

# A Novel Deployment Algorithm Based on Fluid Dynamics Approach

Ji-Guang Chen<sup>1</sup> and Huan-Yan Qian<sup>2</sup>

<sup>1,2</sup>*School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing, 210094, China*

<sup>1</sup>*jiguang\_chen01@126.com*, <sup>2</sup>*hyqian@njust.edu.cn*

## Abstract

*In this paper, we propose a deployment algorithm for mobile sensor networks, based on fluid dynamics in three-dimensional circumstance. In our algorithm, the entire wireless sensor network is viewed as a fluid body while nodes as charged particles. In the algorithm, nodes tend to spread out into the deployment area automatically, driven by the interaction force between nodes and nodes, nodes and obstacles, as well as nodes and boundaries. We achieve the desirable properties of self-adaption, robustness and simpleness, so that it is especially applicable to deployment in unknown environments. The performance indices of coverage and uniformity are evaluated by simulation. Simultaneously, we utilize digital satellite map nearby Midway Islands sea area to check the deployment effect in simulation, and the results show that our approach can guarantee a desired level of coverage. Moreover, the application of digital satellite map also provides a new approach for studying deployment of virtual reality in sensor networks.*

**Keywords:** *Sensor Deployment, Three-Dimensional Deployment, Fluid Dynamics, Mobile Sensor Networks*

## 1. Introduction

It's well known that deployment [1, 2] is the first problem to be solved among many researches on sensor networks. Recently, with the rapid development of mobile robotics [3, 4] industry, and cost reduction of sensor network nodes, the combination of the two facts has made mobile sensor networks more and more popular. Compared with the previous static sensor network, mobile sensor networks have a more flexible choice in deployment. For this reason, its deployment is becoming a new problem to be studied. Nowadays, deployment of mobile sensor networks has become a hot sport in researches, especially in three-dimension circumstances. 3D deployment of mobile sensor networks is so important that we are going to discuss it in this paper.

There have been some researches related to deployment issues of mobile sensor networks in three-dimensional space, while most of them are realized underwater and underground. In the literatures, the authors put forward their deployment approaches in a variety of environments, *e.g.*, D Pompili and IF Akyildiz studied the deployment issue of sensor networks under water in [7,8]. In their paper, deployment strategies for two-dimensional and three-dimensional communication architectures for underwater acoustic sensor networks are proposed, and a mathematical deployment analysis for both architectures is provided. The objective is to determine the minimum number of sensors to be deployed to achieve optimal sensing and communication coverage, which are dictated by application requirements. In another paper [9], the authors proposed a distributed node deployment scheme, which can help to increase the initial network coverage on an iterative basis. Assuming that the nodes

are initially deployed at the bottom of the water and can only move in vertical direction in a 3D space, the approach is to relocate the nodes at different depths based on a local agreement in order to reduce the sensing overlaps among the neighboring nodes. The nodes continue to adjust their depths until there is no room to improve the coverage.

In this paper, the deployment approach of fluid dynamics used is quite different from the previous ones. Thus, there are not so many relevant literatures involved. A fluid dynamics approach for multi-robot chemical plume tracing is addressed in [10], where flow variables of a 'real' fluid are measured by a computational sensor grid and the flow direction is estimated for backward tracing. Another work [11] proposes two gas models, one of which uses a virtual force approach and the other uses a kinetic approach. But the most relevant paper [12, 13] was provided by MR Pac and AM Erkmen, in which they made a fluid dynamics model for mobile sensor networks, while networks are viewed as fluid body, and nodes as particles in fluid. The deployment issues of nodes are turned into solving equations of fluid dynamics. Owing to the characteristics of self-diffusion of fluid, nodes being viewed as particles in fluid can spread to the whole deployment area automatically, which were called 'Self-Deployment' process by the author.

In the literatures mentioned above, studies on deployment have contributed to mobile sensor networks a lot in three-dimensional space. However, further studies are still necessary. Three-dimensional deployment approaches provided in paper [7-9] were explained only in theory, and they were so complex that they would be inappropriate in practice. We need a simple and practicable deployment approach with good adaptability, no matter underwater, underground or in the air. Literature [12, 13] give us some enlightenment on research, and they merely study two-dimensional deployment of mobile sensor networks. However, the power of fluid dynamics lies in three-dimensional applications, so we extend deployment approach to three-dimensional space, hoping to provide some inspiration for three-dimensional deployment in mobile sensor networks.

The rest part of this paper is organized as follows: in Section 2, we construct the fluid dynamics model of three-dimensional deployment for mobile sensor networks. In Section 3, we solve the model mentioned in Section 2 and establish the fluid dynamics algorithm of deployment. In Section 4, we simulate the fluid dynamics algorithm of deployment by using computer software and evaluate its performance. Finally, in Section 5, we reach the main conclusions.

## 2. Fluid Dynamics Model

CFD (Computational Fluid Dynamics), in one form or another, is based on the fundamental governing equations of fluid dynamics - continuity, momentum, and energy equations. For the details of fluid dynamics concepts discussed in this paper and derivations of governing equations, please refer to [14]. These equations are established based on three fundamental physical principles: conservation of mass, Newton's second law and conservation of energy. We regard the whole sensor network as a fluid body, and the mobile node as infinitesimal fluid element. What we should do is to control the movement of these infinitesimal fluid elements with fluid dynamics methods, so as to realize the deployment. In the equations mentioned above, momentum equation is the most suitable one to establish fluid dynamics model of mobile sensor networks, and it is based on Newton's second law of physics. In CFD, the control equation is also divided into two forms - conservation and non-conservation. Since infinitesimal fluid element (node) is movable, the non-conservative form would be adopted. Considering that, fluid itself is also comprised by viscous fluid and inviscid fluid, friction and other affects will also influence our deployment process. Thus, we

consider applying viscous fluid. In summary, the control equation we adopted is viscous non-conservative Neville-Stokes equation, as is shown in Eq. (1).

$$\left\{ \begin{array}{l} \text{x momentum: } \rho \frac{Du}{Dt} = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + \rho f_x \\ \text{y momentum: } \rho \frac{Dv}{Dt} = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + \rho f_y \\ \text{z momentum: } \rho \frac{Dw}{Dt} = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + \rho f_z \end{array} \right. \quad (1)$$

Equation (1) is called the Navier-Stokes equations in which  $\rho$  is fluid density,  $D/Dt$  is the substantial derivative,  $p$  is fluid pressure,  $f_x$ ,  $f_y$  and  $f_z$  are bodyforce components per unit mass in  $x$ ,  $y$  and  $z$  directions, and finally  $u$ ,  $v$  and  $w$  are the velocity components of an infinitesimal fluid element in the respective directions. The convention will be used here that  $\tau_{ij}$  denotes a stress in the  $j$  direction exerted on a plane perpendicular to the  $i$  axis.

It can be seen from Equation (1) that, this is a complicated equation set. Moreover, it is control equation related to fluid, and we are to simplify it, so as to make it more suitable for mobile sensor network deployment. The stress is divided into two parts: one is related to friction, while the other one is related to coefficient of viscosity. As for this, the problem is simplified and is more conform to practice. Friction is only connected with positive pressure. Assuming that the positive pressure along Direction  $X$  is  $F_x$  and friction coefficient as  $\mu_x$ . As for this, the friction of the node along Direction  $x$  shall be  $\mu_x F_x$ . Viscous resistance is in direct proportion to node speed, i.e. higher speed leads to larger viscous resistance. Assuming that the viscous coefficient is  $\lambda$ , and the coefficients along difference directions are the same, so that the viscous resistance along Direction  $x$  may be described as  $\lambda u$ . Similarly, the frictions and viscous resistances along Direction  $y$  and Direction  $z$  may as well be described as  $\mu_y F_y$ ,  $\lambda v$  and  $\mu_z F_z$ ,  $\lambda w$ . As for this, Equation (1) may be transformed into Equation (2):

$$\left\{ \begin{array}{l} \text{x momentum: } \frac{Du}{Dt} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \mu_x N_x + \lambda u + f_x \\ \text{y momentum: } \frac{Dv}{Dt} = -\frac{1}{\rho} \frac{\partial p}{\partial y} + \mu_y N_y + \lambda v + f_y \\ \text{z momentum: } \frac{Dw}{Dt} = -\frac{1}{\rho} \frac{\partial p}{\partial z} + \mu_z N_z + \lambda w + f_z \end{array} \right. \quad (2)$$

In the following, we are to analyze the physical meaning conveyed by Equation (2), as well as its connection with mobile sensor networks. The left side of the equation represents the changing rate of fluid particle's speed along with Time  $t$  in the direction, i.e. the acceleration, which is corresponding to the node's acceleration in the direction in the mobile sensor network. According to the definition in Literature [12],  $\rho$  is regarded as the Local Density, and its expression is (3):

$$\rho_i = 1 + \frac{R_s}{\bar{r}_{ij}} \times n_i = 1 + \frac{R_s n_i^2}{\sum_{j \in N_i} r_{ij}} \quad (3)$$

Note that  $\rho_i$  denotes the Local density at node  $i$ , and  $R_s$  is the sensing radius ( $0 < R_s < R_c$ ) of a deployment neighborhood  $N_i$  for node  $i$ , defined as the set of neighboring nodes  $j$  that fall within this range.  $n_i$  is the number of elements in this set and  $r_{ij}$  is the Euclidean distance between node  $i$  and node  $j$ . Here,  $R_c$  is the communication radius of node  $i$ . This definition

provides a density value that is proportional to the number of deployment neighbors and inversely proportional to the average distance between node  $i$  and its neighbors. Therefore, given the same number of elements in set  $N_i$ , local density would be higher when nodes are closely located. On the other hand, if set  $N_i$  is empty, density takes the value of 1 to account for node  $i$  itself.

In Eq. (2),  $P$  denotes the fluid pressure which is absence in mobile sensor networks for the nodes. It is difficult to get the actual pressure, especially when the deployment environment is changeable and hostile. What we can get in practical deployment is the distance or location information of nodes which can be obtained by communication signal strength and GPS devices respectively. It is more practical to deploy with obtained information that to discuss theoretically. Nodes may be considered as charged particles, and its pressure comes from the field stress imposed from other nodes. As for this, we may acquired Equation (4):

$$f_{ije} = k \frac{Q_i Q_j}{r_{ij}^2} \quad (4)$$

It's very familiar to us that Eq. (4) is called Coulomb's law in which  $f_{ije}$  denotes the electric force node  $i$  subject to node  $j$ ,  $k$  is Coulomb's constant,  $r_{ij}$  is the Euclidean distance between node  $i$  and node  $j$ , and finally  $Q_i$  and  $Q_j$  are the electric charges carried by node  $i$  and node  $j$  respectively.

Let's look back on Eq. (2), consider the  $x$  momentum and there is only  $f_x$  left which is called body forces in fluid dynamics. Body forces, which act directly on the volumetric mass of the fluid element. These forces "act at a distance"; examples are gravitational, electric, and magnetic forces. Hence, we can combine the forces  $f_{ije}$  and  $f_x$  together for they both are electric forces in our mobile sensor networks. So finally we obtain the equation like this:

$$\left\{ \begin{array}{l} \text{x momentum: } \frac{Du}{Dt} = \mu_x N_x + \lambda u + f_x \\ \text{y momentum: } \frac{Dv}{Dt} = \mu_y N_y + \lambda v + f_y \\ \text{z momentum: } \frac{Dw}{Dt} = \mu_z N_z + \lambda w + f_z \end{array} \right. \quad (5)$$

Focusing on  $x$  momentum in Eq. (5), note that  $Du/Dt$  is the time rate of change of velocity of the given fluid element as it moves through space along  $x$  direction,  $\mu_x N_x$  is the Friction force and  $\lambda u$  is the viscous resistance, and  $f_x$  is the net force of gravitational and electric forces. Eq. (5) is also called Governing Equations, then we discuss the Limitations, Static Equilibrium, Initial and Boundary Conditions needed in order to solve the Eq. (5).

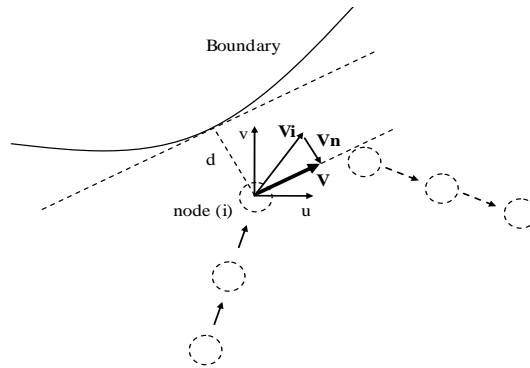
The solutions obtained from Eq. (5) for the velocity components may exceed the locomotion capabilities of sensor nodes. Not only velocity, but also its time derivative, *i.e.*, acceleration, may exhibit a similar behavior. Therefore, we put hard-limiters to the magnitudes of both velocity and acceleration of sensor nodes in case they exceed respective thresholds  $V_{th}$  and  $a_{th}$ .

Nodes are in motion within the region of deployment due to the Coulomb's force, but the process will not carry on forever. When all the nodes are spreading all over the region, the Coulomb's force between them decreases as the time proceeds. When the resultant force on a node is 0, it will lose its acceleration and tend to stop moving. Therefore, static equilibrium is reached. The major factors that influence time and energy consumption for nodes to reach

equilibrium are initial positions and status of node, which are also in correlation with the damping coefficient.

The above argument rests on the assumption that the environment is static. When the environment of deployment is changing, nodes that had already reached equilibrium may need to re-deploy themselves in order to reach new equilibrium. This shows the self-adaption of this algorithm. However, if the deployment environment is changeable, nodes may only reach temporary equilibrium. The equilibrium will break with the environment changes. As for this, furthering order to achieve re-balance, energy will be consumed.

Normally, region of deployment has a certain size. Nodes should be deployed within the boundaries of the region, instead of exceeding them. The boundaries can be physical (e.g., mountains, rivers), or be hypothetical (e.g., latitude and longitude lines). When nodes are moving towards the boundaries, a corresponding approach is required to ensure the nodes won't go over the boundaries, which is known as boundary conditions, as is shown in Figure 3.



**Figure 1. Illustration of the Boundary Condition**

When the distance  $d$  between nodes and the boundary is less than the sensing radius of Node  $R_s$ , the node will change the direction of the resultant force due to Coulomb's force exerted by the boundary, resulting in a change in velocity due to superposition of the original velocity  $V_i$  and vertical boundary velocity  $V_n$ , creating a new velocity  $V_t$ . The change in direction of  $V_t$  will eventually drive the node away from the boundary. Therefore there will always be a safe distance between nodes and boundaries.

### 3. Deployment Algorithm based on Fluid Dynamics Model

Supposing that there is a three-dimensional region  $\Omega$  that needs to be deployed with sensor nodes in order to monitor the region, how to deploy a certain number of sensor nodes to meet the expected deployment results? We hereby present a concept, known as Coverage Ratio. According to the definition in article [15], the formula for Coverage Ratio is shown in Eq. (6):

$$C(R_c) = \frac{4\pi R_c^3 N}{3\Omega} \quad (6)$$

In the above equation,  $N$  stands for the total number of nodes and  $\Omega$  is the volume of the deployment region,  $R_c$  is the communication radius of sensor nodes, and  $C$  is the Coverage Ratio of the region. It is shown in Eq. (6) that, Coverage Ratio  $C$  stands for the ratio between the sum of communication coverage volume of all nodes (total number of  $N$ ) and the volume of deployment region  $\Omega$ . The value of  $C$  indicates the communication coverage of the region.

A larger Coverage Ratio is capable of providing the network with better connectivity and higher service quality. If the expected value of  $C$  is given, we can easily calculate the minimum number of nodes needed, denoted as  $N$  above, provided that the volume of the region is already known. When actually deploying sensors with this algorithm, it is unnecessary to know an exact volume of the region  $\Omega$ . Instead, we only need an approximation of the volume  $\Omega$ , so as to compute an approximate number of nodes needed.

When deploying nodes, the initial position of nodes might be randomly or artificially set. In both cases, the process of deployment will not be affected. Starting from initial time  $t_0$ , nodes are moving under the influence of fluid. The forces exerted on nodes may derive from other nodes, obstacles, or boundaries, and may also be affected by friction resistance  $F$  and viscous resistance  $F_d$ . These forces will form a resultant force and exert on the node. For this reason, the node will move along the direction of the resultant force, and reaches equilibrium eventually. When all the nodes reach equilibrium, the deployment process is hereby completed.

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 various reasons. Disabled nodes may lead to the loss of monitoring coverage in certain regions. When this situation occurs, equilibrium reached based on the deployment algorithm of fluid dynamics may break, so that in turn, all nodes will be forced to relocate themselves to cover these “exposed” areas, reaching new equilibrium.

The entire Deployment Algorithm of Fluid Dynamics is shown in Figure 2.

---

**Algorithm : Deployment Algorithm of Fluid Dynamics**

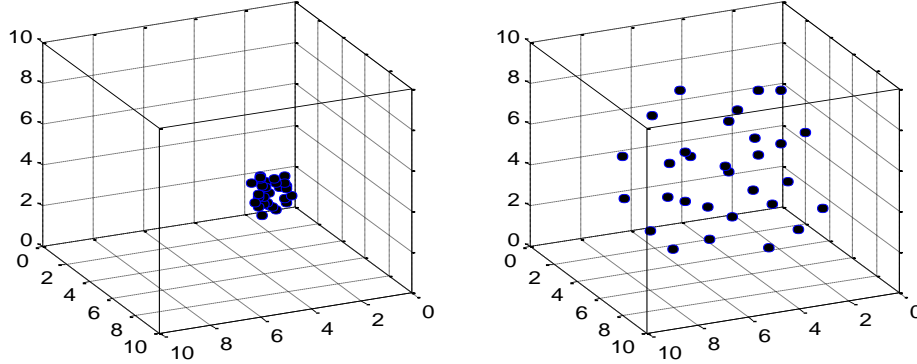
**Requirement:**  $\Omega$ ,  $N$ ,  $C$ ,  $R_c$ ,  $R_s$ .

- 1: Estimate number of nodes  $N$  based on expected Coverage Ratio  $C$  and volume of  $\Omega$
  - 2: At time  $t_0$ , all of the  $N$  nodes are located at their initial positions
  - 3: For each node  $node\_i$  at time  $t$
  - 4: Compute the Coulomb's force  $F_e$  exerted by other nodes in the sensing radius  $R_s$
  - 5: Compute the Coulomb's force  $F_o$  exerted by obstacles in the sensing radius  $R_s$
  - 6: Compute the friction force  $F$  on the node
  - 7: Compute the damping force  $F_d$  on the node
  - 8: Compute the net force on node  $i$ :  $F_i = F_e + F_o + F + F_d$
  - 9: Compute the acceleration  $a_i$  of node  $i$  at time  $t$
  - 10: Compute the velocity  $v_i$  of node  $i$  at time  $t$
  - 11: Determine the relationship between threshold value  $a_{th}$  &  $v_{th}$  and acceleration  $a_i$  and velocity  $v_i$  respectively, and then proceed accordingly.
  - 12: Compute the position  $P_i(t)$  of node  $i$  at time  $t$ , and move the node to this position
  - 13: Determine if node  $i$  reached equilibrium at time  $t$ . Terminate node's motion if yes; otherwise the loop executes again.
  - 14: End For
  - 15: All nodes have reached equilibrium, algorithm ends.
- 

**Figure 2. Deployment Algorithm of Fluid Dynamics**

With a 3D plot, the principle of the algorithm in this paper can be clearly demonstrated, as is shown in Figure 3. This is a 3D plot simulated with software, and the deployment area  $\Omega$  is a space with the dimension of  $10 \times 10 \times 10$ . The initial deployment position of node is a corner

in the area, as is shown in the left plot. The small black dots are exactly the mobile nodes to be deployed. The right plot shows the spatial distribution of nodes after deployment. It can be seen from the plot that, nodes have been expanded from the initial positions to the whole space.



**Figure 3. Demonstration of the Deployment Process**

#### 4. Simulation Results

Numerous simulations have been carried out in various environments to investigate the performance of our approach. Parameter settings of our simulations are indicated in Table 1 with reference to the related figure number.

**Table 1. Simulation Settings by Figure Number**

Parameter	Fig. 4	Fig. 5	Fig. 10
Deployment ( $\Omega$ )	10×10×10	10×10×10	8000×5000×2000 (m)
Node number (NN)	100	100	100
Rs, Rc	2; 4	2; 4	200; 400
damping factor ( $D_f$ )	2.0	2.0	0.6
$a_{th}, v_{th}$	2; 1	2; 1	2; 1
Time (T)	15	15	20
$\Delta t$	0.1	0.1	0.2
Obstacles	0	1	seabed

##### 4.1. Coverage and Uniformity

Generally, coverage can be considered as the service quality of a sensor network. Gage invented the concept of coverage in the research of multi-robot systems [16]. This paper defines it as the ratio between the sum of the coverage volume of all nodes and the volume of the entire target region, as is shown in Equation 7. The definition of the sum of the coverage volume is taken from the concept of union in the Set Theory, thus the coverage is usually no larger than 1.

$$Coverage = \frac{\bigcup_{i=1, \dots, N} \Omega_i}{\Omega} \quad (7)$$

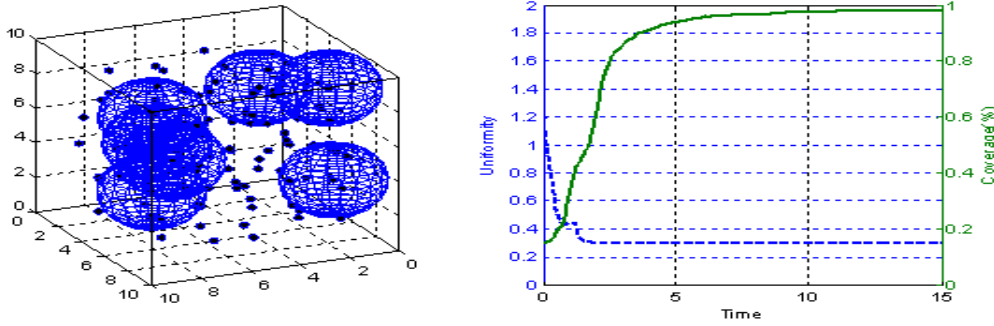
The uniformity of coverage is a well-defined standard to measure the service life of a network. Article [15] describes the concept as the standard deviation of distance between nodes. Smaller standard deviation means better coverage uniformity of the network. However, this approach for measuring uniformity is under perfection. We take the grid approach to calculate uniformity in this paper, *i.e.*, dividing the whole region into N small cubes with the same volume, and then figuring out the standard deviation of the nodes contained in these small cubes, as is shown in Equation 8.

$$U = \left[ \frac{1}{N} \sum_{i=1}^N (n_i - \bar{n})^2 \right]^{\frac{1}{2}} \quad (8)$$

In the equation,  $n_i$  is the total number of nodes in the  $i^{\text{th}}$  small cube and  $\bar{n}$  is the mean of nodes number in each cube. So far, we have discussed the relation between communication and coverage. Article [17] has proved that when the communication range of node is twice or larger than the sensing range, coverage will contain pure connections. In practical deployment, we only have to consider the coverage so as to ensure the connection. At the moment, coverage contains connection problems.

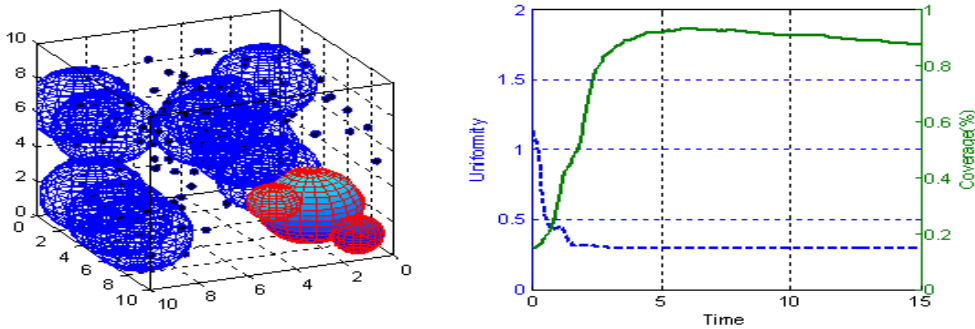
Please refer to Table 1 for simulation parameters in Figure 4. The circumstance after deployment is shown in the left plot. In the plot, the cube represents the volume of the region (area  $\Omega$ ). Tiny black dots represent the positions of nodes; blue spheres are used to indicate the sensing radius. In order to watch them clearly, only the sensing radius of partial nodes are displayed. Nodes can probe the environment and collect information in its sensing radius  $R_s$ . Similarly, it may be influenced by the Coulomb's force from its adjacent nodes, and repulsive force from the obstacles as well. The communication radius of Node  $R_c$  is greater than the sensing radius  $R_s$ . Nodes are able to exchange information mutually within the communication radius. The initial positions of the nodes are in the center of Region  $\Omega$  (being close to the coordinates [5, 5, 5]). Imaginably, node density at the beginning is practically very large. When deployment is started, all the nodes are forced by un-balanced Coulomb's force, starting to move to fill the entire region, and eventually reaching equilibrium, as is shown in Figure 4. The right plot shows the change in Coverage and Uniformity versus time during the deployment process described in the left plot. The x-coordinate is the simulation time T; the y-coordinate on the left is the uniformity; the y-coordinate on the right represents the coverage. We can see from the right plot that, at the initial moment of deployment, the nodes only cover a small part of the region because they are all initially positioned in the center of  $\Omega$ . Therefore the coverage is practically low (<20%). With time elapsed and all nodes are in motion, the coverage tends to grow, and reaches equilibrium after a certain moment (about Time=9). This begins when the coverage reaches maximum (about 100%), and lasts until the simulation is over. The value of uniformity is initially around 1.8 when the simulation begins. Then it decreases rapidly shortly after the deployment begins, eventually, tends to be around 0.3. Because uniformity represents standard deviations, so the smaller it is, the more uniform the network is.





**Figure 4. Normal Deployment**

In the left plot in Figure 5, the solid spheres represent obstacles. In the previous sections we mentioned that obstacle can be considered stationary charge with same sign. But the charge it carries is proportional to its size, which means the larger it is, the more charge it carries. There is only one obstacle in the left plot. We can adjust the distance from nodes to the obstacles by configuring the amount of charges that the obstacle carries. It is shown in the left plot that, due to the repulsive force from the obstacles, nodes will be deployed at a distance with the obstacle to avoid them. This is highly meaningful in applications. To keep the nodes away from hazard or unreachable region can minimize node damage, indicating the algorithm's self-adaption. The right plot shows the change in coverage and uniformity versus time during the deployment process. We can see that because of the obstacles, the region is not completely covered by nodes. The maximum coverage is less than 1. In the meantime, compared with Figure 4, the value of uniformity is higher after nodes have reached equilibrium, due to reduction of uniformity because of obstacles.



**Figure 5. Deployment with Obstacles**

#### 4.2. Virtual Deployment on Digital Map

According to the application scope of sensor networks, there are plenty of them applied in fields like environment monitoring, wild animal tracking, military target detection, *etc.* Normally, these application circumstances are located in the open air. Nowadays, the development of digital satellite map has provided us with a new chance, *i.e.*, taking advantage of these high definition maps to simulate before practical deployment, so as to provide reference for real deployment.

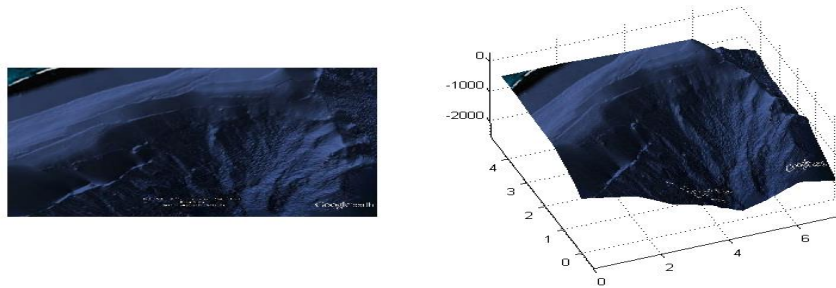
Many government and commercial organizations, such as NASA, ESA, USGS, NOAA, ESRI, and Microsoft, adhere to the WMS protocol for rendering, reprojecting, and serving georeferenced data sets over the Internet. That enables we to access elevation, oceanography,

weather, satellite imagery, and many other raster data sets conveniently. Here we use the Google Earth to accomplish our work.



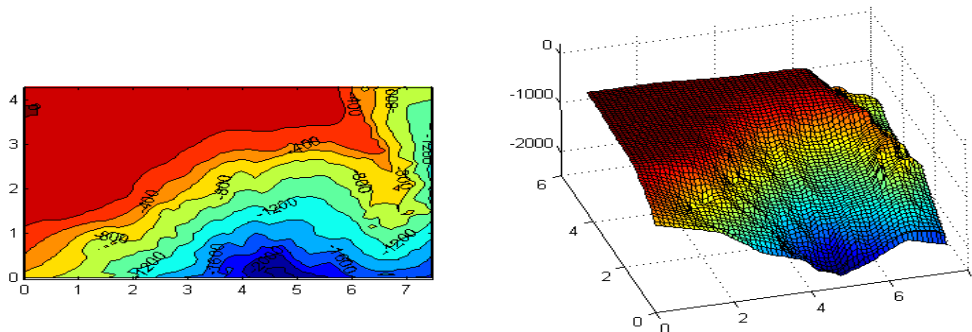
**Figure 6. Deployment Area in Google Earth**

We selected an island nearby Midway Islands in the Pacific Ocean as the deployment area, as is shown in the left plot of Figure 6. The latitude and longitude of the island is (27°51'42"N, 175°52'21"W). We take the southern beach of the island as the precision deployment area, as is shown in the right plot of Figure 6.



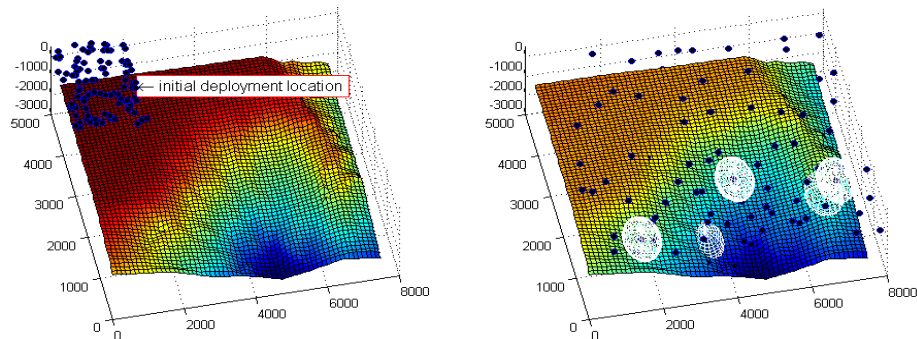
**Figure 7. Deployment Area (Left Plot) and its Three-dimensional Scene (Right Plot)**

The left plot in Figure 7 is the satellite map of this deployment obtained via Google Map. The right plot is the 3D map of this sea area. It is shown in the figure that, the length of the deployment area is 8km, and width is 5km. It is a slope from sea level to 3km below the sea level. We deploy 100 mobile sensor nodes in this area, and these nodes will be automatically deployed according to the algorithm presented previously, so as to monitor the environment of this area. The detailed deployment parameters are shown in Table 1.



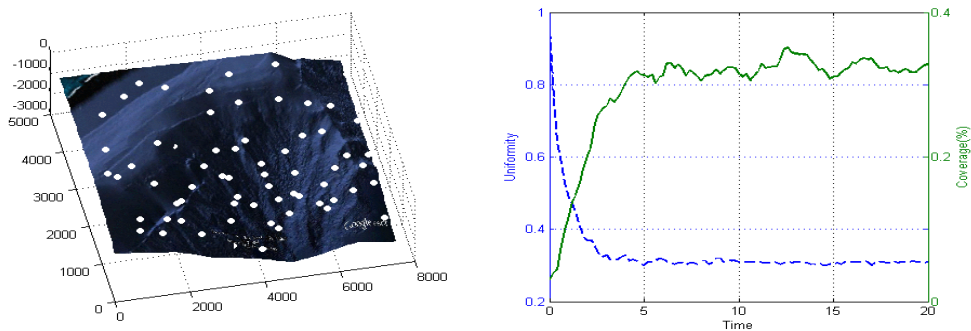
**Figure 8. Contour Map (Left Plot) and Three-dimensional Scene (Right Plot) of the Deployment Area**

In order to better understand the landform of the area, we listed the contour map of the area (Left Plot in Figure 8) and the three-dimensional map (Right Plot in Figure 8) simulated with software. The plots may be analyzed by referring to Figure 7.



**Figure 9. Initial Position (Left Plot) and Final Position (Right Plot) of Nodes**

After having obtained the geographic information of the sea area, we are going to utilize the information to realize the deployment algorithm, as is shown in the left plot of Figure 9. Considering the practical deployment situation, we set the initial position of mobile nodes in the area along the beach, for the convenience of implementation. At the very beginning, the density of nodes is high. According to our algorithm, these nodes will “flow” to other areas, and finally fill the deployment area, as is shown in the right plot of Figure 9. In order to better demonstrate the deployment of nodes, we select some nodes, and draw their sensing radius – the white spheres in the figure.



**Figure 10. Final Deployment Scene (Left Plot) and Performance Curve (Right Plot)**

The left plot in Figure 10 is the real scene of after deployment. In the figure, the white dots represent nodes. It is shown in the figure that, these nodes fill the entire sea area, avoiding obstacles at the bottom. The right plot shows the changing curves of coverage and uniformity during the whole deployment process. It is shown by the figure that, the coverage increases from the initial 0.05 to the final approximately 0.3. The volume of the entire sea area is length×width×depth. The uniformity decreases approximately from the initial 0.9 to the final 0.3. Lower uniformity value means better uniformity performance. It can be seen that, at the beginning, the uniformity of nodes decreases quickly. Dense nodes spread in all directions, reducing the uniformity. With nodes move slowly, the uniformity tends to be stable.

## 5. Conclusion

Most previous researches on sensor networks are limited to 2d planes, neglecting the vertical fall of the deployment area, while taking the area as a flat plane. With people's continual exploration on space, underwater, underground and other regions, 3D sensor networks are gradually becoming a research hot spot. Yet, 2D algorithms are no longer applicable to 3D deployment. As for this, deployment algorithms under 3D environment have to be researched. Up to now, fluid mechanics has become quite mature. In this paper, mobile sensor networks are considered as the fluid, while nodes are regarded as charged particles in the fluid. Particles move along with the fluid. In the meanwhile, a mutual interaction forces between particles, between particle and obstacle, between particle and boundary is created. The result of resultant force is to drive nodes to complete the deployment. The algorithm presented in this paper no longer needs to conduct advance exploration on the deployment area, with features like self-adaption, robustness, and simple-ness. Thus, the algorithm is applicable to deployment in unexplored area. In the simulation part of the paper, the indexes of coverage and uniformity are selected to assess the performance of the algorithm. Previous deployment algorithms basically take advantage of simulation software to realize the simulation process, and they seldom utilize material object to test the deployment on the spot. Thus, these algorithms have considerable limit, and are different from practical deployment. By contrast, the paper makes use of high precision digital satellite map to realize virtual reality deployment of nodes. On this basis, the paper proposes a new research ideology. In the simulation part, deployment is realized in the sea area of Midway Island. According to the simulation result, the deployment result of the algorithm coincides with the prediction, providing reference for field deployment of mobile sensor network.

## Acknowledgments

Project supported by the College and University Natural Science Foundation of the Ministry of Education, Jiangsu (13KJD52004, 12KJD510006).

## References

- [1] M. Younis and K. Akkaya, "Strategies and techniques for node placement in wireless sensor networks: A survey", *Ad Hoc Networks*, vol. 6, no. 4, (2008), pp. 621-655.
- [2] P. Gajbhiye and A. Mahajan, "A survey of architecture and node deployment in wireless sensor network", *Applications of Digital Information and Web Technologies*, 2008. ICADIWT 2008. First International Conference, IEEE, (2008).
- [3] M.-H. Chen, "On the development of autonomously manipulation of group mobile robots for smart living and biomimetic applications", *Computer Science and Automation Engineering (CSAE)*, 2012 IEEE International Conference, IEEE, vol. 2, (2012).
- [4] C.-Y. Chen, "The development of autonomous low-cost biped mobile surveillance robot by intelligent bricks", *Journal of Vibration and Control*, vol. 18, no. 5, (2012), pp. 577-586.
- [5] T.-C. Shih, S.-S. Yeh and P.-L. Hsu, "Development of a Behavior-Based Cooperative Search Strategy for Distributed Autonomous Mobile Robots Using Zigbee Wireless Sensor Network", *Asian Journal of Control*, (2013).
- [6] X. Li, "Servicing wireless sensor networks by mobile robots", *Communications Magazine, IEEE*, vol. 50, no. 7, (2012), pp. 147-154.
- [7] D. Pompili, T. Melodia and I. F. Akyildiz, "Deployment analysis in underwater acoustic wireless sensor networks", *Proceedings of the 1st ACM international workshop on Underwater networks*. ACM, (2006).
- [8] D. Pompili, T. Melodia and I. F. Akyildiz, "Three-dimensional and two-dimensional deployment analysis for underwater acoustic sensor networks", *Ad Hoc Networks*, vol. 7, no. 4, (2009), pp. 778-790.
- [9] K. Akkaya and A. Newell, "Self-deployment of sensors for maximized coverage in underwater acoustic sensor networks", *Computer Communications*, vol. 32, no. 7, (2009), pp. 1233-1244.

- [10] D. Zarzhitsky, "A fluid dynamics approach to multi-robot chemical plume tracing", Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems, IEEE Computer Society, vol. 3, (2004).
- [11] W. Kerr, "Two formal gas models for multi-agent sweeping and obstacle avoidance", Formal Approaches to Agent-Based Systems. Springer Berlin Heidelberg, (2005), pp. 111-130.
- [12] M. R. Pac, A. M. Erkmén and I. Erkmén, "Scalable self-deployment of mobile sensor networks: A fluid dynamics approach", Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference, IEEE, (2006).
- [13] M. R. Pac, A. D. M. Erkmén and I. Erkmén, "Towards fluent sensor networks: A scalable and robust self-deployment approach", Adaptive Hardware and Systems, 2006. AHS 2006, First NASA/ESA Conference, IEEE, (2006).
- [14] J. D. Anderson, "Computational Fluid Dynamics the Basics with Applications", McGraw-Hill, Inc., (1995).
- [15] N. Heo and P. K. Varshney, "A distributed self spreading algorithm for mobile wireless sensor networks", Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE, IEEE, vol. 3, (2003).
- [16] D. W. Gage, "Command control for many-robot systems", Naval Command Control and Ocean Surveillance Center RDT and E DIV San Diego, CA, (1992).
- [17] S. M. Alam and Z. J. Haas, "Coverage and connectivity in three-dimensional networks", Proceedings of the 12th annual international conference on Mobile computing and networking. ACM, (2006).

### Author



**Ji-Guang Chen**, received his B.S. and M.S. degrees in computer science and technology from Henan Normal University, Xinxiang, Henan, China, in 2004 and 2008. Currently, He is pursuing the Ph.D. degree in school of Computer Science and Engineering, Nanjing University of Science and Technology. He is interested in sensor deployment and localization in mobile sensor networks and wireless sensor networks.



**Huan-Yan Qian**, received his B.S. degree in Computational mathematics from Nanjing University in 1977. Now he is Professor and PhD supervisor in school of Computer Science and Engineering, Nanjing University of Science and Technology. His scientific interests include sensor networks, mobile communication and wireless communication Networks.

