherwise the loop is executed again.
- 13: End For
- 14: If all the nodes reach equilibriums, the algorithm ends.

Figure 6. Adaptive self-spreading algorithm

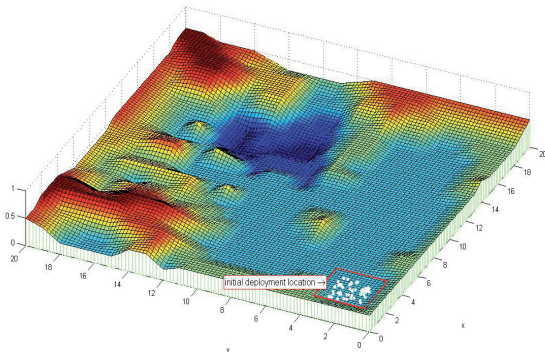


Figure 7. Initial deployment locations

Figure 8 shows the final locations of the nodes after deployment using the deployment algorithm proposed in this article. It can be observed that the nodes are practically distributed in the relatively flat regions rather than around obstacles such as mountains, hills, lakes and *etc.* The region with a hill is marked by a white rectangle, and one can find that the nodes now are distributed around it.

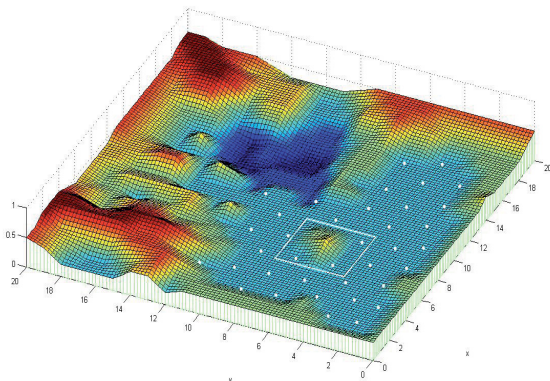


Figure 8. Final locations of nodes after deployment

As shown in Figure 9, to evaluate the adaptivity of the proposed deployment algorithm, the region with a hill marked by a white rectangle was removed. It can be observed that, the equilibriums of the nodes was broken and all nodes were relocated until they reach new equilibri-

ums. No ‘exposed’ area appears, since the nodes around the hill have successfully re-covered the region.

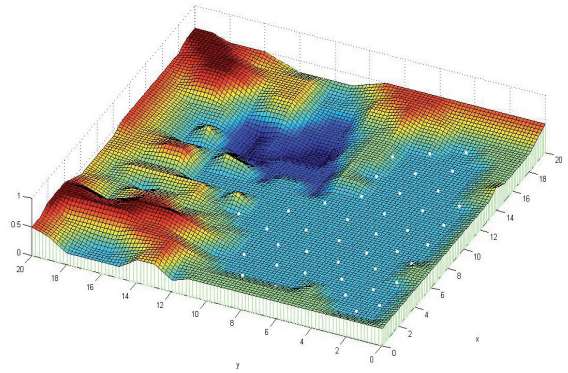


Figure 9. Re-deployment after the removal of hill

These analyses indicate that the algorithm proposed by the previous researchers were optimized and simplified in the present work. The complexity of similar algorithms was effectively reduced while the most important feature, adaptivity, was successfully reserved.

IV. SIMULATION RESULTS

Generally, the coverage can be used for measuring the service quality of a sensor network. As reported in [11], Gage invented the concept of coverage in the researches regarding multi-robot systems [12], which is defined as the ratio of the sum of the coverage area of all the nodes to the area of the entire target region:

$$\text{coverage} = \frac{\bigcup_{i=1..N} A_i}{A} \quad (14)$$

The definition of sum of the coverage area was taken from the concept of union in the Set Theory, which is thus usually less than or equal to 1. The uniformity of coverage is a well-defined metric to measure the life of a network. In [11] the uniformity in sense of the standard deviation of the distances between nodes was described. The smaller the uniformity of coverage is, the more uniformly the network is covered. However, this calculation method for uniformity is imperfect. As shown in Figure 10, regarding the calculation of the uniformity of the node sits at coordinate (0,0), there are 4 adjacent nodes in its sensing radius. According to the approach described in [11], the calculation results of uniformity in Figure 10(a) and 10(b) are the same. However, it can be easily found that the right one exhibits a better uniformity.

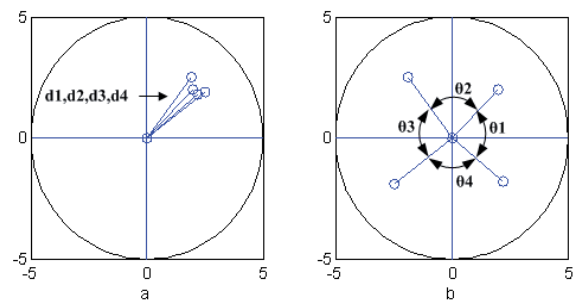


Figure 10. Topology of nodes for uniformity

How to compute uniformity more precisely? The approach described in [11] only considers the distance with-

out the angle between the nodes, leading to a fault result in calculating the cases as shown in Figure 10. In this article, with both the standard deviation of distances and the angles between nodes taken into accounts, the product of the two standard deviations is used for representing the uniformity:

$$\begin{cases} U_i^d = \left[\frac{1}{n} \sum_{j=1}^n (d_{i,j} - \bar{d})^2 \right]^{\frac{1}{2}} \\ U_i^\theta = \left[\frac{1}{n} \sum_{j=1}^n (\theta_j - \bar{\theta})^2 \right]^{\frac{1}{2}} \\ U = \frac{1}{N} \sum_{i=1}^N U_i^d \sum_{i=1}^N U_i^\theta \end{cases} \quad (15)$$

where U is the uniformity, N is the total number of nodes, σ_d is the standard deviation of the distance between the i -th node and its adjacent nodes, σ_θ is the standard deviation of the angle between the i -th node and its adjacent nodes, n is the number of the adjacent nodes of the i -th node, $d_{i,j}$ is the distance between the i -th and j -th nodes, \bar{d} is the mean value of the internodal distances between the i -th node and its adjacent nodes, $\bar{\theta}$ is the mean value of the internodal angles between the adjacent nodes of the i -th node.

Numerous simulations have been carried out under various environmental conditions to evaluate the performance of the proposed algorithm. Table 1 lists the parameter setting in the simulations corresponding to the related figures.

TABLE I.
PARAMETERS SETTING IN THE SIMULATIONS

Parameter	Fig. 11	Fig. 12	Fig. 13	Fig. 14
Deployment Area (A)	10×10	10×10	10×10	10×10
Node number (NN)	30	30	30	30
Charges (Q)	1e-9	1e-9	1e-9	1e-9
Quality (m)	9e-9	9e-9	9e-9	9e-9
Coulomb's constant (k)	9e9	9e9	9e9	9e9
Rs, Rc	2; 4	2; 4	2; 4	2; 4
Damping factor	0.5	0.5	0.5	0.5
a_{th}, v_{th}	2; 1	2; 1	2; 1	2; 1
Time (T)	20	20	40	20
Dt	0.2	0.2	0.2	0.2
Number of obstacles	0	2	0	0
Number of ROI	0	0	0	2
Number of invalid nodes	0	0	5	0

Figure 11 displays the simulation results using the parameters listed in Table 1. The calculation results after deployment are shown in the left part. The red rectangle represents the size of the region (area A), the tiny black dots represent the positions of the nodes; while the small (shaded) and large circles represent the sensing radius and the communication radius, respectively. Nodes can probe the environment and collect information in the sensing radius R_s . Similarly, it can be influenced by the Coulomb force from its adjacent nodes and repulsive force from the obstacles as well. The communication radius of the node R_c is greater than the sensing radius R_s . Nodes can exchange information with the other nodes within its communication radius. The initial positions of the nodes are in

the lower left corner (close to the coordinate [1,1]). Imaginably, the node density at the initial moment is practically very large. When start deploying, all the nodes are forced by the un-balanced Coulomb force, start to move in order to fill the entire region, and eventually reach equilibriums, as the final positions shown in Figure 11. The right part of Figure 11 shows the changes in coverage and uniformity versus time during this deployment process. The x-coordinate is the simulation time T; the y-coordinate on the left represents the uniformity; while the y-coordinate on the right represents the coverage. It can be observed that, at the initial moment of deployment, the nodes can only cover a small part of the region because they are all initially concentrated in the lower left corner, leading to a fairly low coverage (<20%). As time goes, all nodes are in motion and the coverage tends to grow, finally achieving equilibriums at a certain moment (about at Time=7). This is the moment the maximum coverage is achieved and maintained until the simulation is finished. The value of uniformity is approximately 0.5 at the beginning of simulation. Then it grows rapidly after the beginning of deployment, reaches a maximum and starts to fall back. Eventually, the uniformity tends to be at around 0.2. The uniformity represents the standard deviation, i.e., the smaller it is, the more uniform the network is. Why does the maximum appear shortly after the beginning of the deployment instead of at the initial moment? According to (15), the computation of uniformity only considers the standard deviation of distances and angles between the nodes, which cannot reflect how well-distributed all the nodes in the network. To evaluate the distribution, the grid approach is introduced, which will be the focus in future work.

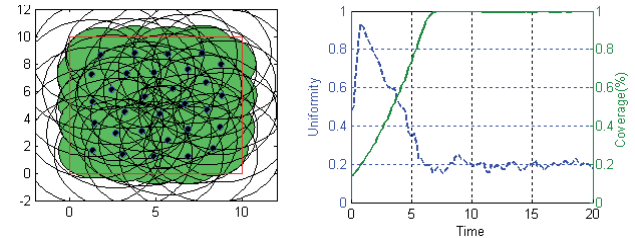


Figure 11. Deployment under normal conditions

In the left part of Figure 12, the red circle represents obstacles. In the previous sections, it has been mentioned that the obstacle can be considered as the stationary charges with the same sign. But the charge it carries is proportional to its size, i.e., the larger the obstacle is, the more charge it carries. There are two obstacles in the left part, in which the smaller one has 3 times charges than that of a regular node and the larger one has 8 times charges. The distance between node and obstacle can be adjusted by configuring the amount of charges the obstacle carries. As shown in the left part of Figure 12, the nodes will be deployed at a distance away from the obstacle so as to avoid them, due to the repulsive force exerted by the obstacles. This is highly meaningful in practical applications. To keep the nodes away from hazard or unreachable regions can minimize the damage on nodes, indicating the the adaptivity of the proposed algorithm. The right part displays the changes in coverage and uniformity versus time during this deployment process. It can be observed that, due to the existence of obstacles, the region is not completely covered by nodes. The maximum of coverage is less than 1. In the meantime, compared with the value in Figure 11, the value of uniformity is

higher after the nodes reach equilibriums. All of these variations are due to the existence of the obstacles.

In the left part of Figure 13, the red dots represent the disabled nodes. The deployment process consists of two stages. In stage 1, the nodes are deployed completely under a regular circumstance until the equilibrium (T=20). In stage 2, some nodes are disabled due to environmental damages or other factors in simulation. As shown in Figure 13, there are 5 nodes out of work, and the equilibrium of the network is broken. The nodes are re-deployed to fill the 'blanks' due to the failed nodes, i.e., the nodes move towards the failed nodes to cover the 'exposed' areas. After slight adjustments, the nodes reach equilibriums once again, as shown in the left part. Based on the coverage and uniformity curves in the right of Figure 13, the process can be illustrated clearly. At Time=20 (where the arrow points), there's significant decrease in terms of coverage rate. In the meantime, the value of uniformity increases greatly. All of these are caused by node failures. After the re-deployment (20<Time<40), the nodes are in new equilibriums and the coverage is close to 1. Yet the value of uniformity remains unchanged at 0.4, which is higher than the previous value 0.2. The decrease in uniformity is also caused by node failures.

In Figure 14, the blue regions in the left part represent the regions of interest (ROI). ROI is the region where nodes need to be deployed with a high density, because it is usually the monitoring focus. Two ROIs can be found in Figure 14, in which the one in the lower left corner is relatively small while the one in the upper right corner is large. As explained in the previous section of this paper, the ROIs can be regarded as the stationary charges with the opposite signs. The amount of charges it carries determines the attraction on the nodes. Assume that the smaller area contains a negative charge, while the larger area contains three negative charges. It can be observed that, after the deployment, there are more nodes gathered around ROIs, suggesting that the nodes in ROIs are of high density. Therefore, the ROIs can be better covered and served. As displayed in the coverage and uniformity curves in the right part, after the equilibrium is reached, coverage is slightly less than that in normal deployment (as shown in Figure 11). That is caused by the attraction of nodes imposed by the ROIs. The value of uniformity is slightly larger than that in Figure 11, which means that the uniformity when deploying with the existence of ROIs is slightly worse than that in normal deployment.

V. CONCLUSIONS

The deployment of mobile sensor networks is a hot spot issue. How to deploy the sensor nodes more uniformly and quickly with lower energy consumption, and maximize the lifespan of the network are each deployment algorithm is pursuing. In practical applications, various kinds of complex problems and difficulties may arise, and therefore a deployment algorithm with scalability, adaptivity, robustness and conciseness, is absolutely necessary. Based on the analyses of the previous achievements, a distributed, adaptive and scalable mobile sensor network deployment algorithm was proposed in the present work. In the algorithm, the obstacles and nodes inside the region of deployment are regarded as the charged particles, and the particles are driven by the Coulomb force exerted by the other particles. Eventually, all the nodes are spread and

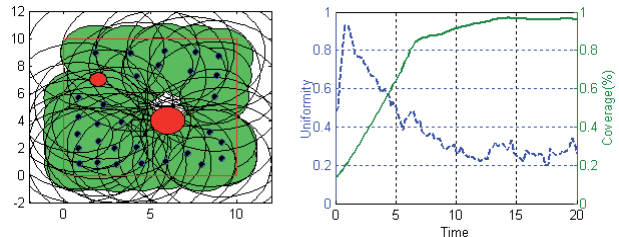


Figure 12. Deployment with obstacles

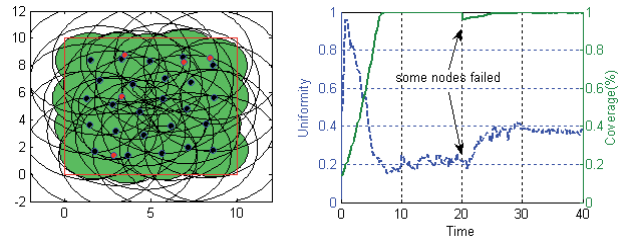


Figure 13. Re-deployment after node failures

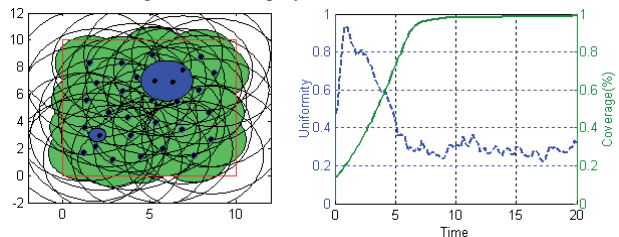


Figure 14. Deployment with Regions of Interest (ROIs)

distributed throughout the entire network, which are repelled by the other particles.

In order to evaluate the performance of the proposed algorithm, the simulations were performed under four frequently-used scenarios: deployment under normal conditions, deployment with obstacles, re-deployment due to unpredicted node failures, and deployment with regions of interest (ROIs). The coverage and uniformity were adopted as the assessment indexes to evaluate the performance of the algorithm. The simulation results indicate that the algorithm performs well under all frequently-used scenarios. Moreover, in a completely unknown deployment region, the algorithm can take full advantage of its adaptivity to achieve a designated goal.

ACKNOWLEDGMENT

We gratefully acknowledge all the participants interviewed for this study, and we offer many thanks for the contributions to data collection by the staff of the Project Management Institute of Hohai University.

REFERENCES

- [1] Johnson M P, Sariöz D, Bar-Noy A, et al. 2012 More is more: the benefits of denser sensor deployment *ACM Transactions on Sensor Networks (TOSN)* 8(3), 22.
- [2] Ozturk C, Karaboga D, Gorkemli B 2012 Artificial bee colony algorithm for dynamic deployment of wireless sensor networks *Turk J Electr Eng Comput Sci* 20(2), 1-8.
- [3] Ozturk C, Karaboga D, Gorkemli B 2011 Probabilistic dynamic deployment of wireless sensor networks by artificial bee colony algorithm *Sensors* 11(6), 6056-65. <http://dx.doi.org/10.3390/s110606056>
- [4] Yick, Jennifer, Biswanath Mukherjee, and Dipak Ghosal 2008 Wireless sensor network survey *Computer networks* 52(12), 2292-2330. <http://dx.doi.org/10.1016/j.comnet.2008.04.002>
- [5] Younis, Mohamed, Kemal Akkaya 2008 Strategies and techniques for node placement in wireless sensor networks: A survey *Ad Hoc*

- Networks* 6(4), 621-55. <http://dx.doi.org/10.1016/j.adhoc.2007.05.003>
- [6] Gajbhiye, Pradnya, and Anjali Mahajan 2008 A survey of architecture and node deployment in wireless sensor network *Applications of Digital Information and Web Technologies* 10(1), 11-21.
- [7] Amundson, Isaac, Xenofon D. Koutsoukos 2009 A survey on localization for mobile wireless sensor networks *Mobile Entity Localization and Tracking in GPS-less Environments* 12(3), 235-254.
- [8] Wang, Bang, Hock Beng Lim, Di Ma 2009 A survey of movement strategies for improving network coverage in wireless sensor networks *Computer Communications* 32(13), 1427-36.
- [9] Liu Liping, Wang Zhi, Sun Youxian 2006 Survey on Coverage in Wireless Sensor Networks Deployment *Journal of Electronics & Information Technology* 28(9), 1752-56.
- [10] Akyildiz I F, Su W, Sankarasubramaniam Y, et al. 2002 Wireless sensor networks: A survey *Computer Networks* 38(4), 393-422. [http://dx.doi.org/10.1016/S1389-1286\(01\)00302-4](http://dx.doi.org/10.1016/S1389-1286(01)00302-4)
- [11] Heo, Nojeong, Pramod K. Varshney 2003 A distributed self spreading algorithm for mobile wireless sensor networks *Wireless Communications and Networking* 3(1), 23-30.
- [12] Gage Douglas W 1992 Command control for many-robot systems *Unmanned System* 10(4), 28-34.

AUTHORS

Ming Li is with the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, 210094, China (e-mail: lm@hhu.edu.cn).

HuanYan Qian is with the School of Computer Science and Technology, Nanjing University of Science and Technology, Nanjing, 210094, China (e-mail: liming76768@sina.com).

This research was supported by the Key Project of the National Social Science Foundation of China (No. 12AZD108); The rural water environment management based on the Geographic Information System(GIS). Submitted 09 October 2015. Published as resubmitted by the authors 15 December 2015.