IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 4, Issue 04, November 2015)

IJDACR
ISSN: 2319-4863

Optimization of Stable Election Protocol through Genetic Algorithm & Particle Swarm Optimization in clustered wireless sensor network

Khushboo Pawar

Abstract — **we study the impact of heterogeneity of nodes, in terms of their energy, in wireless sensor networks that are hierarchically clustered. In these networks some of the nodes become cluster heads, aggregate the data of their cluster members and transmit it to the sink. We assume that a percentage of the population of sensor nodes is equipped with additional energy resources—this is a source of heterogeneity which may result from the initial setting or as the operation of the network evolves.**

Classical clustering protocols assume that all the nodes are equipped with the same amount of energy and as a result, they cannot take full advantage of the presence of node heterogeneity. We studied SEP, a heterogeneous-aware protocol to prolong the time interval before the death of the first node (we refer to as stability period), which is crucial for many applications where the feedback from the sensor network must be reliable.

also the final Optimization of SEP protocol is done through Genetic Algorithm and Particle Swarm Optimization, and found that PSO on SEP yields longer stability region for higher values of extra energy brought by more powerful nodes as Compare to SEP and SEP-GA..

Keywords — **LEACH, Stable Election Protocol, Particle Swarm Optimization, Genetic Algorithm.**

I. INTRODUCTION

Motivation: Wireless Sensor Networks are networks of tiny, battery powered sensor nodes with limited on-board processing, storage and radio capabilities [1]. Nodes sense and send their reports toward a processing center which is called "sink". The design of protocols and applications for such networks has to be energy aware in order to prolong the lifetime of the network, because the replacement of the embedded batteries is a very difficult process once these nodes have been deployed. Classical approaches like Direct Transmission and Minimum Transmission Energy [2] do not guarantee well balanced distribution of the energy load among nodes of the sensor network. Using Direct Transmission (DT), sensor nodes transmit directly to the sink, as a result nodes that are far away from the sink would die first [3]. On the other hand, using Minimum Transmission Energy (MTE), data is routed over minimumcost routes, where cost reflects the transmission power expended. Under MTE, nodes that are near the sink act as relays with higher probability than nodes that are far from the sink. Thus nodes near the sink tend to die fast. Under both

DT and MTE, a part of the field will not be monitored for a significant part of the lifetime of the network, and as a result the sensing process of the field will be biased. A solution proposed in [4], called LEACH, guarantees that the energy load is well distributed by dynamically created clusters, using cluster heads dynamically elected according to a priori optimal probability.

Cluster heads aggregate reports from their cluster members before forwarding them to the sink. By rotating the clusterhead role uniformly among all nodes, each node tends to expend the same energy over time.

Most of the analytical results for LEACH-type schemes are obtained assuming that the nodes of the sensor network are equipped with the same amount of energy—this is the case of homogeneous sensor networks. In this paper we study the impact of heterogeneity in terms of node energy. We assume that a percentage of the node population is equipped with more energy than the rest of the nodes in the same network— this is the case of heterogeneous sensor networks. We are motivated by the fact that there are a lot of applications that would highly benefit from understanding the impact of such heterogeneity. One of these applications could be the re-energization of sensor networks. As the lifetime of sensor networks is limited there is a need to re-energize the sensor network by adding more nodes. These nodes will be equipped with more energy than the nodes that are already in use, which creates heterogeneity in terms of node energy. Note that due to practical/cost constraints it is not always possible to satisfy the

International of Digital Application & Contemporary research Journal
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

constraints for optimal distribution between different types of nodes as proposed in [5].

II. HETEROGENEOUS WSN MODEL

In this section we describe our model of a wireless sensor network with nodes heterogeneous in their initial amount of energy. We particularly present the setting, the energy model, and how the optimal number of clusters can be computed. Let us assume the case where a percentage of the population of sensor nodes is equipped with more energy resources than the rest of the nodes. Let m be the fraction of the total number of nodes n, which are equipped with α times more energy than the others.

We refer to these powerful nodes as advanced nodes, and the rest (1-m)x*n* as normal nodes. We assume that all nodes are distributed uniformly over the sensor field.

Clustering Hierarchy

We consider a sensor network that is hierarchically clustered. The LEACH (Low Energy Adaptive Clustering Hierarchy) protocol [3] maintains such clustering hierarchy. In LEACH, the clusters are re-established in each "round."

New cluster heads are elected in each round and as a result the load is well distributed and balanced among the nodes of the network. Moreover each node transmits to the closest cluster head so as to split the communication cost to the sink (which is tens of times greater than the processing and operation cost.) Only the cluster head has to report to the sink and may expend a large amount of energy, but this happens periodically for each node. In LEACH there is an optimal percentage Popt (determined a priori) of nodes that has to become cluster heads In each round assuming uniform distribution of nodes in space [3], [4], [6], [7].

If the nodes are homogeneous, which means that all the nodes in the field have the same initial energy, the LEACH protocol guarantees that everyone of them will become a cluster head exactly once every 1/Popt rounds. Throughout this paper we refer to this number of rounds, 1/Popt, as epoch of the clustered sensor network.

Initially each node can become a cluster head with a probability Popt On average, nxPopt nodes must become cluster heads per round per epoch. Nodes that are elected to be cluster heads in the current round can no longer become cluster heads in the same epoch. The non-elected nodes belong to the set G and in order to maintain a steady number of cluster heads per round, the probability of nodes €G to become a cluster head increases after each round in the same epoch. The decision is made at the beginning of each round by each node S€G independently choosing a random number in [0,1]. If the random number is less than a threshold T(s) then the node becomes a cluster head in the current round. The threshold is set as:

Where r is the current round number (starting from round 0.) The election probability €G of nodes to become cluster heads increases in each round in the same epoch and becomes equal to 1 in the last round of the epoch. Note that by round we define a time interval where all cluster members have to transmit to their cluster head once. We show in this paper how the election process of cluster heads should be adapted appropriately to deal with heterogeneous nodes, which means that not all the nodes in the field have the same initial energy

Optimal Clustering

Previous work have studied either by simulation [3], [4] or analytically [6], [7] the optimal probability of a node being elected as a cluster head as a function of spatial density when nodes are uniformly distributed over the sensor field. This clustering is optimal in the sense that energy consumption is well distributed over all sensors and the total energy consumption is minimum. Such optimal clustering highly depends on the energy model we use. For the purpose of this study we use similar energy model and analysis as proposed in [4]. According to the radio energy dissipation model illustrated in Figure 1, in order to achieve an acceptable Signal-to-Noise Ratio (SNR) in transmitting an L bit message over a distance

III. PARTICLE SWARM OPTIMIZATION

Introduction

Here we use Swarm Optimization for further optimization of results obtained from SEP protocol. Particle swarm optimization has become a common heuristic technique in the optimization community, with many researchers exploring the concepts, issues, and applications of the algorithm. In spite of this attention, there has as yet been no standard definition representing exactly what is involved in modern implementations of the technique.

The original PSO algorithm was inspired by the social behaviour of biological organisms, specifically the ability of groups of some species of animals to work as a whole in locating desirable positions in a given area, e.g. birds flocking to a food source. This seeking behaviour was associated with that of an optimization search for solutions to non-linear equations in a real-valued search space. In the most common implementations of PSO, particles move through the search space using a combination of an attraction to the best solution that they individually have found, and an attraction to the best solution that any particle in their neighbourhood has found.

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

In PSO, a neighbourhood is defined for each individual particle as the subset of particles which it is able to communicate with. The first PSO model used a Euclidian neighbourhood for particle communication, measuring the actual distance between particles to determine which were close enough to be in communication. This was done in imitation of the behaviour of bird flocks, similar to biological models where individual birds are only able to communicate with other individuals in the immediate vicinity [5], [6]. The Euclidian neighbourhood model was abandoned in favour of less computationally intensive models when research focus was shifted from biological modelling to mathematical optimization. Topological neighbourhoods unrelated to the locality of the particle came into use, including what has come to be known as a global neighbourhood, or gbest model, where each particle is connected to and able to obtain information from every other particle in the swarm. *Algorithm 1*

Particle Swarm Algorithm

01. Begin
02. Parameter settings and swarm initialization
03. Evaluation
04. g = 1
05. While (the stopping criterion is not met) do
06. For each particle
07. Update velocity
08. Update position and local best position
09. Evaluation
10. End For
11. Update leader (global best particle)
12. g + +
15. End While
14. End

The PSO algorithm has several phases consist of Initialization, Evaluation, Update Velocity and Update Position.

Update velocity and position

In each iteration, each particle updates its velocity and position according to its heretofore best position, its current velocity and some information of its neighbours. Equation 5 is used for updating the velocity:

$$\overrightarrow{v_l(t)} = \underbrace{w\overrightarrow{v_l(t-1)}}_{inertia} + \underbrace{c_1 r_1 \left(x_l^{\#}(t-1) - \overrightarrow{x_l(t-1)} \right)}_{Personalinfluence} + \underbrace{c_2 r_2 \left(\overrightarrow{x^*(t-1)} - \overrightarrow{x_l(t-1)} \right)}_{Socialinfluence}$$

IV. GENETIC ALGORITHM

In the computer science field of artificial intelligence, a genetic algorithm (GA) is a search heuristic that mimics the process of natural evolution. This heuristic is routinely used to generate useful solutions to optimization and search problems. Genetic algorithms belong to the larger class of evolutionary algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover.

Methodology

In a genetic algorithm, a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem, evolves toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm. Commonly, the algorithm terminates when either a maximum number of generations has been produced, or a satisfactory fitness level has been reached for the population. If the algorithm has terminated due to a maximum number of generations, a satisfactory solution may or may not have been reached.

A typical genetic algorithm requires:

A genetic representation of the solution domain, A fitness function to evaluate the solution domain.

Initialization

Initially many individual solutions are (usually) randomly generated to form an initial population. The population size depends on the nature of the problem, but typically contains several hundreds or thousands of possible solutions. Traditionally, the population is generated randomly, allowing the entire range of possible solutions (the search space). Occasionally, the solutions may be "seeded" in areas where optimal solutions are likely to be found.

Selection

During each successive generation, a proportion of the existing population is selected to breed a new generation. Individual solutions are selected through a fitness-based process, where fitter solutions (as measured by a fitness function) are typically more likely to be selected. Certain selection methods rate the fitness of each solution and preferentially select the best solutions. Other methods rate only a random sample of the population, as the former process may be very time-consuming.

International Journal of Digital Application & Contemporary research

IJDACR
International Journal Of Digital Application & Contemporary Research

Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

Reproduction
The next step is to generate a second generation population of solutions from those selected through genetic operators: crossover (also called recombination), and/or mutation.

For each new solution to be produced, a pair of "parent" solutions is selected for breeding from the pool selected previously. By producing a "child" solution using the above methods of crossover and mutation, a new solution is created which typically shares many of the characteristics of its "parents". New parents are selected for each new child, and the process continues until a new population of solutions of appropriate size is generated. Although reproduction methods that are based on the use of two parents are more "biology inspired", some research suggests that more than two "parents" generate higher quality chromosomes.

These processes ultimately result in the next generation population of chromosomes that is different from the initial generation. Generally the average fitness will have increased by this procedure for the population, since only the best organisms from the first generation are selected for breeding, along with a small proportion of less fit solutions, for reasons already mentioned above.

Although Crossover and Mutation are known as the main genetic operators, it is possible to use other operators such as regrouping, colonization-extinction, or migration in genetic algorithms.

Termination
This generational process is repeated until a termination condition has been reached. Common terminating conditions are:

- A solution is found that satisfies minimum criteria.
- Fixed number of generations reached.
- Allocated budget (computation time/money) reached
- The highest ranking solution's fitness is reaching or has reached a plateau such that successive iterations no longer produce better results
- Manual inspection
- Combinations of the above

Simple generational genetic algorithm procedure

- Choose the initial population of individuals
- Evaluate the fitness of each individual in that population
- Repeat on this generation until termination (time limit, sufficient fitness achieved, etc.):
- Select the best-fit individuals for reproduction
- Breed new individuals through crossover and mutation operations to give birth to offspring
- Evaluate the individual fitness of new individuals • Replace least-fit population with new individuals.

IV. RESULTS

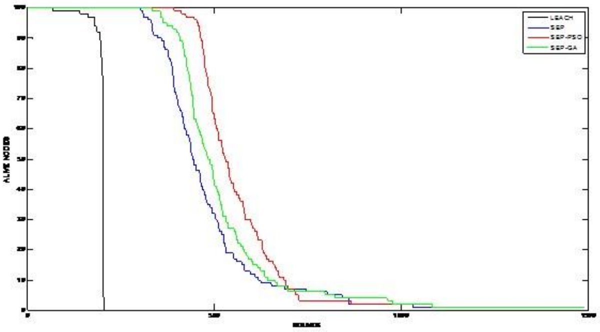

Figure above shows that the Stability of used Protocols in Wireless Sensor network and alive node statistics with respect to number of rounds

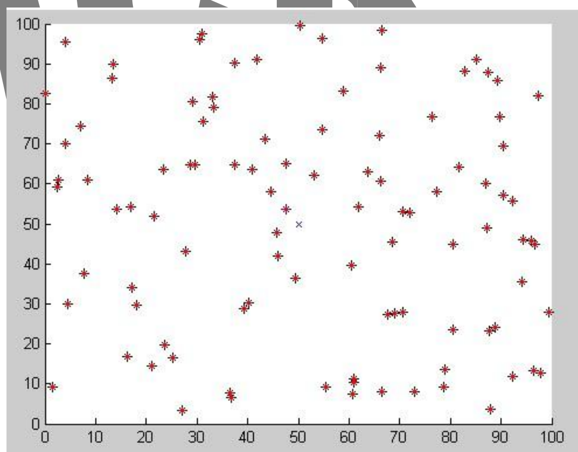

Figure above shows the random arrangement of 100 sensor nodes in 100x100 sensor field of wireless sensor network with base station is at the centre of sensor field.

IV. CONCLUSIONS

We studied SEP (Stable Election Protocol) so every sensor node in a heterogeneous two-level hierarchical network independently elects itself as a cluster head based on its initial energy relative to that of other nodes. Unlike [8], we do not require any global knowledge of energy at every election round. Unlike [10], [5], SEP is dynamic in that we do not assume any prior distribution of the different levels of energy in the sensor nodes. Furthermore, our analysis of SEP is not

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

only asymptotic, i.e. the analysis applies equally well to small sized networks. Finally SEP is scalable as it does not require any knowledge of the exact position of each node in the field.

We are currently extending SEP to deal with clustered sensor networks with more than two levels of hierarchy and more than two types of nodes. We are also implementing SEP in Berkeley/Crossbow motes and examining deployment issues including dynamic updates of weighted election probabilities based on current heterogeneity conditions as well as the integration of SEP with MAC protocols that can provide low cost information about the distribution of energy in the vicinity of each node [13].

Further Optimization of SEP protocol with Genetic algorithm and particle swarm optimization shows that PSO on SEP yields longer stability region for higher values of extra energy brought by more powerful nodes as Compare to SEP and SEP-GA.

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” IEEE Communications Magazine, vol. 40, no. 8, pp. 102–114, August 2002.

[2] T. J. Shepard, “A channel access scheme for large dense packet radio networks,” in Proccedings of ACM SIGCOMM, September 1996, pp. 219–230.

[3] W. R. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, “Energyefficient communication protocol for wireless microsensor networks,” in Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS-33), January 2000.

[4] “An application-specific protocol architecture for wireless microsensor networks,” IEEE Transactions on Wireless Communications, vol. 1, no. 4, pp. 660–670, October 2002.

[5] V. Mhatre and C. Rosenberg, “Homogeneous vs. heterogeneous clustered sensor networks: A comparative study,” in Proceedings of 2004 IEEE International Conference on Communications (ICC 2004), June 2004.

[6] S. Bandyopadhyay and E. J. Coyle, “An energy efficient hierarchical clustering algorithm for wireless sensor networks,” in Proceedings of INFOCOM 2003, April 2003.

[7] “Minimizing communication costs in hierarchically-clustered networks of wireless sensors,” Computer Networks, vol. 44, no. 1, pp. 1–16, January 2004.

[8] W. R. Heinzelman, “Application-Specific Protocol Architectures for Wireless Networks,” Ph.D. thesis, Massachusetts Institute of Technology, 2000.

[9] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, “PSFQ: a reliable transport protocol for wireless sensor networks,” in Proceedings of the 1st ACM international workshop on Wireless Sensor Networks and Applications (WSNA’02), July 2002.

[10] E. J. Duarte-Melo and M. Liu, “Analysis of energy consumption and lifetime of heterogeneous wireless sensor networks,” in Proceedings of Global Telecommunications Conference (GLOBECOM 2002). IEEE, November 2002.

[11] K. Kalpakis, K. Dasgupta, and P. Namjoshi, “Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks,” Computer Networks, vol. 42, no. 6, pp. 697–716, 2003.

[12] H. O. Tan and I. Korpeoglu, “Power efficient data gathering and aggregation in wireless sensor networks,” SIGMOD Record, vol. 32, no. 4, pp. 66–71, 2003.

[13] W. S. Conner, J. Chhabra, M. Yarvis, and L. Krishnamurthy, “Experimental evaluation of synchronization and topology control for in building sensor network applications,” in Proceedings of the 2nd ACM international conference on Wireless Sensor Networks and Applications (WSNA’03), September 2003, pp. 38–49.