Dynamic Deployment of Sensors using Virtual Forces-directed Co-Evolutionary Particle Swarm Optimization (VFCPSO) Method

Abha Jain, Prof. A. P. Laturkar

E&Tc Department, Modern College of Engineering, Pune 411005.

Abstract—

Wireless sensor networks (WSN) present numerous research possibilities due to the bright and promising future of information technology. WSNs are a group of inexpensive, less power consuming, having multiple functions and tiny wireless nodes that work in unison. They sense the environment; do simple tasks like processing of data and are able to communicate wirelessly over a short distance. How effective a WSN will depend on the probability of coverage and on detecting target. Many coverage strategies like force based (Virtual Forces Algorithm), Grid-Based (triangular lattice, hexagon and square grid) and computational geometry based (Voronoi and Delaunay triangulation) are suggested. Another method is the Particle Swarm Optimization method (PSO). This is a type of Evolutionary Algorithm and shows prospects in being able to solve complex optimization problems. The demand to achieve optimum coverage inspires us to move towards hybrid algorithms. The hybrid algorithms combine more than one of the above mentioned approaches. We aspire to achieve optimum Coverage by implementing an algorithm called Virtual Force-directed Co-evolutionary Particle Swarm Optimization (VFCPSO). This algorithm is a hybrid of Virtual Forces Algorithm and Co-evolutionary PSO. It is flexible enough for a network formed from permutations of homogeneous, heterogeneous, stationary and mobile sensors. The VFCPSO predicts deploying actively with better abilities to search overall and to converge reginally. We aim to implement and compare the results of VFA, PSO, VFPSO and VFCPSO. The expectation here is to get a noticeable increase in effective coverage area and a noticeable decrease in average computation time.

Index Terms *—VFA, PSO, VFCSPO, deployment optimization strategies*

How effective a WSN is will largely depend upon the coverage obtained by deploying sensors. The positioning of sensors can impact coverage, expense of communication and management of resources. Since sensors can be spread randomly, one basic consideration in a WSN is the problem of coverage. In general, this reflects how well an area can be under surveillance from sensors as shown in [1]. This paper focuses on studying various strategies for placing sensors and gets a comparative view of some of them.

Initial step in deploying the sensors is to place them randomly; as sometimes *beforehand* information of the terrain is not known. In defence usage, randomly deploying is at times the only way, where the sensors are air-dropped or thrown into the target field. However, random placement is not an ideal arrangement as it may lead to ineffective coverage because of over-clustering or low concentration of sensors in some parts of the area.

This brings us to deploying sensors using intelligent algorithms, after random placement. Such types of deployment pose a requirement on the sensors to be mobile. To make it cost effective, in real circumstances WSNs are a mix of mobile and stationary sensors. This type of a network is called heterogeneous network. The key point of this paper is to apply algorithms based on virtual-forces, particle swarm optimization and a combination of their hybrids. Finally a comparison of performance of all these algorithms can be made and a conclusion can be drawn as to the best algorithm for effective coverage of WSNs.

One such approach to deploy mobile sensors is the Virtual Forces algorithms. Many usages have proven that VFA performs outstandingly when the network consists of only mobile sensors. However, the performance falls when VFA is implemented to a network having mobile and stationary type of nodes. VFA models the sensors as particles that exert a force (repulsive or attractive) between themselves. Based on this model, the sensors can be moved away from or toward each other to achieve efficient coverage for the desired area [2]. This puts a limitation on the sensors to be mobile only. If any stationary sensors are present then they present a hindrance on movement of mobile sensors.

Classical techniques of optimised coverage need lots of computation, which grows exponentially as the problem size increases [3]. It is hence advisable to have such optimization processes that require moderate memory and computation and still give expected performance. This leads us to look at another algorithm based on Particle Swarm Optimisation. Particle Swarm Optimization (PSO) is a popular bio-inspired, multi-dimensional technique [4]. The power of PSO is that it is easy to implement, gives good quality solutions, is very efficient to compute and converges swiftly. Each node is modelled as a particle which in turn can be viewed as a potential solution to the optimisation. As it can be imagined that a flock of birds are flying in a particular area, looking for food, the particles can be imagined to be moving through area to be searched for finding an optimum result.

However, with increase in magnitude, the complexity of computation for PSO grows exponentially. As a result the computation time acts as a hold-up of PSO. So, another co-evolutionary method is proposed which is an improved PSO but on the lines of co-evolution of populations or co-existence of several species. This achieves a parallel processing in a multi-dimensional search space, thereby improving computational time and efficiency [5].

However, an algorithm known as VFCPSO (Virtual Force-directed Co-evolutionary Particle Swarm Optimisation) can further make the quality of solution better and also, make it sturdier. This happens by taking a hybrid the VFA and CPSO [6].

The paper here is divided as follows. Section II consists of the prelude about the detection model and assumptions made for evaluation purposes for the WSN. Section III explains the VFA, the concept behind it and equations that influence movements of sensors. Section IV explains the approach of PSO and provides an explanation of how an optimal position of sensors is computed. Section V explains the CPSO and its approach in solving multi-dimensional search in a multi-dimensional space. Section VI explains the hybrid VFCPSO algorithm which achieves the most optimum deployment for a network of mixed nodes, i.e. mobile and stationary. Section VII explains the expected results as an outcome of implementing the algorithms and comparing their performances.

- For any WSN two types of detection models are available: binary and probabilistic. Without adding any new limitations we assume a binary detection model. This makes evaluation simpler.
- We adopt the discrete coordinate system, where the monitoring area is assumed to be a grid of two dimensions.
- For evaluation purposes it is assumed to have a WSN with a powerful super node. This node can be viewed as the sink node. It can be assumed to be a centralized processing center for all the algorithms that will be implemented.
- Mobile sensor nodes are able to move to the exact location that has been calculated and scheduled.
- Each sensor is aware of its positioning beforehand from the starting itself with help of some global positioning system.
- After initially deploying all sensors randomly all the nodes can communicate with the sink node.

The virtual force algorithm is an algorithm to organize itself and is modeled on disk packing theory [7] and virtual field concept that has been inspired from robotics [8].

Each sensor can be seen as a source which exerts a force on other sensors. This force can be positive (attractive) or negative (repulsive). Since these forces are of imaginary kind, the algorithm takes the name virtual forces algorithm. If two sensors are closer than desired, they are repelled away by the negative force and if the sensors are farther than desired, then they are attracted towards each other with a positive force.

Let us suppose there is some number of sensors in a network. Then total force acting on a particular sensor will be the sum of forces acing on that sensor from other sensors.

In case of a pre-defined threshold, the distance between a particular sensor and other sensors is calculated sequentially. If the actual distance is greater than threshold then an attractive force is applied between the two sensors. If the actual distance is lesser than threshold then a repulsive force is applied between the two sensors. In this manner the force on each sensor from all other sensors is calculated and a vector summation of these forces is taken. The resultant forces are not applied immediately, but in the end, when resultant forces for all sensors have been calculated, the actual movements on sensors are performed.

Enough literature is available to show that VF algorithm gives outstanding performance when all nodes are only mobile nodes. However its performance goes down when the network contains both stationary and mobile nodes. The movement of sensors is resisted by forces that the stationary sensors exert on mobile sensors and the VFA cannot tackle this problem. Also the success of VFA depends on threshold distance and finding proper value of the threshold distance may become difficult in various situations and requirements.

The drawback of VFA can be eliminated by applying PSO method which searches for the optimal results globally and it has no effect even if stationary nodes are present in space. Eventually it gives better performance as compared to VFA in terms of self-adaptation and global search. It is a swarm-intelligence based evolutionary algorithm. Each sensor node is modeled as a particle. Each particle can be seen as a likely result of this process of optimization.

Supposing, we consider a swarm of a particular size. Then this swarm is considered to be made up of some particles which are actually sensors. Then for each particle it is assumed that it knows its local best position and its current position. Assuming that the global best position can be found in a certain number of iterations, then, in every iteration, the new global best position is compared with its previous one. The particle's current velocity while accelerating towards another particle with bounded random acceleration, are then updated after comparing it to the global best position.

For a global search, there is a gradual reduction in granularity and a trade-off is achieved between speed and precision. For the solution to remain valid velocity of particles must be renewed randomly and fitness associated with new granularity must be re-analyzed.

Even if PSO seems for solving multi-dimensional function optimization in continuous space and parallel computing mechanism is adopted, still the execution time is a big hurdle of PSO. This is especially true of large scale WSNs which consist of a large number of mobile sensor nodes.

Co-evolutionary PSO algorithms model the co-existence of several species. We can improve the searching ability of PSO in a high-dimension search space by dividing this search space into lower-dimensional subspaces. This is achieved by splitting the solution vectors into smaller vectors. This is the main concept behind CPSO.

For finding the optimality of a *n*-dimensional search space vector, the vector is divided into different elements which are called swarms. Each of this swarm tries to optimize every element of the solution vector. So ultimately what is achieved is that the *n*-dimensional optimization problem is divided into many 1-dimensional optimization problems, which can be solved in parallel. An easy method to do this is to add in succession the global best particle from each swarm and make an *n*-dimensional vector. Although, the CPSO algorithm converges faster, it may become trapped in sub-optimal locations in search space.

Due to the limitations of CPSO as mentioned in section V, we consider a hybrid of CPSO with VFA for dynamic deployment in WSNs, and it is called as the Virtual Forces-directed

Co-evolutionary Particle Swarm Optimization (VFCPSO). In VFCPSO, in order to optimally deploy, the global search of can be got from the hybrid CPSO algorithm. This is performed in a co-evolutionary manner.

Unlike, the PSO algorithm, the velocity of each particle is also updated according to virtual forces of sensor nodes. The steps followed are as follows:

1. Initially define the *n*-dimensional search space
2. Create and initialize *n* such one-dimensional PSOs. This means that the *n*-dimensional search space is divided into n spaces of one-dimension and each space is a PSO.
3. For each PSO compute the effective coverage of that area which is constituted by stationary nodes.
4. Within each PSO, find the area that has effective coverage for each swarm.
5. In each iteration keep comparing with the local best position and global best position of each particle of each swarm. Update these parameters only if they are better than previous values. Else retain the previous values.
6. Compute the virtual forces amongst sensor nodes using equations from VFA
7. Update PSOs on with the new found values.
8. Again compute the effective coverage of area
9. Keep updating PSOs until the stopping condition is satisfied.

For comparing the performances of VFA, PSO and VFCPSO algorithms in the optimization of dynamic deployment in WSNs, we will be simulating different WSN scenarios using NS2. We plan to observe the variations in performances of these algorithms for varying number of stationary and mobile nodes. We hope to see a drastic improvement in performance of VFCPSO as compared to VFA and PSO. We will also be observing the performances of these algorithms for varying number of iterations and hope to see that the effective coverage area of VFCPSO is still highest compared to VFA and PSO.

1. ang and Yu-Chee Tseng; "The Coverage Problem in a Wireless Sensor Network," *in Mobile Networks and Applications 10, 519–528, 2005*
2. Qin Xu and Qianping Wang; "Coverage Optimization Deployment Based on Virtual Force-Directed in Wireless Sensor Networks" presented at *2012 International Conference on Computer Technology and Science* (ICCTS 2012)
3. Raghavendra V. Kulkarni and Ganesh Kumar Venayagamoorthy, "Particle Swarm Optimization in Wireless Sensor Networks: A Brief Survey" *in IEEE Transactions on Systems, Man and Cybernetics-Part C: Applications and Reviews.*
4. J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks,* vol. 4, 27 Nov.–1 Dec. 1995, pp. 1942–1948.
5. F. Van den Bergh and A.P. Engelbrecht, "A cooperative approach to particle swarm optimization" in *IEEE Trans. on Evolut. Comput. 2004, 8, 225-239.*
6. Xue Wang, Sheng Wang and Jun-Jie Ma, "An Improved Co-evolutionary Particle Swarm Optimization for Wireless Sensor Networks with Dynamic Deployment" in *Sensors 2007, ISSN 1424-8220*.
7. Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces", *in IEEE INFOCOM, 2003, 1293-1303*.
8. Marco Locateli and Ulrich Raber, "Packing equal circles in a square: a deterministic global optimization" approach *in Discr. Appl. Math. 2002, 122, 139-166.*
9. A. Howard, M. J. Matari´c, and G. S. Sukhatme, "Mobile sensor network deployment using potential field: a distributed scalable solution to the area coverage problem" in *Proc. of Int. Symp. Distrib. Auton. Robot. Syst. 2002, 299-308.*
10. Shijian Li, Congfu Xu, Weike Pan and Yunhe Pan, "Sensor Deployment Optimization for Detecting Maneuvering Targets", in *7th Int. Conf. Inform. Fusion IEEE 2005, 1629-1635.*