ISSN 2348-1218 (print)
International Journal of Interdisciplinary Research and Innovations ISSN 2348-1226 (online)
Vol. 6, Issue 2, pp: (314-320), Month: April - June 2018, Available at: www.researchpublish.com

Particle Swarm Optimization for Ad-Hoc Sensor Networks

Ms. Chaitali Shah, Ms. Sweta Dargad

Mathematics department, ITM Universe, Vadodara, India,
Computer Science Engineering, ITM Universe, Vadodara, India

Abstract: **Adhoc Sensor Networks consists of autonomous sensor nodes used for monitoring any environment. Issues that are faced in Adhoc Sensor Networks face are communication link failures, memory and computational constraints, and limited energy. To solve these constraints biological optimization techniques are used. A simple, effective, and computationally efficient optimization algorithm like Particle Swarm Optimization can be implemented to solve certain issues of Adhoc Sensor Networks. PSO can give solutions to optimal deployment, node localization, clustering, and data aggregation of the sensor nodes. We have presented a brief survey on how PSO can solve certain problems very efficiently. The multi-hop feature of an ad hoc network suggests the use of clustering to simplify routing and management. This paper presents a optimized solution to find the cluster head position for efficient communication of a True Positive alarm in monitoring applications.**

Keywords: **PSO, ACO, Sensor Networks, Optimization, Clusterhead.**

I. INTRODUCTION

Adhoc Sensor network is a wireless network consisting of spatially distributed autonomous devices using sensors to monitor physical or environmental conditions. Applications for areas including health care, utilities, and remote monitoring have been created. In health care, wireless devices make less invasive patient monitoring and health care possible. For utilities such as the electricity grid, streetlights, and water municipals, wireless sensors offer a lower-cost method for collecting system health data to reduce energy usage and better manage resources. Remote monitoring covers a wide range of applications where wireless systems can complement wired systems by reducing wiring costs and allowing new types of measurement applications. Remote monitoring applications include:

- Environmental monitoring of air, water, and soil
- Structural monitoring for buildings and bridges
- Industrial machine monitoring
- Process monitoring
- Asset tracking

Adhoc Sensor Networks monitors an environment by sensing its physical properties. It is a network of tiny, inexpensive autonomous nodes that can acquire, process, and transmit sensory data over wireless medium[4]. Adhoc Sensor Networks has become an area of intense research activity, earlier Adhoc networks were only intended for Military Operations because they had the limitation of infrastructure to be built for long distance or high-altitude surveillance[3]. Also Adhoc Sensor Networks is used in emergency response of some applications like estimating the fire in forests, environment control, tracking endangered animals etc. technical challenges include dense ad-hoc deployment, dynamic topology, spatial distribution, and constrains in bandwidth, memory, computational resources, and energy. A large number of wireless sensor nodes can be deployed in hostile areas without human involvement, e.g. by air-dropping from an aircraft for remote monitoring and surveillance purposes. Such airdropped nodes form adhoc sensor networks, as they don't have any fixed infrastructure. Once the sensor nodes are deployed on the ground, the data collected can be transmitted back to the base station to provide the necessary situational information.

The main problem faced by these deployed sensor nodes is limited energy storage and memory which might prevent them from relaying data directly to the base station. Clusters are formed consisting of nearby sensor nodes and a cluster head is elected on the basis of metrices like highest power or energy. (CHs) provide the transmission relay to base station such as a satellite. These sensor nodes are deployed with the help of some aircrafts which have limited payload. We need to drop the sensor nodes in the Region of Interest (ROI), so as to maximize the throughput of the sensor Network. Also the communication connectivity should be high so as reduce the use of Power consumed. Random placement of sensor nodes might increase the cost and time of communication.

These limitations provoke the planning placement of Sensor nodes so as to optimize the throughput of sensor network. After initial random drop of the sensor nodes the reorganization process with maximum possible utilization of sensors is desired. There are many algorithms used for deployment, data-fusion, Election of Cluster Head, Localization. The optimization algorithms like PSO, ACO and Genetic Algorithms are very popular to optimize the metrices like energy, power, memory, bandwidth etc.

There exist a lot of research work [5] related to the placement of sensor nodes in network topology design. Most of them focused on optimizing the location of the sensors in order to maximize their collective coverage. In another approach, particle swarm optimization has been applied to a routing protocol in order to also achieve energy efficiency. The work by Sarangi and Thankchan [12] shows how PSO is able to calculate cheaper energy routes, outperforming the genetic algorithm. Moreover, nodes deployment to maximize coverage has been considered by Jain and Sharma[11]. In their proposal, a PSO algorithm is used to modify a previous method, increasing an area coverage without spending more nodes.

In this paper we have attempted to solve the problem of coverage of all the sensor nodes coverage while considering energy efficiency using particle swarm optimization (PSO) algorithm. Algorithms like Genetic Algorithm and Ant Colony Optimization are also widely used to solve such problems but PSO gives good performance and leads to faster convergence than genetic algorithm used to solve the deployment optimization problem in [1].

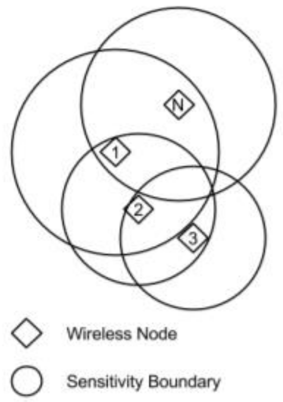

Figure 1. System Model of Cluster Formation of WSN Nodes

The solution presented here in this paper assumes that sensor nodes are mobile. An optimized solution for finding the cluster head position for efficient communication between the sensor nodes is desirable and attempted. The next section presents the introduction to PSO algorithms. Modeling of sensor network and the deployment algorithm is presented in section 3, followed by simulation results in section 4. Conclusion and future work is included in last section.

II. PSO

PSO is a population based stochastic optimization technique Developed by Dr. Eberhart and Dr. Kennedy in 1995[1], inspired by social behavior of bird flocking or fish schooling. PSO is an Artificial intelligence technique that can be used to find approximate solutions to extremely difficult or impossible numeric maximization and minimization problems.

The standard PSO algorithm works by having a population (called a swarm) of candidate solutions (called particles). The individuals, called, particles, are flown through the multidimensional search space with each particle representing a possible solution to the multidimensional problem. The movements of the particles are guided by their own best known position in the search space as well as the entire swarm's best known position. When improved positions are being discovered, they are used to guide the movements of the swarm. The process is repeated until that a satisfactory solution is discovered. Each Particle is searching for the optimum. Each particle is moving and hence has a velocity Each particle remembers the position it was in where it had its best result so far(Pbest). But this would not be be much good on its own; particles need help in figuring out where to search. The Particles in the swarm co-operate. They exchange information about what they have discovered in the places they have visited.

A particle knows the finiteness of those in its neighborhood, and uses the position of the one with best fitness. (global best).

This position is simply used to adjust the particle's velocity.

Formally, let f: Rn → R be the function which needs to be minimized. The function takes a candidate solution as argument in the form of a vector of real numbers, and then produces a real number as output, which indicates the objective function value of the given candidate solution. The gradient of f is not known. The goal is to find a solution A for which f(A) ≤ f(B) for all in the search space. That is, the solution A is the global minimum.

In each timestep, a particle has to move to a new position. It does this by adjusting its velocity using formule 1

Having worked out a new velocity, its position is simply its old position plus the new velocity. (formule 2)[2]

$$v_{k+1}^{i} = wv_{k}^{i} + c_1 r_1 (p_{k}^{i} - x_{k}^{i}) + c_2 r_2 (p_{k}^{g} - x_{k}^{i}) \quad (1)$$

$$x_{k+1}^{i} = x_{k}^{i} + v_{k+1}^{i} \quad (2)$$

where w is called the inertia weight.

v_{k+1}: Velocity of particle at $k+1$ th iteration

v_k : Velocity of particle at kth iteration

c_1 : acceleration factor related to pbest

c_2 : acceleration factor related to gbest

r1: random number between 0 and 1

r2: random number between 0 and 1

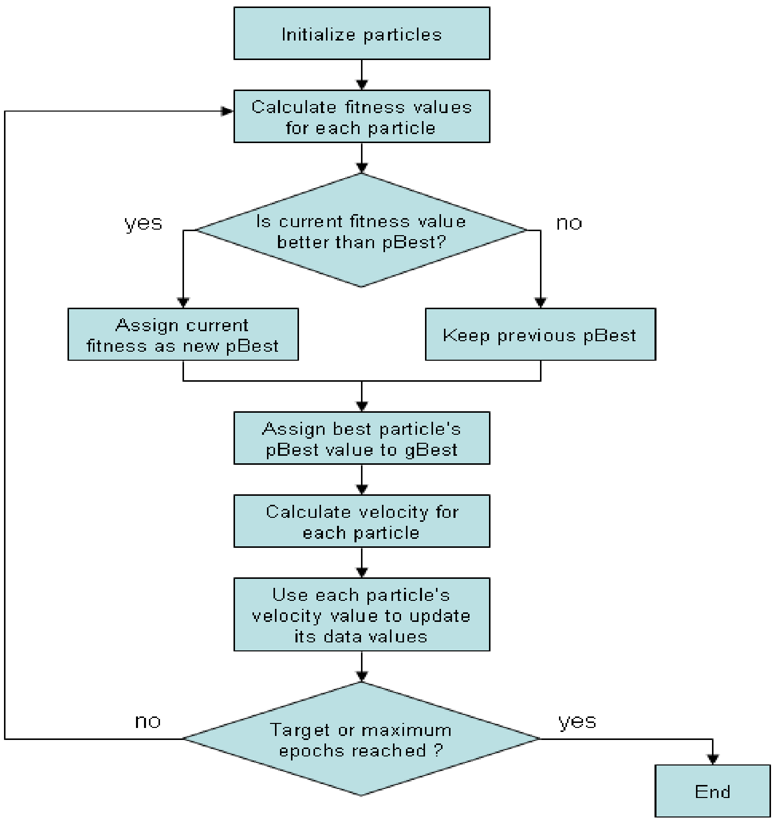

Figure 2. low Chart of PSO algorithm

III. PROPOSED WORKFLOW

The flow of PSO can briefly be described as follows. First, PSO initiates a group of particles, and assigns each particle a random initial position Xid and a random initial velocity Vid. Then PSO calculates the fitness value f for each particle. Note that the best position Pidb of each particle is its current position Xid in the first updating. The best position of the swarm Pgdb is the best position which achieves the best fitness value of the swarm. With the two bests, PSO updates the velocity and position for each particle, which are calculated according to the formula 1 and 2 respectively. Then PSO reevaluates the fitness value for each particle using its current velocity and position. With the fitness value of each particle, PSO updates the best position Pidb for each particle and the best position Pgdb for the swarm. Then PSO uses the two bests to update the velocity and position for each particle again, according to the formula 1 and 2 respectively. This procedure repeats until using up the number of iterations set at the beginning.

For each particle: Initialize particle. For each particle Calculate fitness value. If the fitness value is better than the best fitness value (pBest) in history. Set current value as the new pBest. For each particle Find in the particle neighborhood, the particle with the best fitness Calculate particle velocity according to the velocity equation (1). Apply the velocity constriction Update particle position according to the position equation (2) Apply the position constriction. While maximum iterations or minimum error criteria is not attained

We have set the default value for n=5 and m= 2 where n is Number of Particles and m is Number of Variables. We have also set the LB= [-5 -5] and UB =[5 5], where LB and UB are Upper Bound and Lower Bound. We have used this for minimization function. We have set maximum number of iteration to be maxite=20 and set maximum number of runs to be maxrun=10. The inertia weight is set to be w = 0.9. Now we run the code for run=1:maxrun

pso initialization

```
for i=1:n
  for j=1:m
    x0(i,j)=round(LB(j)+rand()*(UB(j)-LB(j)));
  end
end


x=x0;             % initial population
v=0.1*x0;               % initial velocity


for i=1:n
  f0(i,1)=ofun(x0(i,:));
end


[fmin0,index0]=min(f0);
pbest=x0;                  % initial pbest
gbest=x0(index0,:);        % initial gbest
```

pso algorithm

```
  ite=1;
tolerance=1;
```

International Journal of Interdisciplinary Research and Innovations ISSN 2348-1218 (print)
ISSN 2348-1226 (online)
Vol. 6, Issue 2, pp: (314-320), Month: April - June 2018, Available at: www.researchpublish.com

```
% update inertial weight
while ite<=maxite && tolerance>10^-12
 %w=wmax-(wmax-wmin)*ite/maxite;

 % pso velocity updates
 for i=1:n
  for   j=1:m
   v(i,j)=w*v(i,j)+rand()*(pbest(i,j)-x(i,j))+rand()*(gbest(1,j)-x(i,j));
  end
 end

 %      pso position update
 for i=1:n
  for j=1:m
   x(i,j)=x(i,j)+v(i,j);
  end
 end

 %      handling boundary violations
 for i=1:n
  for j=1:m
   if x(i,j)<LB(j)
    x(i,j)=LB(j);
   elseif x(i,j)>UB(j)
    x(i,j)=UB(j);
   end
  end
 end

 %evaluating fitness
 for i=1:n
  f(i,1)=ofun(x(i,:));
 end

 % updating pbest and fitness
 for i=1:n
```

```
  if f(i,1)<f0(i,1)
    pbest(i,:)=x(i,:);
    f0(i,1)=f(i,1);
  end
end

[fmin,index]=min(f0);   % finding out the best    particle
ffmin(ite,run)=fmin;     %       storing  best fitness
ffite(run)=ite;          %       storing  iteration count

%updating gbest and best fitness
if fmin<fmin0
  gbest=pbest(index,:);
  fmin0=fmin;
end

%calculating tolerance
if ite>100
  tolerance=abs(ffmin(ite-100,run)-fmin0);
end

% displaying iterative results
if ite==1
  disp(sprintf('Iteration   Best particle Objective fun'));
end

disp(sprintf('%8g  %8g %8.4f',ite,index,fmin0));
ite=ite+1;
end
```

A. Optimization of Energy Consumption

After optimization of coverage, all the deployed sensor nodes move to their own positions. Now we can disregard the assumption of sensor mobility since our goal is to minimize energy usage in a cluster based sensor network topology by finding the optimal cluster head (CH) positions. For this purpose, we assume a power consumption model [10] for the radio hardware energy dissipation where the transmitter dissipates energy to run the radio electronics and the power amplifier, and the receiver dissipates energy to run the radio electronics. This is one of the most widely used models in sensor network simulation analysis. For our approach, both the free space (distance2 power loss) and the multi-path fading (distance 4 power loss) channel models were used. Assume that the sensor nodes inside a cluster have short distance dis to cluster head but each cluster head has long distance Dis to the base station. Thus for each sensor node inside a cluster, to transmit an l-bit message a distance dis to cluster head, the radio expends

IV. CONCLUSION AND RESULTS

In this paper we discussed the application of PSO algorithm to optimize the coverage of ad hoc sensor network. The Sensor Network deployment and energy consumption in cluster-based topology is discussed here. We have used PSO to identify the Cluster Head based on the energy dissipation. As the first optimization objective to place the sensors with mobility, and a distance based energy model to reduce cost based on clustering method. The PSO technique was simulated over distributed nodes to find the node with the highest processing load. The simulation results show that PSO algorithm has faster convergence rate and while demonstrating good performance. In the future work, we will take the uncertainty in the position of the sensors due to the initial random deployment into account.

REFERENCES

[1] Kennedy and R. C. Eberhart: Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks, Perth, Australia (1995) 1942-1948

[2] Xiaoling, Wu, et al. "Swarm based sensor deployment optimization in ad hoc sensor networks." International Conference on Embedded Software and Systems. Springer, Berlin, Heidelberg, 2005.

[3] Damien B. Jourdan, Olivier L. de Weck: Layout optimization for a wireless sensor network using a multi-objective genetic algorithm. IEEE 59th Vehicular Technology Conference (VTC 2004-Spring), Vol.5 (2004) 2466-2470

[4] Particle Swarm Optimization in Wireless-Sensor Networks: A Brief Survey

[5] K. Chakrabarty, S. S. Iyengar, H. Qi and E. Cho: Grid coverage for surveillance and target location in distributed sensor networks. IEEE transactions on computers, Vol.51 (2002) 1448-1453

[6] Tillett, J. Rao, R. Sahin, F. , "Cluster-head identification in ad hoc sensor networks using particle swarm optimization," IEEE int. conf. on Personal Wireless Communications Proc., 2002,pp. 201- 205

[7] D. Turgut, S.K. Das, R. Elmasri, and B. Turgut, "Optimizing Clustering Algorithm in Mobile Ad hoc Networks Using Genetic Algorithmic Approach", Proceedings of IEEE GLOBECOM 2002, Taipei, Taiwan. 2002.

[8] Kennedy and R. C. Eberhart, "Particle Swarm Optimization", Pme edings of IEEE International Conference on Neural Networks, Perth, Australia. pp. 1942- 1948,1995

[9] Hu, Bing, Hongsheng Li, and Sumin Liu. "Localization Algorithm of Mobile Node in WSN Based on Monte Carlo." Intelligent Networks and Intelligent Systems, 2009. ICINIS'09. Second International Conference on. IEEE, 2009.

[10] Xiaoling, Wu, et al. "Swarm based sensor deployment optimization in ad hoc sensor networks." International Conference on Embedded Software and Systems. Springer, Berlin, Heidelberg, 2005.

[11] Jain, N., & Sharma, K., (2013) Modified Discrete Binary PSO based Sensor Placement for Coverage in WSN Networks, International Journal of Electronics and Computer Science Engineering (IJECSE).

[12] Sarangi, S., &Thankchan, B. (2012).A Novel Routing Algorithm for Wireless Sensor Network Using Particle Swarm Optimization. IOSR Journal of Computer Engineering (IOSRJCE), ISSN, 2278-0661.