ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

**IJARCCE**

**International Journal of Advanced Research in Computer and Communication Engineering**

**ICITCSA 2017**

**Pioneer College of Arts and Science, Coimbatore**

**Vol. 6, Special Issue 1, January 2017**

# Effective Global Sensor Deployment for Coverage Problem in WSN using Ant Colony Optimization

**T.B. Saranya Preetha[1], V. Sowmiya[2], K. Yamuna devi[3]**

Assistant Professor, Department of Computer Science, PKR Arts College for Women, Gobi, Erode, Tamilnadu, India[1]

Research Scholar, Department of Computer Science, PKR Arts College for Women, Gobi, Erode, Tamilnadu, India[2,3]

**Abstract—** This paper aims to identify optimal deployment locations of the given sensor nodes with a pre-specified sensing range, and to schedule them such that the network lifetime is maximum with the required coverage level. Since the upper bound of the network lifetime for a given network can be computed mathematically, this knowledge is used to compute locations of deployment such that the network lifetime is maximum. In this thesis ultimate goal is to realize an automated monitoring network so that detection applications of various emergency events can be practically implemented. Further, the nodes are scheduled to achieve this upper bound. This project uses artificial bee colony algorithm and particle swarm optimization for sensor deployment problem followed by a heuristic for scheduling. In addition, ANT colony optimization technique is used to provide maximum network lifetime utilization. The comparative study shows that artificial ACO performs better than bee colony algorithm for sensor deployment problem. The proposed heuristic was able to achieve the theoretical upper bound in all the experimented cases.

## I. INTRODUCTION

Wireless sensor networks have recently come into prominence because they hold the potential to revolutionize many segments of our economy and life, from environmental monitoring and conservation, to manufacturing and business asset management, to automation in the transportation and health care industries. The design, implementation, and operation of a sensor network requires the confluence of many disciplines, including signal processing, networking and protocols, embedded systems, information management and distributed algorithms. Such networks are often deployed in resource-constrained environments, for instance with battery operated nodes running un-tethered.

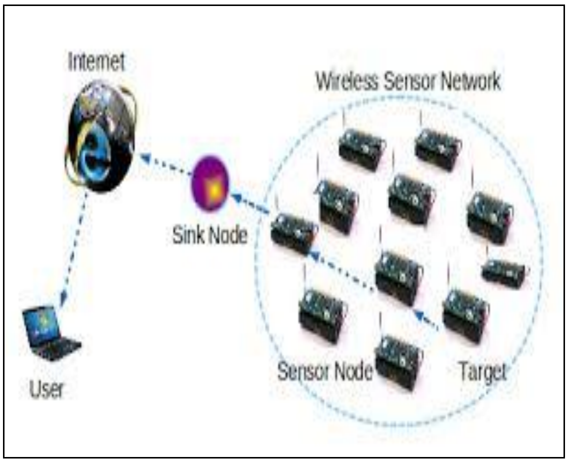

Fig.1 WSN Overview

These constraints dictate that sensor network problems are best approached in a hostile manner, by jointly considering the physical, networking, and application layers and making major design tradeoffs across the layers. Advances in wireless networking, micro-fabrication and integration (for examples, sensors and actuators manufactured using micro-electromechanical system technology, or MEMS), and embedded microprocessors have enabled a new generation of massive-scale sensor networks suitable for a range of commercial and military applications.

The technology promises to revolutionize the way we live, work, and interact with the physical environment. In a typical sensor network, each sensor node operates.

Sensor networks extend the existing Internet deep into the physical environment. The resulting new network is orders of magnitude more expansive and dynamic than the current TCP/IP networks and is creating entirely new types of traffic that are quite different from what one finds on the Internet now. Information collected by and transmitted on a sensor network describes conditions of physical environments for example, temperature, humidity, or vibration and requires advanced query interfaces and search engines to effectively support user-level functions.

- The proposed system is capable to optimize the solution even if the criteria are dynamically added.
- Works even in heterogeneous nodes environment.
- A high, low and attribute base class coverage level (sensing redundancy) means more sensors should be deployed with high overlapping degree, which directly affects the overall sensing coverage ratio under limited number of deployable sensors.
- Improve CASA protocol suite is implemented as a proof-of-concept prototype to corroborate the

protocol feasibility and demonstrate the emergency detection capability of MoNet.

## II. RELATED WORK

**Yi Zou and Krishnendu Chakrabarty et al [1] describe** The effectiveness of cluster-based distributed sensor networks depends to a large extent on the coverage provided by the sensor deployment. They propose a virtual force algorithm (VFA) as a sensor deployment strategy to enhance the coverage after an initial random placement of sensors. For a given number of sensors, the VFA algorithm attempts to maximize the sensor field coverage. A judicious combination of attractive and repulsive forces is used to determine virtual motion paths and the rate of movement for the randomly-placed sensors. Once the effective sensor positions are identified, a one-time movement with energy consideration incorporated is carried out, i.e., the sensors are redeployed to these positions. They also propose a novel probabilistic target localization algorithm that is executed by the cluster head. The localization results are used by the cluster head to query only a few sensors (out of those that report the presence of a target) for more detailed information. Simulation results are presented to demonstrate the effectiveness of the proposed approach.

**Yunxia Chen et al [2]** describe a **l**ifetime per unit cost, defined as the network lifetime divided by the number of sensors deployed in the network, can be used to measure the utilization efficiency of sensors in a wireless sensor network (WSN). Analyzing the lifetime per unit cost of a linear WSN, they find that deploying either an extremely large or an extremely small number of sensors is inefficient in terms of lifetime per unit cost. They thus seek answers to the following questions: how many sensors should be deployed and how to deploy them to maximize the lifetime per unit cost. Numerical and simulation results are provided to study the optimal sensor placement and the optimal number of deployed sensors.

**P. Corke et al [3]** describe a sensor network deployment method using autonomous flying robots. Such networks are suitable for tasks such as large-scale environmental monitoring or for command and control in emergency situations. They describe in detail the algorithms used for deployment and for measuring network connectivity and provide experimental data they collected from field trials. A particular focus is on determining gaps in connectivity of the deployed network and generating a plan for a second, repair, pass to complete the connectivity.

**Krishnendu Chakrabarty et al [4]** describe a novel grid coverage strategies for effective surveillance and target location in distributed sensor networks. They represent the sensor field as a grid (two or three-dimensional) of points (coordinates) and use the term target location to refer to the problem of locating a target at a grid point at any instant in time. They first present an integer linear programming (ILP) solution for minimizing the cost of sensors for complete coverage of the sensor field. They solve the ILP model using a representative public-domain solver and present a divide-andconquer approach for solving large problem instances. They then use the framework of identifying codes to determine sensor placement for unique target location. They provide coding-theoretic bounds on the number of sensors and present methods for determining their placement in the sensor field. They also show that grid-based sensor placement for single targets provide asymptotically complete (unambiguous) location of multiple targets in the grid.

**Pankaj K. Agarwa et al [5]** describe the problem of covering a two-dimensional spatial region P, cluttered with occluders, by sensors. A sensor placed at a location p covers a point x in P if x lies within sensing radius r from p and x is visible from p, i.e., the segment px does not intersect any occluder. The goal is to compute a placement of the minimum number of sensors that cover P. They propose a landmark-based approach for covering P. Suppose P has $\varsigma$ holes, and it can be covered by h sensors. Given a small parameter $\varepsilon > 0$, let $\lambda := \lambda(h, \varepsilon) = (h/\varepsilon) \log \varsigma$. They prove that one can compute a set L of $O(\lambda \log \lambda \log (1/\varepsilon))$ landmarks so that if a set S of sensors covers L, then S covers at least $(1 - \varepsilon)$-fraction of P. It is surprising that so few landmarks are needed, and that the number does not depend on the number of vertices in P. They then present efficient randomized algorithms, based on the greedy approach, that, with high probability, compute O(h˜ log λ) sensor locations to cover L; here ˜h ≤ h is the number sensors needed to cover L. They propose various extensions of their approach, including: (i) a weight function over P is given and S should cover at least $(1 - \varepsilon)$ of the weighted area ofP, and (ii) each point of P is covered by at least t sensors, for a given parameter $t \geq 1$.

**Jing LI et al [6]** describe a sensing coverage is a fundamental problem in sensors networks. Different from traditional isotropic sensors with sensing disk, directional sensors may have a limited angle of sensing range due to special applications. In this paper, they study the coverage problem in directional sensor networks (DSNs) with the rotatable orientation for each sensor. They propose the optimal coverage in directional sensor networks (OCDSN) problem to cover maximal area while activating as few sensors as possible. Then they prove the OCDSN to be NP-complete and propose the Voronoi-based centralized approximation (VCA) algorithm and the Voronoi-based distributed approximation (VDA) algorithm of the solution to the OCDSN problem. Finally, extensive simulation is executed to demonstrate the performance of the proposed algorithms.

## III.SENSOR DEPLOYMENT

Since the sensor nodes can be deterministically deployed, the optimal deployment locations and the schedule are decided at the base station, prior to actual deployment. The existing method has two phases: **sensor deployment** and **sensor scheduling**. The nodes are initially deployed randomly. Based on the theoretical upper bound of network lifetime, the optimal deployment locations are computed using ABC algorithm. A heuristic is then used to schedule the sensor nodes such that the network lifetime is maximum. In addition, to schedule the sensor nodes such that the theoretical upper bound of network lifetime can be achieved, the existing system proposes a weight-based method for determining the cover sets. It includes the following main steps:

- Weight assignment
- Cover formation
- Cover optimization
- Cover activation and Energy reduction.

Sensor coverage is important while evaluating the effectiveness of a wireless sensor network. A lower coverage level (simple coverage) is enough for environmental or habitat monitoring or applications like home security. Higher degree of coverage (k-coverage) will be required for some applications like target tracking to track the targets accurately .or if sensors work in a hostile environment such as battle fields or chemically polluted areas. More reliable results are produced for higher degree of coverage which requires multiple sensor nodes to monitor the region/targets.

In some cases, for the same application, the coverage requirement may vary. For example, for forest fire detections, the coverage level may be low in rainy seasons, but high in dry seasons. An example of Q-coverage is video surveillance systems deployed for monitoring hostile territorial area where some sensitive targets like a nuclear plant may need more sensors cooperate to ensure source redundancy for precise data. Both sensor deployment and scheduling are important to ensure prolonged network lifetime. Traditionally, the problems of sensor placement and scheduling have been considered separately from each other. A balanced performance is crucial for most applications.

Different sensor deployment strategies can cause very different network topology, and thus different degrees of sensor redundancy. A good sensor deployment with sufficient number of sensors which ensures a certain degree of redundancy in coverage so that sensors can rotate between active and sleep modes is required to balance the workload of sensors.

For performing this process the sensor battery plays vital roll in searching the sensors finding the location sending the data and receiving the data from different users. So that the device wants to be active for long time for that the saving the battery power is done on the system.

Since the sensor nodes can be deterministically deployed, the optimal deployment locations and the schedule are decided at the base station, prior to actual deployment. The proposed method has two phases: sensor deployment and sensor scheduling. The nodes are initially deployed randomly. Based on the theoretical upper bound of network lifetime, we compute the optimal deployment locations using ABC algorithm. A heuristic is then used to schedule the sensor nodes such that the network lifetime is maximum. Algorithm 1 describes the proposed method.

**Algorithm 1 Proposed Approach**

```
Input: S, T
Output: Optimal location of S and sensor schedule
Deploy S randomly
Compute upper bound of network lifetime using (2)
Recompute sensor node positions using ABC algorithm such that the upper bound of network lifetime is maximum
Design sensor schedule using the proposed heuristic for sensor scheduling such that the network lifetime upper bound is achieved.
```

## IV. ALGORITHMS

### A. Heuristic for sensor deployment

If any sensor node is idle (without monitoring any target), the node is moved to the least monitored targets' location. This is to ensure that all sensor nodes play their part in monitoring the targets. The sensor nodes are then sorted based on the number of targets it cover. The sensor node is placed at the middle of all the targets it covers. The next nearest target is identified and the sensor node is placed at the middle of all these targets. If it can cover this new target along with targets it was already monitoring, allow this move, and else discard the move. This is done till the sensor node cannot cover any new target. At the end, upper bound is computed. The drawback of this approach is that it depends on the initial position of the sensor nodes. Though it may perform well for dense deployments, consistency cannot always be guaranteed.

**Algorithm 2 A heuristic for Sensor Deployment**

```
Place sensor nodes randomly
for i = 1 to m do
    if Si does not monitor any target then
        Move Si to the least monitored target
        Recompute sensor-target coverage matrix
    end if
end for
S = Sensor nodes sorted in ascending order of number of targets it covers
for i = 1 to m do
    repeat
        Place Si at the center of all targets it covers
        Move Si to the center of all targets it covers and its next nearest target
        if Si can cover a new target then
            Recompute sensor-target matrix
        else
            Discard move
        end if
    until Si can cover another target
end for
Compute upper bound of network lifetime using
```

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering

ICITCSA 2017

Pioneer College of Arts and Science, Coimbatore

Vol. 6, Special Issue 1, January 2017

### B. ABC Based Sensor Deployment

Artificial Bee Colony (ABC) Algorithm is an optimization algorithm based on the intelligent behavior of honey bee swarm. The colony of bees contains three groups: employed bees, onlookers and scouts.The employed bee takes a load of nectar from the source and returns to the hive and unloads the nectar to a food store.

After unloading the food, the bee performs a special form of dance called waggle dance which contains information about the direction in which the food will be found, its distance from the hive and its quality rating.

**Algorithm 3 ABC Algorithm**

```
Initialize the solution population B
Evaluate fitness
cycle = 1
repeat
    Search for new solutions in the neighborhood
    if new solution is better than old solution then
        Memorize new solution and discard old solution
    end if
    Replace the discarded solution with a new randomly generated solution
    Memorize the best solution
    cycle = cycle + 1
until cycle = maximumcycles
```

### C. PSO Based Sensor Deployment

Particle Swarm Optimization (PSO) consists of a swarm of particles moving in a search space of possible solutions for a problem. Every particle has a position vector representing a candidate solution to the problem and a velocity vector. Moreover, each particle contains a small memory that stores its own best position seen so far and a global best position obtained through communication with its neighbor particles.

**Algorithm 4 PSO Algorithm**

```
Initialize particles
repeat
    for each particle do
        Calculate the fitness value
        if fitness value is better than the best fitness value ( pbest) in history then
            Set current value as the new pbest
        end if
    end for
    Choose the particle with the best fitness value of all the particles as the gbest
    for each particle do
        Calculate particle velocity according to velocity update equation (5)
        Update particle position according to position update equation (6)
    end for
until maximum iterations or minimum error criteria is attained.
```

### D. Heuristic For Sensor Scheduling

As mentioned earlier, another objective of this paper is to schedule the sensor nodes such that the theoretical upper bound of network lifetime can be achieved.

To achieve this, we propose a weight-based method for determining the cover sets. It includes the following main steps:

1) Weight assignment
2) Cover formation
3) Cover optimization
4) Cover activation and Energy reduction.

**Algorithm 5 Heuristic for Sensor Node Scheduling**

```
Input M, B
Initialize k/Q, max_run, priority calculated using battery power
for r = 1 to max_run do
    for iteration = 1 to    mi=1 bi do
        if cover possibility exists then
            Determine cover based on priority
            Optimize cover
            Activate optimized cover and reduce battery power
        else
            break
        end if
    end for
    Calculate network lifetime (nlife)
    if nlife < U then
        Consider weight due to covered targets to compute priority to check for better lifetime
    else
        break
    end if
end for
```

1) **Weight Assignment:** Weight assignment is performed to decide the priority of sensor nodes. The more the weight of a sensor node, the higher the priority. Cover sets are decided based on this priority.

**2) Cover Formation:** A cover can be generated in different ways if the network has nodes which make all the targets k/Q covered.

**Algorithm 6 Cover Formation**

```
Input: Sorted S in descending order of assigned weight
Output: Cov_S
Initialize Cov_S = φ
for i = 1 to m do
    if Si contributes to coverage then
        Cov_S = Cov_S ∪ {Si}
    end if
    if coverage requirement met then
        break;
    end if
end for
```

**Cover Optimization**: Once the coverage requirement is met, the obtained cover set is optimized .By optimizing the generated cover, the proposed scheme attempts to minimize the energy usage. It should be noted that this is the second phase of redundancy elimination, the first one being at the cover formation.

**Algorithm 7 Cover Optimization**

```
Input: Cov_S
Output: Opt.Cov_S
Initialize Opt.Cov_S = φ
for i = length(Cov_S) down to 1 do
    if Cov_S − {Si.Cov_S} meets k/Q coverage requirement then
        Ignore Si.Cov_S
        Cov_S = Cov_S − {Si.Cov_S}
    else
        Opt.Cov_S = Opt.Cov_S ∪ {Si.Cov_S}
    end if
end for
```

### E. Cover Activation And Energy Reduction

The sensor nodes in the optimized cover are activated. The total energy that each node consumes should not fall beyond the minimum usable energy, Emin.

**Algorithm 8 Cover Activation and Energy Reduction**

```
Input: Opt.Cov_S
for i = 1 to length(Opt.Cov_S) do
    Si.state = true
    decrement bi
    if bi ≤ Emin then
        for j = 1 to n do
            M ij = 0 end for
```

### F. Experimental Results Analysis

The following **Table 5.1** describes experimental result for Heuristics, ABC, PSO algorithm and ACO algorithm in sensor deployment energy detection analysis. The table contains total number of wireless sensor node deployment and number of node count energy detection for Heuristics algorithm, number of node count energy detection for ABC algorithm, number of node count energy detection for PSO algorithm, number of node count energy detection for ACO algorithm details are shown.

The following **Fig 5.1** describes experimental result for Heuristics, ABC, PSO algorithm and ACO algorithm in sensor deployment energy detection analysis.

The figure contains total number of wireless sensor node deployment and number of node count energy detection for Heuristics algorithm, number of node count energy detection for ABC algorithm, number of node count energy detection for PSO algorithm, number of node count energy detection for ACO algorithm details are shown.

TABLE I. SENSOR DEPLOYMENTS (NODE ENERGY DETECTION)

| S.NO. | No. WSN NODE (n) | Heuristics (n) | ABC (n) | PSO (n) | ACO (n) |
|---|---|---|---|---|---|
| 1 | 50 | 32 | 26 | 19 | 15 |
| 2 | 100 | 74 | 65 | 55 | 50 |
| 3 | 150 | 127 | 115 | 98 | 95 |
| 4 | 200 | 168 | 151 | 142 | 134 |
| 5 | 250 | 207 | 196 | 185 | 175 |

The following **Table 1** describes experimental result for Heuristics, ABC, PSO algorithm and ACO algorithm in sensor deployment time interval analysis. The table contains total number of wireless sensor node deployment and number of node time taken for Heuristics algorithm, number of node time taken for ABC algorithm, number of node time taken for for PSO algorithm, number of node time taken for ACO algorithm details are shown.

## V. CONCLUSION

In this project is compute deployment locations for sensor nodes using artificial bee colony algorithm such that the network lifetime is maximum. Artificial bee colony algorithm performs better than PSO algorithm for this problem. In order to avoid the battery drain of all nodes at a time, sensor node scheduling can be done so that only minimum number of sensor nodes required for satisfying coverage requirement needs to be turned on. The other nodes can be reserved for later use. This method helps to prolong the network lifetime. A heuristic algorithm is powerful enough to schedule the sensor nodes in such a way that the network lifetime matches the theoretical upper bound of network lifetime. Network lifetime is extended by using this method of deploying at optimal locations such that it achieves maximum theoretical upper bound and then scheduling them so as to achieve the theoretical upper bound.

## REFERENCES

[1] J. Wang, R. Ghosh, and S. Das, “A survey on sensor localization,” J. Control Theory Appl., vol. 8, no. 1, pp. 2–11, 2010.

[2] C.-F. Huang and Y.-C. Tseng, “The coverage problem in a wireless sensor network,” in Proc. 2nd ACM Int. Conf. Wireless Sensor Netw. Appl., 2003, pp. 115–121.

[3] Z. Yun, X. Bai, D. Xuan, T. Lai, and W. Jia, “Optimal deployment patterns for full coverage and k-connectivity (k≤6) wireless sensor networks,” IEEE/ACM Trans. Netw., vol. 18, no. 3, pp. 934–947, Jun. 2010.

[4] G. Tan, S. Jarvis, and A.-M. Kermarrec, “Connectivity-guaranteed and obstacle-adaptive deployment schemes for mobile sensor networks,” in Proc. 28th Int. Conf. Distrib. Comput. Syst., Jun. 2008, pp. 429–437.

[5] S. Mini, S. K. Udgata, and S. L. Sabat, “A heuristic to maximize network lifetime for target coverage problem in wireless sensor networks,” Ad Hoc Sensor Wireless Netw., vol. 13, nos. 3–4, pp. 251–269, 2011

[6] Valeria Loscrí Enrico Natalizio Francesca Guerriero, “Particle Swarm Optimization Schemes Based on Consensus for Wireless Sensor Networks”, MSWiM’12, October 21–25, 2012, Paphos, Cyprus.

[7] Habib Mostafaei, Mehdi Esnaashari, and Mohammad Reza Meybodi , “A Coverage Monitoring algorithm based on Learning

Automata for Wireless Sensor Networks” , Comput. Sci. Technol., vol. 26, no. 1, pp. 117–129, 2010.

[8] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, “A comprehensive survey: Artificial bee colony (ABC) algorithm and applications,” Artif. Intell. Rev., 2012, pp. 1–37.

[9] D. Karaboga, S. Okdem, and C. Ozturk, “Cluster based wireless sensor network routing using artificial bee colony algorithm,” Wireless Netw., vol. 18, no. 7, pp. 847–860, 2012.

[10] Z. Yun, X. Bai, D. Xuan, T. Lai, and W. Jia, “Optimal deployment patterns for full coverage and k-connectivity (k≤6) wireless sensor networks,” IEEE/ACM Trans. Netw., vol. 18, no. 3, pp. 934–947, Jun. 2010.

[11] C.-Y. Chong and S. Kumar, “Sensor networks: Evolution, opportunities,and challenges,” Proc. IEEE, vol. 91, no. 8, pp. 1247–1256, Aug. 2003.

[12] M. Chaudhary and A. K. Pujari, “Q-coverage problem in wireless sensor networks,” in Proc. Int. Conf. Distrib. Comput. Netw., 2009,pp. 325–330.

[13] D. Karaboga and B. Akay, “A survey: Algorithms simulating bee swarm intelligence,” Artif. Intell. Rev., vol. 31,nos. 1–4, pp. 61–85, 2009.

[14] D. Karaboga and B. Basturk, “On the performance of artificial bee colony (ABC) algorithm,” Appl. Soft Comput., vol. 8, pp. 687–697, Jan.2008.

[15] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, “A comprehensive survey: Artificial bee colony (ABC) algorithm and applications,” Artif. Intell. Rev., 2012, pp. 1–37.