

Optimization of Relative Weight Based Clustering Using Genetic Algorithmic Approach

Sharmila Anand John Francis, Elijah Blessing Rajsingh and Giss George
Department of Computer Science and Engineering, School of Computer Science and Technology,
Karunya University, Coimbatore, Tamil Nadu, India

Abstract: Optimization always plays a vital role to enhance the performance of the network. In this study, optimization is done with the recently proposed Relative Weight Based Clustering (RWC) algorithm to achieve minimum number of cluster heads in the network. The relative weight based clustering selects cluster heads based on the relative weights of the nodes. The optimal choosing of cluster heads enhances the performance of network by reducing the reaffiliation count and cluster head change. This study uses Genetic Algorithm (GA) to optimize relative weight based clustering based on the fitness function. The goal of genetic algorithm is to choose the one with highest fitness value as the best chromosome. Simulation study is carried out with the genetic algorithm optimization tool, LibGA and the improved performance metrics are shown for the optimized RWC and original RWC algorithm that enhances the stability of the network.

Key words: Genetic algorithm, optimization, cluster head selection process, relative weight, simulation parameter, India

INTRODUCTION

Mobile Ad hoc networks play an important role in places, where there is lack of availability of wired backbone or fixed infrastructure. The clustered network organization contains many clusters with one cluster head node for each cluster. The cluster heads are vested with additional responsibilities of coordinating, monitoring, allocating resources to its members, inter-cluster and intra-cluster communication among the nodes in the network (Sharmila and Elijah, 2008). The set of cluster heads is known as the dominant set. A dominating set is a subset of nodes in the network that assures that every node is either in the subset or a neighbor of a node in the subset (Chatterjee *et al.*, 2002). Due to the mobility nature of the mobile nodes, frequent change of cluster heads occurs that perturbs the stability of the network and affecting other protocols that rely on it and incurring high computation overhead caused due to information exchange among the nodes. Therefore, it is desirable to choose optimum number of cluster heads in the network. Due to the optimal selection of cluster heads is an NP-hard problem; various evolutionary methods are used to solve this problem. This study uses genetic algorithm to optimize the relative weight based clustering to enhance the stability of the network (Sharmila and Elijah, 2009).

The genetic algorithm is a search technique to compute exact or approximate solutions to the optimization and search problems (Chambers, 1995). GA performs a random search of a defined N-dimensional solution space. These algorithms search or operate on a given population of potential solutions. Every member of a population is associated with the certain fitness value that represents degree of correctness or quality of solution. The genetic algorithm is used to optimize many clustering protocols (Turgut *et al.*, 2002). In Turgut *et al.* (2002), the cluster head election procedure of WCA (Chatterjee *et al.*, 2002) is optimized using genetic algorithm by computing the fitness value by taking the summation of weights of all cluster heads. The goal of the genetic algorithm is to select lowest fitness value chromosome as the best suited one. This study uses genetic algorithms as an optimization technique to optimize the cluster head selection process of recently proposed relative weight based clustering algorithm to achieve optimal number of cluster heads in the network.

Optimizing relative weight based clustering algorithm:

The relative weight based clustering selects cluster heads based on the relative weights of nodes. To select stable cluster heads, five crucial factors viz. interest level, battery power, memory availability, mobility and node-degree that greatly influence the cluster head

selection are identified. The relative weights of the mobile nodes are computed using AHP methodology and nodes are sorted in descending order according to the computed weights and stored in sort-list data structure. The highest weight node is selected as a cluster head and its neighbors become its members and it is removed from the sort-list to restrict its participation in further cluster head selection process. The sort-list is traversed till the end until the whole network is clustered with all the nodes is either a cluster head or a cluster member in the network. The optimization of RWC clustering algorithm is done to minimize the number of cluster heads in the network. To minimize the number of cluster heads, each cluster head serves maximum number of nodes within their clusters. The goal of genetic algorithm is to choose the one with highest fitness value as the best chromosome.

GA operations: The following genetic operations (Chambers, 1995; Goldberg, 1953) are used to optimize the cluster head selection process of RWC.

Encoding the data: It is a string representation of the given data, which would be the mobile nodes in the network. Each mobile node has unique node ID for unique representation of nodes. The genotype is a string of N positions; each position corresponds to a particular object. The number of nodes N are randomly generated and unique node IDs are used encode the chromosome using integer permutation.

Initial population: The population is randomly generated and is equal to the number of nodes in the network.

Selection: The GA selects the cluster heads for each chromosome and computes the fitness value for each chromosome. Due to the different set of cluster heads for each chromosome, the fitness value is different for each chromosome. According to the fitness values, Roulette Wheel method is used for selection of chromosomes.

Crossover: It is an indispensable operation to have more diverse population and depends on the rate specified that is best suited for a specific application and are found experimentally. This study uses X order 1 method with the specific crossover rate that inherits the elements between the two crossover points (inclusive) from the selected parent in the same order and position as they appeared in that parent. The remaining elements are inherited from the alternate parent in the same order as they appear, beginning with the first position following the second crossover point and skipping over all elements that are already present in the offspring.

Mutation: It avoids premature convergence by occasional random alternation of randomly determined bit in the

given string with a specified rate. The swap method is used with the specific mutation rate that randomly selects two genes at two positions and swaps them to create the new child.

Replacement: The use of append method is to save the best strings into the next generation as the new set of solutions are replaced with the old solutions during the reproduction process. If it is not saved, there may lose of best strings.

Elitism: If the new solution is better than the previous one, it updates the current solution with the new solution.

Computation of fitness value: The fitness value is computed for each chromosome and it is explained in fitness algorithm in the study. Based on the fitness value, the best suited chromosome is selected.

Applying genetic algorithms to relative weight based clustering: The GA operations are applied to relative weight based clustering to obtain optimal number of cluster heads in the network. The nodes with their node IDs along with their neighbor list and their corresponding weight values computed from relative weight clustering protocol are stored separately in a data_list (Chatterjee *et al.*, 2002; Turgut *et al.*, 2002) that eases the computation of fitness function. The genetic algorithm steps that are applied RWC.

Initial population: Randomly generate the initial population with the pool size being equal to the number of nodes in the network. This will produce the same number of chromosomes in the form of integer strings.

Repeat until requirements met: While new pool size < old pool size, repeat steps 3-7. Repeat step 2 until the number of generation or the convergence is met.

Selection: Apply roulette wheel method with fitness values.

Crossover method: X order1 method with 0.02 crossover rate.

Mutation: The swap method with 0.08 mutation rate is used.

Compute objective function: Compute the fitness function N-H for each chromosome in the population. Where, N is total number of nodes in the network and H is the number of selected cluster heads. The chromosome with the highest fitness value is the best suited chromosome.

Replacement: The append method is used to save the best strings for next generation.

Elitism: Check if the new children are better than the current best. If so, replace the best by the child.

The unique ID for N number of nodes in the network is randomly generated in the range from 1 to N. These node IDs of all N nodes are encoded in a single chromosome using integer permutation as string of integers. The initial population is randomly generated to the pool size. To achieve the completeness and uniqueness property, no duplication of node IDs appear and the order of placing of node IDs are randomly encoded in the chromosome. Each chromosome is traversed from the beginning till the end to obtain cluster heads in the network that are stored in corresponding Cluster_head_list. Each node ID is checked whether, it is not a cluster head or cluster member of another cluster head and its weight (from the already computed value in Data_list) is higher than its neighbors for particular predefined transmission range. If all these three conditions are satisfied, the node is selected as the cluster head and it is inserted into Cluster_head_list and its neighbors become its cluster members and they are restricted to participate in further cluster head selection process. After traversing each chromosome from the beginning till the end, each node in the chromosome should be either a cluster head or a member obtaining total cluster heads in the network stored in Cluster_head_list. Then, the fitness value is computed for the chromosome by finding out the total number of cluster heads in Cluster_Head_list (H) and subtracting it from total number of nodes in the Network (N). Due to the different order of placing node IDs in different chromosome, different Cluster_head_list are obtained for each chromosome thereby obtaining different fitness values. The study shows the computation of fitness algorithm. Based on the fitness values, the highest fitness value chromosome are selected as the best suited chromosome.

Computation of fitness algorithm

Step 1: The fitness value is equal to 0 at the beginning.

Step 2: Repeat step 3 for each gene in the chromosome.

Step 3: Assign node to be equal to gene (Turgut *et al.*, 2002; Chambers, 1995). If a node is not a cluster head and is not a member of another cluster head and weight is higher than its neighbors, select the node as a cluster head and insert its node ID to the Cluster_head_list.

Step 4: If all the nodes in the chromosome are either a cluster head or a member of a cluster, the total number of selected cluster heads is obtained in Cluster_head_list for the chromosome. Compute the fitness value N-H or the chromosome.

MATERIALS AND METHODS

With the use of NS2 simulator with CMU wireless extension and by adopting random way point mobility model to the mobile nodes, various performance metrics are studied and it is compared with previous relative weight based clustering algorithm. The number of mobile nodes in the network is varied between 20-60 nodes and the transmission ranges from 50-300 m with the maximum speed of the mobile nodes as 20 m sec⁻¹. Table 1 shows the simulation parameters of the network.

The LibGA genetic algorithm tool is used to optimize the number of cluster heads in relative weight based clustering algorithm. LibGA is a library of routines written in C language for developing genetic algorithms. The GA parameters are set or modified using configuration file that are not necessary for compilation. The data file is created that contains node ID and its neighbor list along with the weights that are already computed from RWC algorithm and it is invoked during the execution of program. The configuration file could be changed according to the user requirement and the following information is set.

Configuration information:

- User data: n20.ahp
- Data type: Integer permutation
- Init pool entered: Random
- Chromosome length: Number of nodes in the network
- Number of trials: Run until convergence

The fitness function is coded in C language and incorporated into LibGA tool and the routines are executed until the convergence of the genetic algorithm is met. The best set of cluster heads is obtained.

Table 1: Simulation parameters

Parameters	Values
Simulation dimension	1000×1000 m
Simulation time	60 sec
Number of nodes	20, 40 and 60 nodes
Transmission range of mobile nodes	50, 100, 150, 200, 250, 300 m
Pause time	5 sec
Maximum speed of mobile nodes	20 m sec ⁻¹
Hello packet broadcast interval	0.4 sec

RESULTS AND DISCUSSION

The relative weight based clustering is optimized using GA to obtain optimal number of cluster heads in the network. The various performance metrics are studied and compared with the original RWC algorithm by varying the number of nodes and its transmission ranges. The metrics considered for evaluation are:

- Cluster number
- Reaffiliation count
- Cluster head changes

The cluster number is the number of cluster heads in the network. The optimum number of cluster heads enhances the stability of the network. Reaffiliation count is the count of number of nodes that gets detaches from one cluster head and attaches with another cluster head under the current dominating set reaffiliation count should be less to reduce the information exchange due to reaffiliation. The cluster head change occurs if two cluster heads comes into the transmission range of each other. The highest weight cluster head takes up the role of a cluster head and the other gives up its role leading to the cluster head change. Frequent cluster head change degrades the stability of the network. Therefore, to achieve the target of stability of the network, the network should contain optimum cluster number with reduced reaffiliation count and cluster head changes.

Figure 1 shows the comparison of cluster number for original RWC and Optimized RWC (ORWC) for different number of nodes by varying transmission ranges. The cluster number is minimized and optimal number of cluster heads are obtained in optimized RWC.

Figure 2 shows the reaffiliation count for RWC and optimized RWC. The reaffiliation count tends to decrease for optimized RWC as the cluster number is minimized. It increases for low transmission ranges and reaches a maximum for optimal transmission range and decreases with further increase of transmission ranges.

Due to the mobility nature of the nodes, the nodes tends to move out of the cluster very frequently for low transmission ranges causing high reaffiliation count and it decreases as the nodes stay inside the cluster for a long time as the coverage area of cluster head is more for high transmission ranges.

Figure 3 shows the comparison of cluster head changes for RWC and optimized RWC. Due to the optimal selection of cluster heads in optimized RWC, cluster head changes are reduced drastically than RWC and MobDHop (Er and Seah, 2004; Sharmila and Eljiah, 2009).

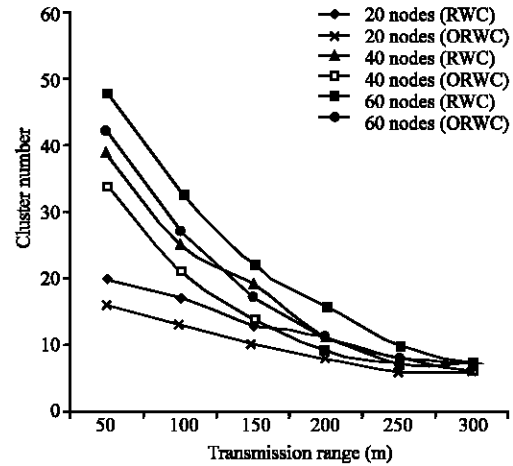


Fig. 1: Cluster number

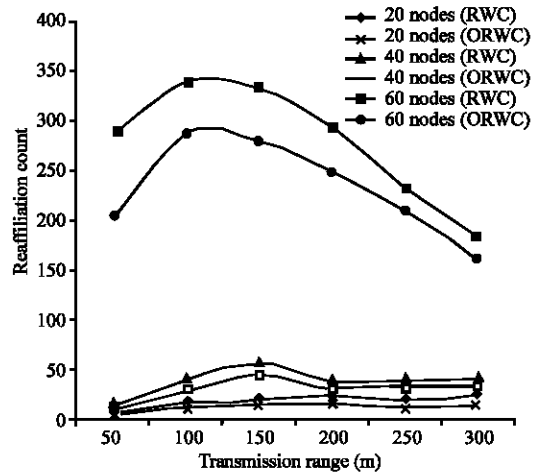


Fig. 2: Reaffiliation count

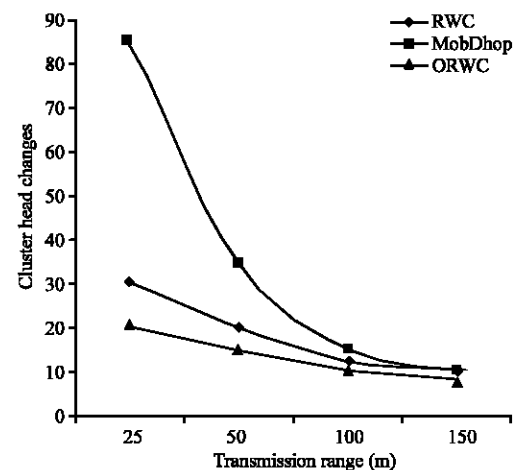


Fig. 3: Cluster head changes

Since, the stability of the network always depends on the cluster head changes, the high cluster head changes leads

to low stability of the network. Therefore, the optimization of RWC selects optimal cluster heads that reduces the cluster head changes thereby enhancing the stability of the network to a great extent.

CONCLUSION

The novel relative weight based clustering protocol is enhanced by optimizing the number of cluster heads using genetic algorithmic approach. The possible solutions given by original RWC are mapped to genetic algorithm technique in order to find the better solution from a pool of solutions. GA techniques were applied to optimize the cluster head selection process of RWC such that each cluster head handles the maximum possible number of nodes in its cluster. The simulation results show that fewer cluster heads are obtained by applying GA to RWC than the original RWC. The cluster head changes are drastically reduced leading to the increase of cluster head life time thereby enhancing the stability of the network to a great extent. In future, some load balancing technique would be incorporated to balance the load among the cluster heads that improves the performance of the network to a great extent.

REFERENCES

- Chatterjee, M., S.K. Das and D. Turgut, 2002. WCA: A Weighted Clustering Algorithm (WCA) for mobile ad hoc networks. *Cluster Comput.*, 5: 193-204.
- Chambers, L., 1995. *Practical Handbook of Genetic algorithm, Application*, Vol. 1, CRC Press.
- Er, I.I. and W.K.G. Seah, 2004. Mobility-based d-hop clustering algorithm for mobile ad hoc networks. *IEEE Wireless Common. Networking Conf.*, 4: 2359-2364.
- Goldberg, D.L., 1953. *Genetic Algorithm in Search Optimization and Machine Learning*, Addison Wesley.
- Sharmila, A.J.F. and B.R. Elijah, 2009. Relative weight based clustering in mobile ad hoc networks, held in 11th IEEE International Conference on Computers Modeling and Simulation, Cambridge, UK.
- Sharmila, J.F. and B.R. Elijah, 2008. Performance analysis of clustering protocols in mobile ad hoc networks. *J. Comput. Sci.*, 4 (3): 192-204.
- Turgut, D., S.K. Das, R. Elmasri and B. Turgut, 2002. Optimizing clustering algorithm in mobile ad hoc networks using genetic algorithmic approach. *IEEE. Global Telecommunications Conference. GLOBECOM Apos*, 02, 1 (17-21): 62-66.