A DEPLOYMENT ALGORITHM FOR MOBILE WIRELESS SENSOR NETWORKS BASED ON THE ELECTROSTATIC FIELD THEORY

Jiang Xu[1,2,*], Huanyan Qian[1], Wenhao Ying[2] and Jun Zhang[3]

[1]School of Computer Science and Engineering, Nanjing University of Science and Technology, Nanjing210094, China
[2]School of Computer Science and Engineering, Changshu Institute of Science and Technology, Changshu215500, China
[3]Department of Coastal Defense Artillery, Shenyang Artillery Academy, Shenyang110161, China
Email:skydestinyx@gmail.com, [*]Corresponding Author

Submitted: Oct. 15, 2014 ***Accepted: Jan. 23, 2015*** ***Published: Mar. 1, 2015***

Abstract- This paper proposes a deployment algorithm based on the electrostatic field theory for mobile wireless sensor networks. The nodes and obstacles in the deployment area are taken as the charged particles; and the particles will move due to the Coulomb's force from other particles or obstacles. Finally, the nodes automatically spread to the whole area by the resultant action and complete the deployment. Four metrics, including coverage, uniformity, deployment time and average displacement distance, are used to evaluate the performance of the algorithm. The simulation results show that the proposed algorithm can give full play to its self-adaptive advantages and achieve the desired deployment effect; it is a kind of deployment algorithm with self-adaptive characteristics.

Index terms*:* Mobile Wireless Sensor Networks, Electrostatic field, Deployment algorithm.

I. INTRODUCTION

A large number of research has been done to solve problems encountered by mobile wireless sensor network deployment [1-5]. Solutions in practice can be divided into three categories: Methods based on geometrical model [6-8], Methods based on virtual potential field [9-10], and Methods based on biological intelligence [11-13]. The algorithm proposed in this paper belongs to the second one, since the electrostatic field is a kind of virtual potential field.

The most classical literature regarding Methods based on virtual potential field was proposed by Howard, Mataric, and Sukhatme [14]. They provided a solution to the problem of deploying a mobile sensor network in unknown dynamic environments and described a potential-field-based deployment approach, in which nodes are treated as virtual particles that are subject to virtual forces. These forces repel nodes from each other and obstacles, and ensure that nodes can spread out from an initial compact configuration till the coverage area of the network is maximized. In addition, nodes are also subject to a viscous friction force. This force is used to make sure that the network will eventually reach the state of static equilibrium, i.e. all nodes will ultimately stop moving. Similarly, algorithms proposed by [15] and [16] use both repulsive and attractive force components to maximize coverage and uniformity for a given number of sensors.

Recently, deploying nodes of mobile sensor network using virtual potential field is attracting more and more researchers' attention. In [18], the authors proposed an approach and in which Delaunay triangulation is formed with nodes. If two nodes are connected in the Delaunay diagram, they are said to be adjacent. Force can only be exerted from adjacent nodes within the communication range. Simulation results showed that the approach has higher coverage rate and less convergence time than traditional virtual force algorithm. In Ref. [19], Jun Li and his companions tried to solve the connectivity and overlap problems faced in the traditional virtual force algorithm (VFA); and then they developed an extended virtual force-based approach to achieve the ideal deployment result. Simulation results show that the virtual force approach can effectively reach the ideal deployment in the mobile sensor networks with different ratio of communication range to sensing range. Furthermore, it gets better performance in coverage rate, distance uniformity and connectivity uniformity than prior VFA. In the paper [20], they present an energy-efficient and self-deployment scheme to utilize the attractive force generated from the centroid of a sensor's local Voronoi polygon as well as the repulsive force frequently used in

self-deployment schemes using the potential field. The simulation results show that their scheme can achieve a higher coverage and enable less sensor movements in shorter times than self-deployment schemes using the traditional potential field.

In [17], Toumpis and Leandros assumed working in such a scene: We can freely place the delay nodes in the deployment area, however, the source and sink nodes are fixed. Then they put forward a question: How to place nodes so that the number of nodes needed to support the traffic is minimum? Obviously, we can use regular ways to solve this problem, but we might meet more troubles and spend more time. However, their solution is simpler and quicker with the use of electrostatic field, which shows the advantages of introducing electrostatic field in mobile sensor networks.

The deployment algorithms mentioned in the above literatures are all Methods based on virtual potential field. They share something in common that the deployment of mobile sensor nodes is considered as a coverage process. Nodes are affected by virtual force and scattered themselves all over the deployment area; eventually, these nodes reach equilibrium state and finish the deployment of the entire network. Although these algorithms all assume that the deployment environment is unknown and they are adaptive themselves, they can complete the task without knowing the environment. However, they all imply one condition that the size of the deployment region and the number of nodes are known.

However, if the size of the deployment environment is unknown beforehand, these algorithms can only provide coverage to the size extent of the area which is previously fixed by the number of nodes to be deployed. Thus, a certain quality of service could not be guaranteed with these approaches, which constrain the adaptivity and scalability of the deployment algorithms.

This paper provides the approximate size of the deployment region, the algorithm can compute the number of nodes needed by setting appropriate parameters, thereby complete the deployment. The main contributions are as follows: Firstly, we present an adaptive deployment strategy that can guarantee a good coverage and uniformity. Secondly, we can finish the deployment work under the condition that only part of the deployment environment is known.

II. PRELIMINARIES: ELECTROSTATIC FIELD

Potential fields are a commonly used and well understood method in mobile robotics, where they are typically applied to tasks such as local navigation and obstacle avoidance. In our work, we apply potential fields to the deployment problem. Consider the following scene: In vacuum, two point charges of the same sign are Q_1 and Q_2, the distance between them is $2a$, as is shown in Figure 1:

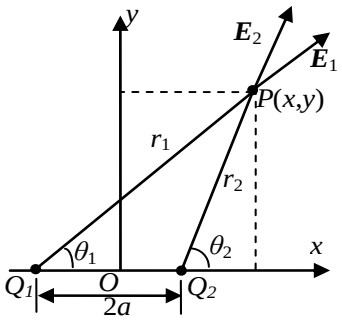

Figure 1. A pair of point charges

The electric potential at point $P(x, y)$ yielded by Q_1 and Q_2 is shown in Equation 1, r_1 and r_2 stands for the Euclidean distance from P to Q_1 and Q_2, k is Coulomb's Constant.

$$U = U_1 + U_2 = \frac{kQ_1}{r_1} + \frac{kQ_2}{r_2} \tag{1}$$

As for point charges, according to the definition in fundamental field theory, the electric-field intensity can be computed via its potential gradient,

$$E = -\nabla U \tag{2}$$

In the Equation, ∇ is gradient operator, as is shown in Equation 3.

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

Now we are to generalize the situation shown in Figure 1. Assume all the sensor nodes and obstacles in the mobile sensor network are considered as point charges. U_n and U_o respectively stands for the superimposed electric potential field at P, induced by all the nodes and obstacles, then the potential at P can be expressed as $U_n + U_o$. Imagine a single node N_i in the mobile sensor

network as a point charge with charge q, then the electric force F_i exerted on this node at point P in Figure 1 can be expressed in Equation 4:

$$F_i = Eq = -\nabla(U_n + U_o)q \tag{4}$$

It can be easily inferred by Newton's Second Law, node N_i will accelerate and start moving due to the electric force F_i.

III. DEPLOYMENT MODEL

a. Equations of Control

Nodes are subject to resistance from the environment in actual deployment process. Suppose the damping factor is μ, the resistance is proportional to its velocity v_i, and mass of the node is m, then the acceleration of the node can be computed by Equation 5:

$$\alpha_i = (F_i - \mu v_i)/m \tag{5}$$

Suppose the initial time of deployment is t_0, the initial velocity of all nodes is v0and $v_0 = 0$, and the initial position of a certain node N_i is $P_i(t_0)$, then the electric force exerted on node N_i can be computed via Equation 4, therefore the acceleration $\alpha_i(t_0)$ at moment t_0 is shown in Equation 6.

$$\alpha_i(t_0) = F(t_0)/m \tag{6}$$

t_1 is reached after any infinitesimal time interval t, if t is short enough, we can assume N_i is moving with uniform acceleration in infinitesimal time interval t, then its velocity $v_i(t_1)$ at moment t_1 can be expressed as

$$v_i(t_1) = v_i(t_0) + \alpha_i(t_0)\Delta t = \alpha_i(t_0)\Delta t \tag{7}$$

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

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

When node N_i reaches its new position $P_i(t_1)$, its acceleration $\alpha_i(t_1)$ at moment t_1 needs to be re-calculated based on electric force and resistance exerted on the new position. Therefore the position, velocity, and acceleration of node N_i at moment t_1 can be computed. And so on, the

position, velocity, and acceleration of node N_i at any moment t_n can be computed. Further generalization shows that the position, velocity, and acceleration of all the nodes at any moment t_n can be computed.

Considering that in practical application, the velocity and acceleration of nodes are finite, two thresholds vth and ath are given to limit node's velocity and acceleration. They will not exceed the thresholds.

b. Boundary and Equilibrium

Normally the 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), they can also be hypothetical (e.g. latitude and longitude lines). When nodes are moving towards or near the boundaries, we need a corresponding approach to ensure that the nodes won't exceed boundaries, which are known as boundary conditions, as is shown in Figure 2.

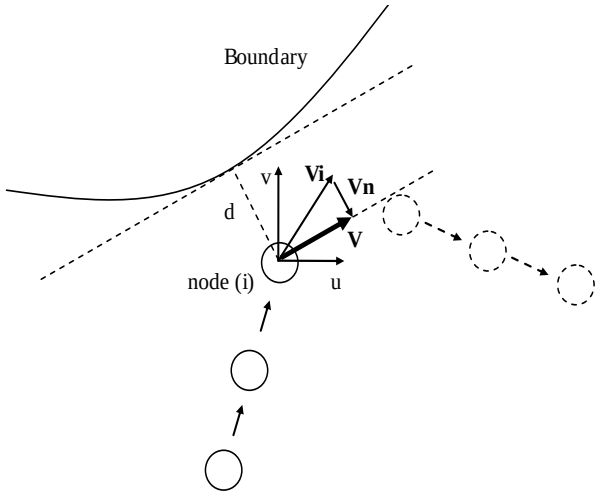

Figure 2. Illustration of the boundary condition

Suppose the boundary of the deployment region is consisted of a large number of charges with the same sign, when the distance d between node and the boundary is less than the sensing radius of the nodes R_s, node will change the direction of the net force due to Coulomb's force exerted by the boundary, as a result, a change will take place in velocity due to the superposition of the original velocity V_i and vertical boundary velocity V_i, then it will produce a new velocity V, the change in direction of V will eventually drive the node away from the boundary. Therefore there will always be a safe distance between the node and the boundary.

Nodes are moving within the region of deployment due to the Coulomb's force, but the process will not go 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 net force on a node is close to 0, it will lose its acceleration and tend to stop moving, therefore equilibrium is reached. The major factors that influence the time and energy consumption for nodes to reach equilibrium are initial positions and conditions of the nodes.

The above argument relies on the assumption that the environment is static. When the environment of deployment is changed, nodes that had already reached equilibrium may need to re-deploy themselves in order to reach new equilibrium. It shows the adaptivity of this algorithm. However, if the environment of deployment is changeable, then nodes may only reach temporary equilibrium. The equilibrium will be broken due to the change in the environment, which requires further re-balance, while on the other hand, more energy is wasted.

IV. ADAPTIVE SELF-SPREADING ALGORITHM

Suppose there is a region A that needs to be deployed with sensor nodes in order to monitor the region. How to deploy a certain number of sensor nodes to reach the expected deployment results? We hereby present a concept, known as Coverage Ratio. According to the definition from article [21], the formula for Coverage Ratio is shown in Equation 9:

$$C(R_c) = \frac{\pi R_c^2 N}{A} \tag{9}$$

In the above equation, N stands for the total number of nodes, A stands for the area of the region of deployment, R_c is the communication radius of sensor nodes, and C is the Coverage Ratio of the region. From Equation 9, Coverage Ratio C stands for the ratio between the sum of communication coverage area of all the nodes (total number of N) and the area of region of deployment A .The value of C indicates the communication coverage of the region. A larger Coverage Ratio is capable of providing the network better connectivity and higher service quality. Conversely, given the expected value of C , we can easily calculate minimum number of nodes needed, denoted as N above, provided that the area of the region is known. When we actually deploy sensors using this algorithm, an exact size of the region is not required. Instead, we only

need an approximation of the area A, in order to compute an approximate number of nodes needed.

As is shown in Figure 3, the black circle represents the starting position of the node; the red circle represents the ending position of the node; the green circle represents the perception range of the node. When the nodes are deployed, the starting position of the node may be random, as is shown in the left figure. However, this has no impact on the deployment process of the nodes. From the starting moment, the node starts to move by the action of the electric field force, which may come from other nodes, obstacles, or boundaries, and may suffer from the viscous resistance. By the action of these forces, the node will move along with the direction of the resultant force, and reach an equilibrium state eventually. When all the nodes reach the equilibrium state position, the deployment process ends accordingly.

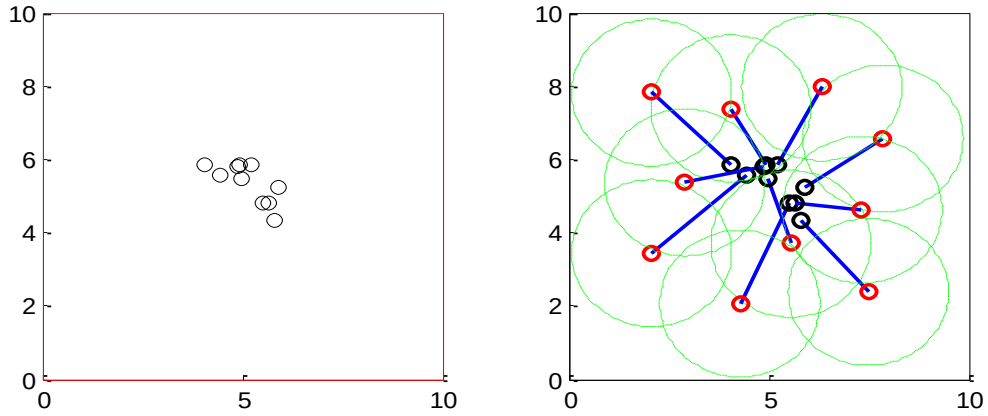

Figure 3. The process of deployment

In 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, equilibrium reached by using Adaptive Self-Spreading Algorithm will break so that in turn, all nodes relocate themselves to cover these “exposed” areas, thus reaching a new equilibrium.

The entire Adaptive Self-Spreading Algorithm is shown in Figure 4.

Requirement: A, N, C, Rc, Rs, m.
1: Estimate the number of nodes N based on expected Coverage Ratio C and area of the region A
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 $\boldsymbol{F}_e$ exerted by other nodes in the sensing radius Rs
5: Compute the Coulomb's force $\boldsymbol{F}_e$ exerted by obstacles in the sensing radius Rs
6: Compute the damping force $\boldsymbol{F}_d$ on the node
7: Compute the net force on node_i: $\boldsymbol{F}_i=\boldsymbol{F}_e+\boldsymbol{F}_o+\boldsymbol{F}_d$
8: Compute the acceleration $\boldsymbol{a}_i$ of node_i at time t
9: Compute the velocity $\boldsymbol{v}_i$ of node_i at time t
10: Determine the relationship between threshold value $\boldsymbol{a}_{th}$ &$\boldsymbol{v}_{th}$ and acceleration $\boldsymbol{a}_i$ and velocity $\boldsymbol{v}_i$ respectively, and then proceed accordingly.
11: Compute the position $\boldsymbol{P}_i(t)$ of node_i at time t, and move the node to this position
12: Determine if node_i reached equilibrium at time t. Terminate node's motion if yes; otherwise the loop executes again.
13: End For
14: All nodes have reached equilibrium, algorithm ends.

Figure 4. Adaptive Self-Spreading Algorithm

Through the analysis above, we optimized and simplified the algorithm proposed by previous researchers. We managed to reduce the complexity of similar algorithms but successfully maintained the most important feature: adaptivity.

V. SIMULATION AND EXPERIMENT RESULTS

Numerous simulations and experiments have been carried out in various environments to investigate the performance of our approach. Figure 5 shows the moving node used in our experiments. We use a MICA2 as the controller board, which requires a 3V power supply. We mounted a 6V battery holder as the power supply for the two servo motors under the chassis. The servo motors are mounted symmetrically on left and right side for the differential driving. Typically, servo motors have a feedback loop inside, and rotate through a limited angle proportional to the input signal. Our servo motors are home-modified in order to control the rotation speed of the rotor rather than the rotation angle. On top of the standouts there is a printed circuit board (PCB) designed to integrate the peripheral circuit of motor control and sensors, and a socket to connect the MICA2 board. The Parameter settings of our simulations are indicated in Table 1 with reference to the related figure number.

Figure 5. The nodes used in our system

Table 1 Simulation settings by figure number

Parameter	**Values**
Deployment Area (A)	10×10
Charges (Q)	1e-9
Quality(m)	9e-9
Coulomb constant (k)	9e9
Rs, Rc	2; 4
Damping factor	0.5
ath, vth	2; 1
Parameter	Values

a. Coverage and Uniformity

Generally, coverage can be considered as the measure of quality of service of a sensor network. Gage firstly proposed the concept of coverage degree in the study of multi-robot system [22]; according to Literature [21], it is defined as the ratio of the total covered area of all nodes to the whole target area. Among which, the total area covered by the nodes is taken from the union in the set; therefore, the coverage rate is usually less than or equal to 1. The covered uniformity is a good standard to measure the network lifetime. According to Literature [22], it is defined as the

standard deviation of the distance among nodes; the smaller value of the standard deviation is, the better uniformity of network coverage is.

This paper also studies some other metrics related to Coverage and Uniformity. Figure 6 shows the change in Coverage over the number of nodes. We also do the experiment with twenty nodes in our laboratory room. As shown in figure 6, the thick red line indicates the experiment result. We can find that the experiment performance is a little worse than the simulation ones. As is shown in the chart, for a deployment region of fixed size, Coverage grows as the number of nodes deployed. But there is an optimization required to maximize Coverage with minimum number of nodes.

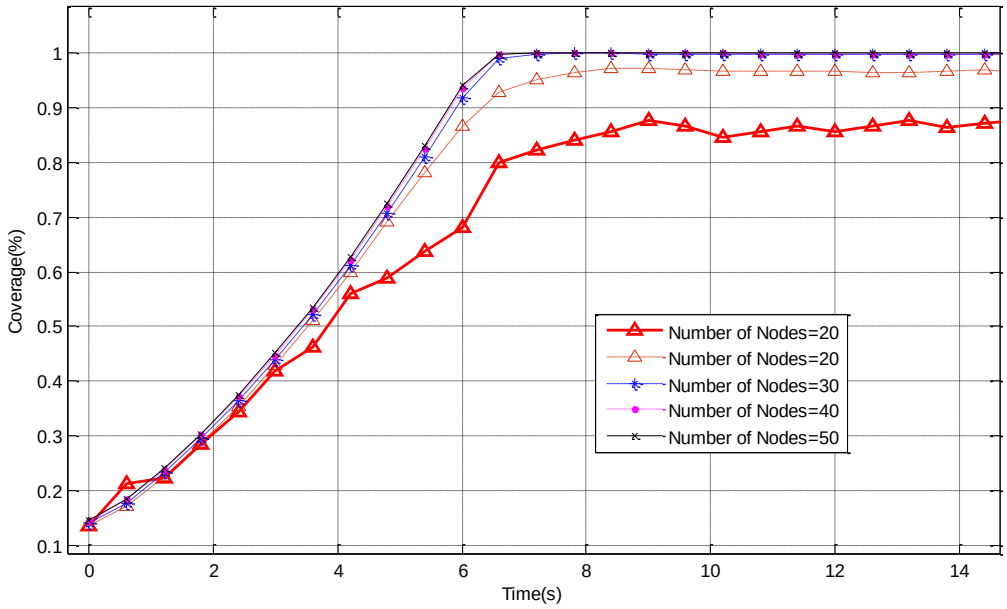

Figure 6. Coverage versus Time with different nodes number

Figure 7 is the curve of Coverage versus deployment region size. The x-coordinate represents the side length of the square region, while y-coordinate is the Coverage. Assume the network is connected, increasing the size of the deployment region will undoubtedly decrease the Coverage.

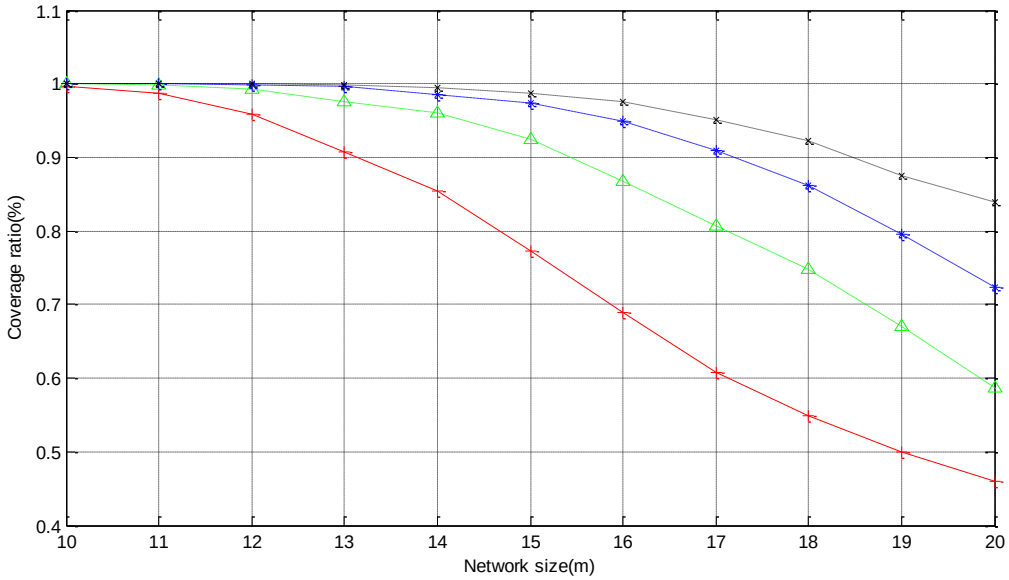

Figure 7. Coverage versus Network size

Figure 8 indicates the change in the correlation between uniformity and the number of nodes in the network. It can easily be seen that, uniformity is increasing when the number of nodes is increasing. But this doesn't necessarily mean that deploying 70 nodes in a region has worse uniformity than deploying 30 in the same one. In this paper, the computation of uniformity is based on the standard deviations of the distances and angles between adjacent nodes. It is meaningless to compare the value of uniformity among different number nodes. With the static equilibrium of nodes in the deployment region, its uniformity is becoming better gradually. In the meantime, the index of uniformity decreases and stabilizes around its minimum value.

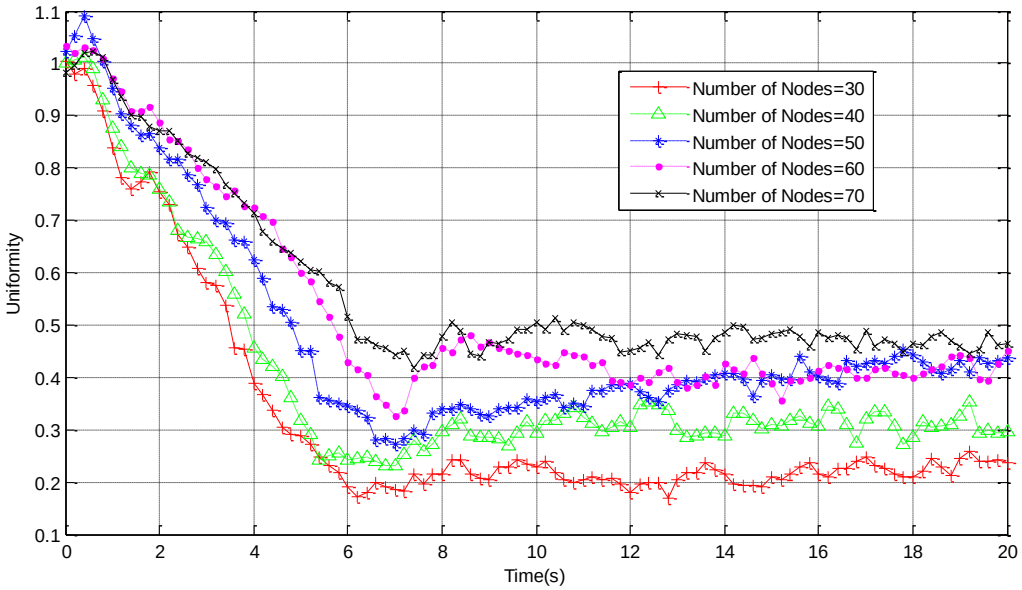

Figure 8.Uniformity versus Time with different nodes number

b. Time

In the case of emergency such as fire or rescue, the time needed for the network to deploy all the nodes at their final location is an utterly important factor to consider. Generally, the time of coverage is determined by algorithm complexity and the physical time needed for the nodes in motion. The total time elapsed is defined here as the time elapsed until all the nodes reach their final locations.

Figure 9 investigates the relation between Termination time and Damping factor. We can see from the figure that, Termination time tend to be quite large when damping factor is relatively small, decreases first and bounce back as the damping factor increases (1<factor<5), and reaches its minima at a certain damping factor. Damping factor determines the damping force that acts on a node when it is in motion. There is less resistance on the node when damping factor is relatively small, therefore its acceleration and velocity is relatively large, and when a node runs into an obstacle or its boundary, it will slow down due to a sudden repulsive forceso that it needs more time to reach equilibrium. As the damping factor increases, there is a certain value of it that will make the velocity of the node 0 when it hits the boundary. At this moment, the time needed for deployment is the shortest. As damping factor continues to increase, resistance will also increase when nodes are in motion. Therefore the velocity slows down and a longer time is

needed to reach equilibrium. In practical applications, damping factor represents the resistance on the moving node and it should be measured according to different environments.

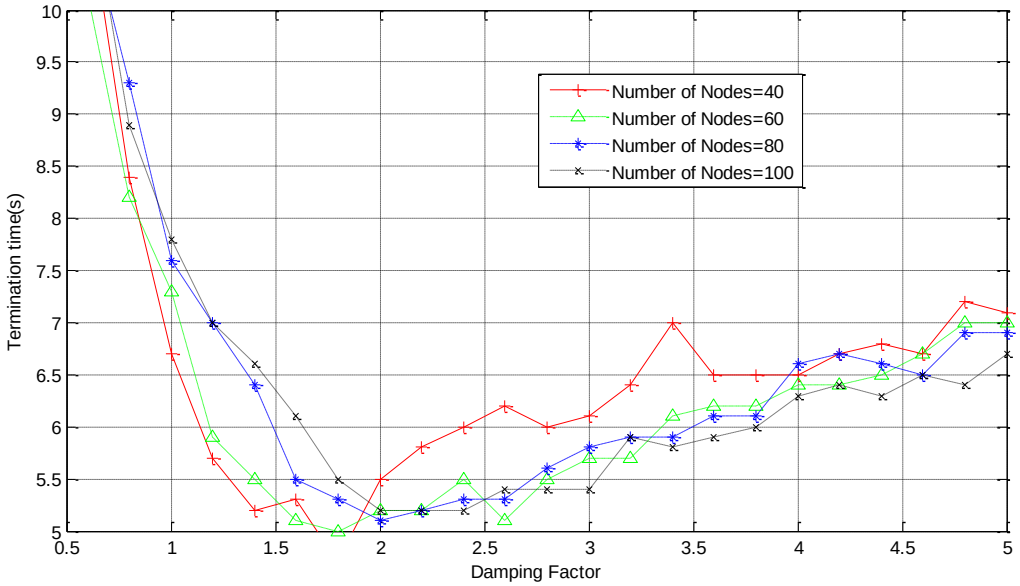

Figure 9. Termination time versus Damping factor

Figure 10 discusses the relationship between Termination time and Electric charge. The amount of charges a node carries directly determines the Coulomb's force acting on it in the electric field. The more charge a node carries, the larger Coulomb's force is acting on it. Under the same parameters, we use different numbers of nodes to simulate. As is shown in the chart, the deploy time of the nodes decreases first, then increases according to the electric charge on the nodes. It can be explained that, when the electric charge is less than 1, the Coulomb's force acting on it is relatively small, therefore its acceleration and velocity is small, and nodes need more time to reach equilibrium. With the increase of the electric charge, the Coulomb's force acting on the node is increasing. There will be an exact amount of charge, at which the velocity of the node becomes 0 when it hits the boundary. At this moment, the time needed for deployment is the shortest. As the electric charge further increases, the Coulomb's force acting on node becomes larger. A relatively larger velocity is achieved when the node arrives at the boundary, therefore the node needs more time to reach static equilibrium. In applications, the amount of charges a node carries can be corresponded to the power consumption of nodes in motion. The more energy

a node consumes in a time unit, the more power it runs on and the larger amount of electric charges it has.

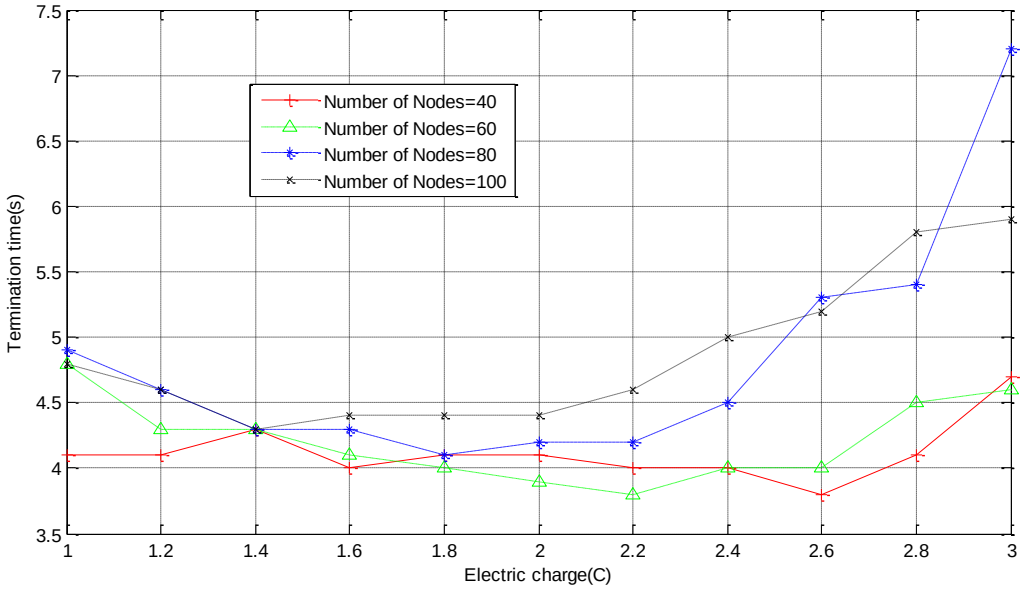

Figure 10. Termination times versus Electric charge

Figure 11 shows the relationship between deploy time and the area of deployment region, i.e. network size. As is shown in the chart, with the increase of the network size, it takes longer time to deploy all the sensor nodes, which is easy to understand. However, we have to notice one thing, under a certain condition, the increase of network size will not only take more deploy time but also decrease the coverage. It becomes important to choose the appropriate number of nodes according to different network size.

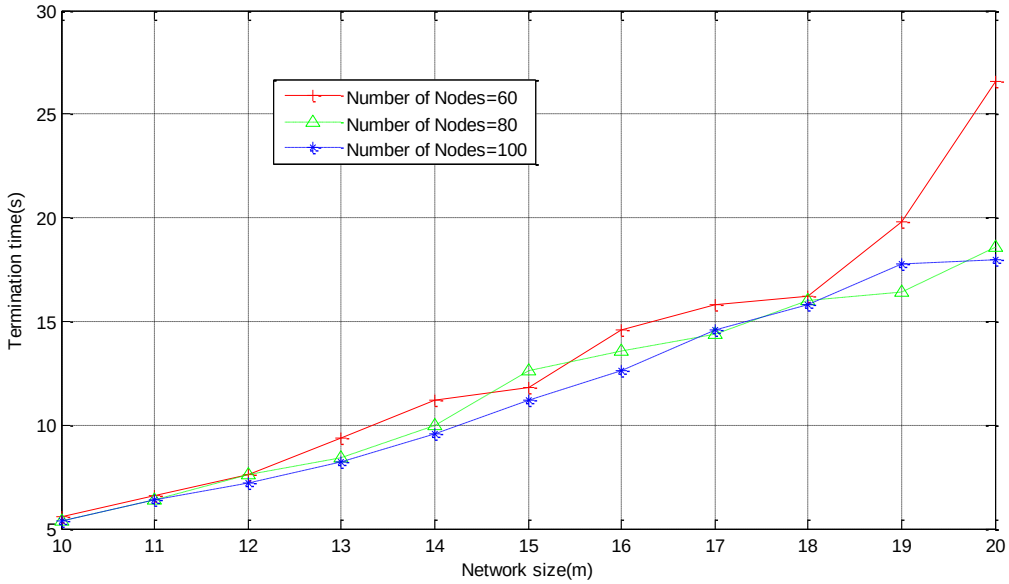

Figure 11. Termination time versus Network size

We may conclude from the above analyses that, in order to obtain a shorter deploy time when deploying sensor nodes, we need to consider a large number of factors together, such as network size, damping factor, and electric charge, etc. After the network size has been determined, we can select the appropriate number of nodes, and then determine appropriate amount of electric charges based on the damping factor in deploying environment in order to obtain the shortest deploy time.

c. Distance

The average distance traveled by each node is linked to the required energy for its movement. So the expected distance traveled is important for the estimation of required energy when each node has a limited energy supply. The variance of traveled distance is also important to determine the fairness of the deployment algorithm and for system energy utilization.

Figure 12 studies the distance the node moves under different damping factors. As is shown in the chart, the increase in damping factor will decrease the distance, which is easy to explain. What we have to pay attention to is that, the damping factor has something to do with the topography of the environment in actual deployment. It's not correct that a larger damping factor is better. The oversize of damping factor will lead to a longer deployment time.

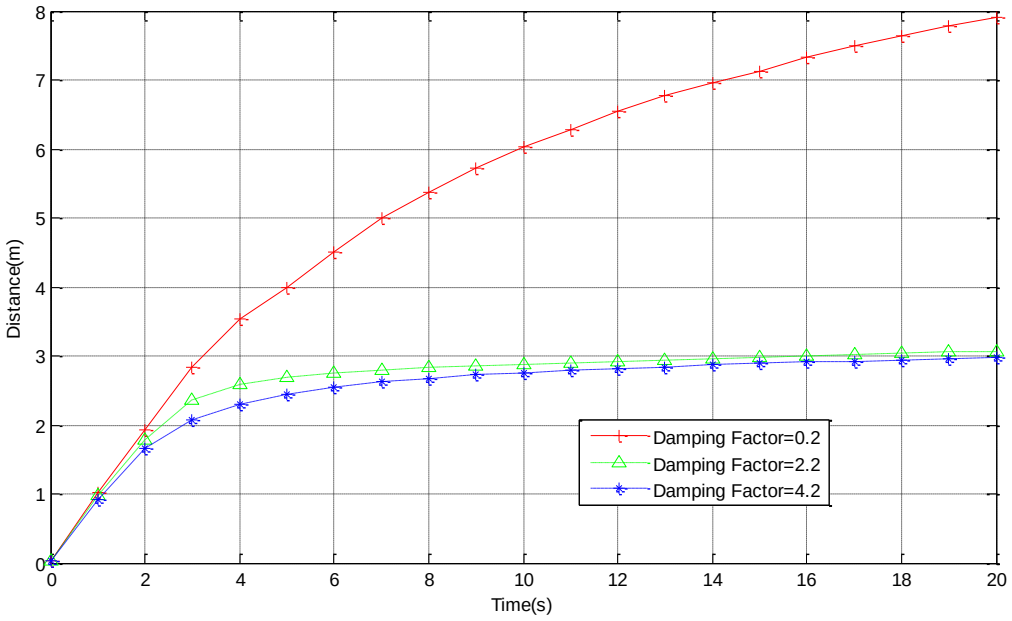

Figure 12. Distance versus Time with different damping factor

Figure 13 reveals the relationship between the average distance that each node moves and the number of nodes being deployed. From the chart, we can see the distance starts to grow dramatically in the first few moments (Time<6) of the deployment, and then tends to stabilize. At the beginning of the deployment, nodes have higher densities, as a result there is a relatively large Coulomb's force and acceleration between them. The velocity of the moving nodes is increasing quickly, making the nodes spread over the deployment region. When nodes hit the boundary, the repulsive force from the boundary will exert on them and slow down the velocity, as a result, they will reach equilibrium. That is why the curve changes in this way. We can also tell from the chart that the distance that each node moves increases as the number of nodes increases, but this change is not very obvious.

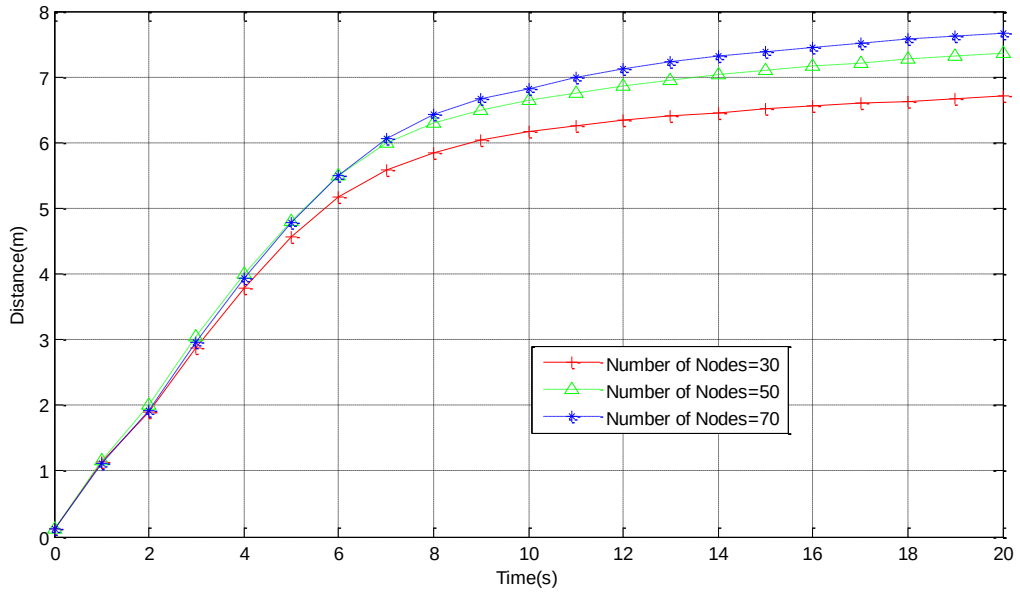

Figure 13. Distance versus Time with different Nodes number

Figure 14 discusses how the distance that each node moves changes with the network size. The initial positions are located in the center. The plot tells us that the distance that each node moves increases with the network size. At the first few moments of the deployment (Time<5), three curves coincide, which indicates that the average distances are the same. With the further movement of the nodes, the network of the smallest size ((Network size=10)) reaches equilibrium first, therefore the average distance that nodes move tends to remain steady. However, the network of larger size (Network size=15, 20) requires more distance in order to reach equilibrium.

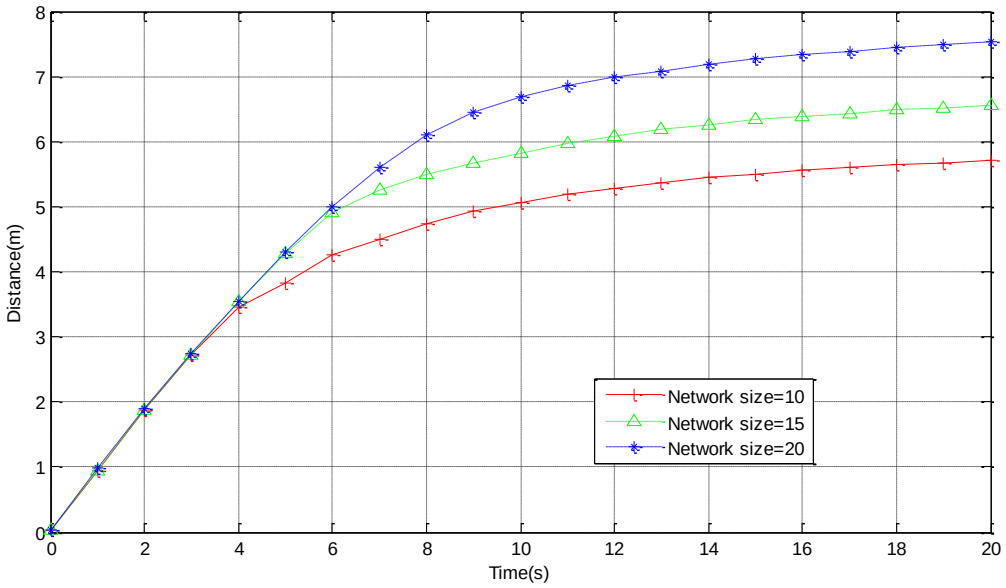

Figure 14. Distance versus Time with different Network size

We can conclude from the above analyses that in order to acquire a smaller average distance that every node moves in actual deployment, we should pay more attention to the initial positions of the sensor nodes. Network size and damping factor are largely objective and related to the environment, while selecting appropriate initial positions can be under control and much easier. An appropriate initial position can significantly reduce the average distance that each node moves, and as a result, we can save more energy.

VI. CONCLUSIONS

The deployment of mobile sensor network has always been the hot issue in this research field. How to save time, effort and be evenly in nodes deployment and maximize the time that the network exists is what we seek in every deployment algorithm. In practical applications, the problems and difficulties we meet are complex and various. Therefore, a kind of extensible, self-adaptive, robust and simple deployment algorithm is required. This paper proposes a distributed, self-adaptive and extensible mobile sensor network deployment algorithm based on the existing researches. According to this algorithm, the obstacles and nodes in the deployment area are taken as the charged particles and the particles will move due to the Coulomb's force from other

particles or obstacles. Finally, all the nodes automatically spread to the whole area by the force and complete the deployment.

Based on the algorithm in this paper, a lot of simulation tests are made to check the effect of this algorithm. We also simulate the normal deployment, the obstacle deployment, node failure redeployment and interest area deployment. Moreover, the four indicators, coverage, uniformity, deployment time and displacement distance, are used to evaluate the performance of the algorithm.

The simulation results demonstrate that the performance index of this algorithm works well in various scenes.

ACKNOWLEDGEMENTS

This work was supported by University Natural Science Research Project in Jiangsu Province(13KJB520001).

REFERENCES

[1] N. Bartolini, T. Calamoneri, T. La Porta and S. Silverstri, “Mobile sensor deployment in unknown fields”, IEEE International Conference on Computer Communications (INFOCOM 2010), 2010 Proceedings IEEE, pp: 1-5, San Diego, CA, 2010.

[2] C.H. Lin and C.T.King, “Sensor-deployment strategies for indoor robot navigation”, IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol.40, 2010, pp: 388-398.

[3] K. F. Ssu, W. T. Wang, F. K. Wu and T. T. Wu, “k-barrier coverage with a directional sensing model”, International Journal on Smart Sensing and Intelligent Systems, vol. 2, No.1, 2009, pp: 75–93.

[4] J. Chen, J. Li and T. H. Lai, “Trapping mobile targets in wireless sensor networks: An energy-efficient perspective”, IEEE Transactions on Vehicular Technology, vol. 62,No.7, 2013, pp:3287-3300.

[5] M. Erdelj, T. Razafindralambo and D. Simplot-Ryl, “Covering Points of Interest with Mobile Sensors”, IEEE Transactions on Parallel and Distributed Systems, vol. 24, No.1, 2013, pp: 32-43.

[6] T. W. Sung and C. S. Yang, “A cell-based sensor deployment strategy with improved coverage for mobility-assisted hybrid wireless sensor networks”, International Journal of Ad Hoc and Ubiquitous Computing, vol.5, 2010, pp: 189-198.

[7] T. M. Cheng and A. V. Savkin, "Decentralized Control of Mobile Sensor Networks for Asymptotically Optimal Blanket Coverage Between Two Boundaries", IEEE Transactions on Industrial Informatics, vol.9, No.1, 2013, pp: 365-376.

[8] H. Mahboubi, K. Moezzi, A. G. Aghdam, K. Sayrafian-Pour and V. Marbukh, "Distributed deployment algorithms for improved coverage in a network of wireless mobile sensors", IEEE Transactions on Industrial Informatics, vol.10, No.1, 2014, pp: 163-174.

[9] G. Tan, S.A. Jarvis and A. M. Kermarrec, "Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks", IEEE Transactions on Mobile Computing, vol.8, 2009, pp: 836-848.

[10] N. Bartolini, G. Bongiovanni, T. F. L. Porta, and S. Silvestri, "On the vulnerabilities of the virtual force approach to mobile sensor deployment", IEEE Transactions on Mobile Computing, vol.13, No.11, 2014, pp: 2592-2605.

[11] J. Z. Du, K. Wang, H. Liu and D.K. Guo, "Maximizing the Lifetime of k-Discrete Barrier Coverage Using Mobile Sensors", IEEE Sensors Journal, Vol.13, No. 12, 2013, pp: 4690-4701.

[12] S.K. Udgata, S.L. Sabat and S. Mini, "Sensor deployment in irregular terrain using artificial bee colony algorithm", iNature & Biologically Inspired Computing(NaBIC), World Congress on, pp: 1309-1314, Coimbatore, Dec. 2009.

[13] N.K.Suryadevara, A. Gaddam, R.K.Rayudu and S.C. Mukhopadhyay, "Wireless Sensors Network based safe Home to care Elderly People: Behaviour Detection", Sens. Actuators A: Phys. (2012), doi:10.1016/j.sna.2012.03.020, Volume 186, 2012, pp. 277 – 283.

[14] C. Ozturk, D. Karaboga and B. Gorkemli, "Probabilistic Dynamic Deployment of Wireless Sensor Networks by Artificial Bee Colony Algorithm", Sensors, vol.11, 2011, pp: 6056-6065.

[15] A. Howard, M. J. Matarić and G.S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem", Distributed Autonomous Robotic Systems 5, 2002, pp: 299-308.

[16] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces", Twenty-Second Annual Joint Conference of the IEEE Computer and Communications (INFOCOM 2003), vol.2, pp: 1293-1303, 2003.

[17] N. K. Suryadevara, S. C. Mukhopadhyay, R.Wang, R.K. Rayudu and Y. M. Huang, Reliable Measurement of Wireless Sensor Network Data for Forecasting Wellness of Elderly at Smart Home, Proceedings of IEEE I2MTC 2013 conference, IEEE Catalog number CFP13IMT-CDR, ISBN 978-1-4673-4622-1, May 6-9, 2013, Minneapolis, USA, pp. 16-21.

[18] B. Shucker and J. K. Bennett,"Scalable control of distributed robotic macrosensors", Distributed Autonomous Robotic Systems 6, 2007, pp: 379-388.

[19] T. Stavros and L. Tassiulas, "Packetostatics: Deployment of massively dense sensor networks as an electrostatics problem", 24th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2005), vol. 4, pp: 2290 – 2301, 2005.

[20] S. Bhardwaj, D.S. Lee, S.C. Mukhopadhyay and W.Y. Chung, "Ubiquitous Healthcare Data Analysis And Monitoring Using Multiple Wireless Sensors for Elderly Person", Special issue on Modern Sensing Technologies, Sensors and Transducers Journal, ISSN 1726-5479, Vol . 90, pp. 87-99, April 2008.

[21] X.Y. Yu, W. P. Huang, J. J. Lan and X. Qian, "A novel virtual force approach for node deployment in wireless sensor network", Distributed Computing in Sensor Systems (DCOSS), 2012 IEEE 8th International Conference on., pp:359-363, Hangzhou, May 2012.

[22] S. C. Mukhopadhyay, Anuroop Gaddam and Gourab S. Gupta, Wireless Sensors for Home Monitoring - A Review, Recent Patents on Electrical Engineering 1, 32-39, 2008.

[23] J. Li, B. H. Zhang, L. G. Cui and S. C. Chai, "An extended virtual force-based approach to distributed self-deployment in mobile sensor networks", International Journal of Distributed Sensor Networks, vol.2012, Article ID 417307, 14 pages, 2012.

[24] I. Larrabide, M. Kim and L. Augsburger, "Fast virtual deployment of self-expandable stents: Method andin vitro evaluation for intracranial aneurysmal stenting", Medical image analysis, vol.16, 2012, pp: 721-730.

[25] N. Heo and P. K. Varshney, "A distributed self spreading algorithm for mobile wireless sensor networks", Wireless Communications and Networking (WCNC 2003), vol. 3, pp: 1597-1602, New Orleans, LA, 2003.

[26] D.W. Gage, "Sensor Abstractions to Support Many-Robot Systems", SPIE Mobile Robots VII, vol. 1831, pp: 235-246, Boston, 1993.