International Journal of Computer Science and Network (IJCSN)
Volume 1, Issue 4, August 2012 www.ijcsn.org ISSN 2277-5420

A Differential Evaluation Algorithm for routing Optimization in Mobile Ad-hoc Networks

[1]Anju Sharma, [2]Madhavi Sinha

[1] Computer Science, Birla Institute of Technology , Mesra, Ranchi
Jaipur Campus Jaipur, Rajasthan, 302017, India

[2] Computer Science, Birla Institute of Technology , Mesra, Ranchi
Jaipur Campus Jaipur, Rajasthan, 302017, India

Abstract

Mobile ad-hoc networks have a dynamic topology due to node mobility, limited channel Bandwidth, and limited battery power of nodes. In order to efficiently transmit data to its destination, the appropriate routing algorithms must be implemented in mobile ad-hoc networks. In this paper we propose a routing optimization algorithm to efficiently determine an optimal path from a source to a destination in mobile ad-hoc networks . The proposed algorithm is designed using a Differential Evaluation(DE) that is a population based stochastic function optimizer using vector differences for perturbing the population. The proposed method is compared with Genetic algorithm(GA), Particle Swarm Optimization(PSO) and Simulation Annealing(SA).

Keywords: *Mobile ad-hoc networks, Differential Evaluation, Genetic algorithm, Particle Swarm Optimization and Simulation Annealing.*

1. Introduction

A wireless ad-hoc network is a network which does not use any infrastructure such as access points or base station. In a typical ad hoc network , mobile nodes come together for a period of time to exchange information, while exchanging information , the nodes may continue to move, and so the network must be prepared to adapt continually. In this dynamic network each node is considered as a mobile router but in an energy-conserving manner. The idea of ad hoc networking is sometimes also called infrastructure-less networking, consists of autonomous nodes that collaborate in order to transport information. Usually these nodes act as end systems and routers at the same time.

Routing protocol is the set of rules defining the router machine(h/w and s/w) find the way that packets containing information have to follow to reach intended destination. The routing protocol must perform efficiently in environment in which nodes are stationary and bandwidth is not a limiting factor. Yet, the same protocol must still function efficiently when the bandwidth available between nodes is low and the level of mobility and topology change is high. In terms of the routing problem in mobile ad hoc networks, if the optimal path has not been determined for transmitting data from a source to a destination, then serious problem such as high transmission delay and high energy consumption by these nodes will occur. Thus it is certainly necessary for a routing optimization algorithm to solve this problem.

Another important requirement for mobile ad-hoc network routing protocol is a time-constraint service to determine a path from a source to a destination since the topologies of mobile ad-hoc networks are more frequently changed than those of other types of networks. In order to solve this problem, most recent studies on such problems seem to focus on evolutionary computation. Differential Evaluation is very appealing due to the great convergence characteristics that it presents when compared to other algorithms from evolutionary computation. DE obtains solutions to optimization problems using three basic operations: Mutation, crossover and selection. The mutation operator generates noisy replicas (mutant vector) of the current population inserting new parameters in the optimization process. The crossover operator generates the trial vector by combining the parameters of the mutant vector with the parameters of a parent vector selected from the population. In the selection operator the trial vector competes against the parent vector and the one with better performance advances to the next generation. This process is repeated over several generations resulting in an evolution of the population to an optimal value.

In this paper, Differential Evolution is discussed to solve the ad-hoc routing optimization problem by

considering the linear equality and inequality constraints. And the results were compared with GA, PSO as SA. The algorithm described in this paper is capable of obtaining optimal solutions efficiently.

2. Related Work

Ad hoc routing protocols can be divided into two categories: topology based and position based [1]. ***Topology based*** routing protocols use the information about the links that exists in the network to perform packet forwarding. Position-based routing protocols use the geographical position of nodes to make routing decisions, which results in improving efficiency and performance. In recent developments, ***position-based*** routing protocols exhibit better scalability, performance and robustness against frequent topological changes. Topology-based routing can be further divided into two approaches: Proactive and reactive approach. Proactive routing protocols periodically broadcast control messages in an attempt to have each node always know a current route to all destinations. Proactive approach maintains routing information about the available paths in the network even if these paths are not currently used. But the drawback of this approach is that the maintenance of unused paths. Reactive routing protocols maintain only the routes that are currently in use thereby reducing the burden on the network, are more appropriate for wireless environments because they initiate a route discovery process only when data packets need to be routed. There is no periodic routing packets required. The destination sequenced distance vector and the wireless routing protocol are popular examples of table driven protocols. Dynamic source routing ,on demand distance vector routing and associativity-based routing are representative on demand (reactive) protocols.

Some routing protocols for delay tolerant networks have also been proposed to overcome frequent, long duration connectivity disruptions. They are classified into three types: deterministic, enforced and opportunistic approach. The deterministic approach can be designed when the information of network is known in advance. The enforced approach provides special mobile nodes to make a connection between disconnected parts of network. The opportunistic approach can be used to delay tolerant network routing. They presented the opportunistic routing design space by drawing the correspondence between the proposed delay tolerant network taxonomy and the basic opportunistic routing building blocks.

2.1 Problem Formulation

In 1995, Price and storn proposed a new floating point encoded evolutionary algorithm for global optimization and named it DE owing to a special kind of differential operator, which they invoked to create new offspring from parent chromosomes instead of classical crossover or mutation.

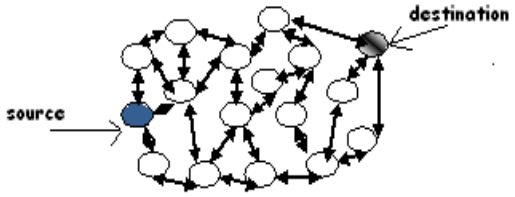

Fig. 1 Network model

In the network model of Fig. 1, we make some assumptions to apply the proposed DE algorithm. We assume that every node is bi-directionally communicate with neighboring nodes via the link between the nodes. Every node has the same data processing capabilities and communication range. The goal is to search an optimal solution for the routing optimization problem.

Problem 1

If the solution vector(donor vector(link)) in the network model used to perturb each network member, and is created using any two randomly selected member of the network as well as the best vector of the current generation, then this can be expressed for the *i*th solution vector at time t=t+1 as

$$\vec{V}_i(t+1)=\vec{X}_i(t)+\lambda.(\vec{X}_{bes}(t)-\vec{X}_i(t))+F.(\vec{X}_{r2}(t)-\vec{X}_{r3}(t))$$

Where λ is another control parameter of DE in [0,2], $\vec{X}_i(t)$ is the target vector and $\vec{X}_{best}(t)$ is the best member of the network regarding fitness at current time.

Problem 2

If the vectors to be perturbed is selected randomly and two weighted difference vectors are added to the same to produce the donor vector. Thus for each target vector, a totality of five other distinct vectors are selected from the rest of the network. The process can be expressed in the form of an equation as

$$\vec{V}_i(t+1)=\vec{X}_i(t)+F_1.(\vec{X}_{r2}(t)-\vec{X}_{r3}(t))+F_2.(\vec{X}4(t)-\vec{X}_{r5}(t))$$

Here F_1 and F_2 are two weighing factors selected in the range from 0 to 1. To reduce the number of parameters we may choose $F_1=F_2=F$

3. Optimization Using Differential Evaluation

Differential Evaluation is one of the most recent population based stochastic evolutionary optimization techniques. DE is a heuristic method for minimizing non-linear and non-differentiable continuous space functions. Differential evaluation includes Evolution strategies(ES) and conventional Genetic Algorithms(GA). Differential evaluation is a population based search algorithm, which is an improved version of Genetic Algorithm. One extremely powerful algorithm from Evolutionary computation due to convergence characteristics and few control parameters is differential evolution. Like other evolutionary algorithms, the first generation is initialized randomly and further generations evolve through the application of certain evolutionary operator until a stopping criteria is reached . The optimization process in DE is carried with four basic operations namely. Initialization, Mutation, Crossover and Selection

3.1 Initialization

DE starts with the population of NP D-dimensional search variable vectors. We will present subsequent generations in DE by discrete time steps like t =0,1,2,…..t, t+1, etc. Since the vectors are likely to be changed over different generations we may adopt the following notations for representing the *i*th vector of the population at the current generation (i.e., at time t = t) as

$$\overrightarrow{X_i}(t) = [x_{i,1}(t), x_{i,2}(t), x_{i,3}(t).....x_{i,D}(t)]$$

These vectors are referred in literature as “genomes” or “chromosomes”. DE is a very simple evolutionary algorithm. For each search-variable, there may be a certain range within value of the parameter should lie for better search results. At the very beginning of DE run or at t = 0, problem parameters or independent variables are initialized somewhere in their feasible numerical range. If the jth parameter of the given problem has its lower and upper bound as x_j^L and x_j^U , respectively, then we may initialize the jth component of the ith population members as

$$x_{i,j}(0) = x_j^L + rand(0,1)\cdot(x_j^U - x_j^L)$$

where rand(0,1) is a uniformly distributed random number lying between 0 and 1

3.2 Mutation operation

The mutation operation is applied to the set of genes of all the chromosomes with the mutation probability q. The mutation operation changes or flips a gene of the candidate chromosomes to keep away from the local optima. In this operation it randomly select a population of chromosomes and then select a gene of this chromosome. We should check that chromosome is feasible , if not, then change its state into feasible by using the repair function. In this scheme, to create $\vec{V}_i(t)$ for each *i*th member, three other parameters say r_1,$r_2$and r_3 are chosen in a random fashion from the current population and F is a scalar number that scales the difference of any two of the three vectors and the scaled difference is added to the third one that we obtained the donor vector $\vec{V}_i(t)$. We can express the process for the *j*th component of each vector as

$$v_{i,j}(t+1) = x_{r1,j}(t) + F.(x_{r2,j}(t) - x_{r3,j}(t)).....$$

Next to increase the potential diversity of the population a crossover scheme comes to play.

3.3. Crossover operation

The crossover operation between two chromosomes is conducted among each corresponding set of genes with the crossover probability p. first two chromosomes are selected as the crossover partner, next, the crossover operation changes the corresponding genes of the two chromosomes. In the crossover operation, all the corresponding lower genes are exchanged when a gene of a chromosome is exchanged with the corresponding gene of another chromosome. It adds varieties to the swarm. It includes two modes, index crossover mode and binomial crossover mode. The algorithm uses the binomial crossover mode which can be defined as:

$$u_{i,j}(t) = v_{i,j}(t) \quad if\ rand(0,1) < C_r,$$
$$= xi, j(t) \quad else.....$$

Where C_r is a crossover factor and rand is a random decimal figure between [0,1]. To keep the population size constant over subsequent generations, the next step of the algorithm calls for “selection” to determine which one of the target vector and the trial vector will survive in the next generations at time t+1.

3.4 Selection operation

DE actually involves the Darwinian principle of "survival of fittest" in this selection process which may be outlined as

$$\vec{X}(t+1) = \vec{U}_i(t) \quad \text{if } f(\vec{U}_i(t)) \leq f(\vec{X}_i(t)),$$
$$= \vec{X}_i(t) \text{ if } f(\vec{X}_i(t)) < f(\vec{U}_i(t)),.....$$

Where f ( ) is the function to be minimized. So if the new trial vector yields a better value of the fittest function, it replaces its target in the next generations. Hence the population either gets better or remains constant.

4. Other Optimization Techniques

In order to evaluate the proposed Differential Evaluation algorithm , we compare it with other optimization techniques, which are the Genetic Algorithm(GA), Particle Swarm Optimization(PSO) and Simulation Annealing(SA).

4.1 Genetic Algorithm(GA*)*

The genetic Algorithm, which was introduced by Holland[2] and was further described by Goldberg[3] is a stochastic optimization technique. The genetic algorithm [5] is a search heuristic that mimics the process of natural evolution. GA belongs to the larger class of evolutionary algorithms(EA). The GA procedure is based on the principle of survival of fittest. The algorithm identifies the individual with the optimizing fitness values, and those with lower fitness will naturally get discarded from the population. But there is no absolute assurance that a genetic algorithm will find a global optimum. Due to Dynamism and unpredictable nature, a MANET is a challenging environment for software designers.

In a directed graph G=(V,E) each element x_i can be defined as

$$X_i = \begin{cases} 1, \textit{if edge } e_i \textit{ is selected in the subgraph} \\ 0, \textit{otherwise} \end{cases}$$

Where parameters are as follows:

$V=\{v_1,v_2,v_3,\ldots v_n\}$- vertex set of G,

$E=\{e_1,e_2,e_3\ldots e_n\}$ - finite set of edges of G.

Let $W=\{w_1,w_2,w_3\ldots\ldots w_n\}$ represent the weight or cost of the edge. Then minimum value of the graph can be formulated as

$$\min f(x) = \sum_{i=1}^{m} \{w_i x_i \mid x \varepsilon T\}$$

In genetic algorithm[6], the crossover operation between two chromosomes is conducted among each corresponding set of genes with the crossover probability p . For each parameter a random value based on binomial distribution is generated in the range[0,1].

The mutation operation is applied to the set of genes of all the chromosomes with the mutation probability q. the mutation operation changes or flips a gene of the candidate chromosomes to keep away from the local optima.

4.2 Particle Swarm Optimization(PSO)

Kennedy and Elberhart introduced the concept of function-optimization by means of a particle swarm[7]. Particle swarm optimization(PSO) is a population based on stochastic optimization technique, which simulates the social behavior of organisms, such as bird flocking and fish schooling to describe an automatically evolving system. PSO is a multi-agent parallel search technique. Particles are conceptual entities, which fly through the multi-dimensional search space as in Mobile ad-hoc network. At any particular instant, each particle has a position and velocity. At the beginning a population of particles is initialized with random positions and velocities can be denoted by the parameters $\overrightarrow{X_i}$ and $\overrightarrow{V_i}$ respectively. Each particle stores the value and location of the best solution found called the local best (Lbest) also all particles are aware of the value and location of the best solution found by all other particles, called global best (Gbest). At each iteration the particles compare the Lbest and Gbest to choose a direction independently based on the distance differences from current location to the Gbest and to the Lbest location. The distance between two locations can be evaluated as

$$D = \sqrt{d_1^2 - d_1^1)^2 + (d_2^2 - d_2^1)^2}$$

The distance will be evaluated to find the values of L_{best} and G_{best}. Then these two parameters must be compared, if $L_{best} > G_{best}$ is true then G_{best} and L_{best} are replaced. It calculates L_{best} . so the particle can move to new position.

In iterative optimization process, the positions and velocities of all the particles are altered by the

following recursive equations. This equation defines the position and velocity of the *i*th particle[9].

$$V_{i\max}(t+1)=\omega V_i(t)+C_1.\varphi_1.(p_i(t)-X_i(t))+C_2.\varphi_2.(G_{best}(t)-X_i(t))$$

$$X_i(t+1)=X_i(t)+V_i(t)$$

Where parameters are as follows:
V_{max} = maximum velocity
P_i = *i*th particle
ω = the inertial weight factor
φ_1 *and* φ_2 = two uniformly distributed random numbers in the interval [0,1]
C_1=constant multiplier termed as "Self confidence"
C_2= constant multiplier termed as "Swarm confidence"

This process is iterated for a certain number of time steps, or until some acceptable has been found by the algorithm.

4.3 Simulation Annealing(SA)

Simulation Annealing(SA) is a global optimization method that distinguishes between local optima. After an initial point of the algorithm, it takes a step and the function is evaluated. It is based on two results of statistical physics . First if a physical system has a given energy when the thermodynamic balance is reached at a given temperature, then the probability of the system is proportional to the Boltzmann factor. Second the metropolis algorithm can be utilized to simulated the evolution of a physical system at a given temperature. It is quite robust with respect to non-quadratic surfaces. In fact, Simulation annealing can be used as a local optimizer for difficult functions[10] . This algorithm decreases a given temperature by multiplying the cooling parameter δ of the initial temperature t_{in} by the final temperature t_{fin}.

In each iteration, new solutions, X are produced by one of the two neighborhood generating operations that adapt to the current solution, X_a . The probability of selecting the neighborhood generating operations depends on the given operation threshold, ℓ . Distance between two neighbors can be evaluated by :

$$D=\cos t(X)-\cos t(X_b)$$

If the value of D[16] between the cost of X and the cost of X_b is less than zero, then X is accepted as X_b, otherwise, a random number is distributed in the interval(0,1) is selected, then this number is compared with the Boltzmann factor exp(D/t), then X is accepted as X_b, otherwise X_b is accepted.

5. Performance Criteria

In this section, we compare the proposed Differential evaluation algorithm with three Genetic Algorithm, Particle swarm optimization and Simulation Annealing vie computer experiments[16].

Each mobile node in the network start its journey from a random location to a random destination with a randomly chosen speed. Once the destination is reached, another random destination is targeted after a pause by the mobile node. Once the node reaches the boundary area mentioned in the network, it chooses a period of time to remain stationary.

We measure the routing cost of the Differential Evaluation with the number of iterations: 10,50,100 and 200. In general, if the number of iterations increases in the Differential Evaluation , the probability of finding the optimal solution increases. The minimum routing cost in these algorithms termed as J_{max}.

Table 1: The parameters used in the different algorithm

Algorithms	Parameters	values
Differential Evaluation	C_r rand J_{max}	**>0** **1/0.5** **10/50/100/200**
Genetic Algorithm	p q J_{max}	**1/0.5** **1/0.5/0.25** **100**
Particle Swarm Optimization	φ J_{max}	**1/0.5** **50/100**
Simulation Annealing	t_{in} t_{fin} δ ℓ J_{max}	**0.1** **0.0005** **0.1** **0.1/0.3/0.5/0.7/0.9** **100**

On the other hand by increasing the number of nodes in the network, we see that the differential evaluation with the large number of iterations finds an optimal solution with better performance. The result of average execution for all the cases also increase in proportion to the number of nodes. If we take a fixed value of iteration as 100, then the value of two parameters minimum routing cost and average execution time can be evaluated as shown in the table 2.

Table 2: Performance criteria for different algorithms

Algo. / Parameter	GA	PSO	SA	DE
Minimum routing cost	700	600	600	500
Average execution time	0.563	0.461	0.271	0.240

The result shows that the Differential Evaluation takes a shorter time than the Genetic Algorithm, Particle Swarm Optimization and Simulation Annealing.

Finally, for the routing problem in the mobile ad-hoc networks, we observe that the proposed Differential Evaluation algorithm can efficiently solve this problem in terms of routing cost and it is pertinent to solve the problem within a reasonable execution time.

Also, we can consider the networking *context* in which a protocol's performance is measured. Essential parameters that should be varied include:
A. Network size --measured in the number of nodes Network connectivity--the average degree of a node (i.e. the average number of neighbors of a node)
B. Topological rate of change--the speed with which a network's topology is changing.
C. Link capacity--effective link speed measured in bits/second, after accounting for losses due to multiple access, coding, framing, etc.
D. Fraction of unidirectional links--how effectively does a protocol perform as a function of the presence of unidirectional links?
5 *Traffic patterns*--how effective is a protocol in adapting to non-uniform or bursty traffic patterns?
6 *Mobility*--when, and under what circumstances, is temporal and spatial topological correlation relevant to the performance of a routing protocol? In these cases, what is the most appropriate model for simulating node mobility in a MANET?
7 *Fraction and frequency of sleeping nodes*--how does a protocol perform in the presence of sleeping and awakening nodes?

A MANET protocol should function effectively over a wide range of networking contexts--from small, collaborative, ad hoc groups to larger mobile, multihop networks.

In summary, the networking opportunities for MANETs are intriguing and the engineering tradeoffs are very challenging.

6. Conclusion

In this paper we proposed a Differential Evolution algorithm for Mobile ad-hoc network. The performance evaluation of different algorithms show the better performance of the DE for the parameters, minimum routing cost and average execution time in comparison to other algorithms GA, PSO and SA. Finally we suggest that in future performance evaluation of DE for MANET's need to be more comprehensive. Evaluation should consider a range of realistic mobility models and

REFERENCES

[1] Royer, E.M.& Toh, C.K. (1999). *A review of current routing protocols for ad-hoc mobile wireless networks*. IEEE personal communications,6,46-55 .
[2] Holland,J.(1975). *Adaptation in natural and artificial systems*. Ann arbor: univ. of Michigan Press.
[3] Goldberg,D.E. (1989). *Genetic algorithms in search, optimization & machine learning*. Reading: Addison-wesley
[4] Ahn,C.W.,Ramakrishna,R.S.,Kang, C.G.,&Choi,I.C.(2001). *Shortest path routing algorithm using Hopfield neural network*. Electronics Letter,37(19),1176-1178.
[5] K.Deb,(2000). "*An efficient constraint handling method for genetic algorithms*", Computer Methods in Applied Mechanics and Engineering, Elsevier, Netherlands, 186(2-4):311-338.
[6] E.Baburaj, V.Vasudevan "An Enhanced tree based MAODV protocol for MANET'S using Genetic Algorithm"
[7] Kennedy J and Elberhart R and shi Y(2001), swarm intelligence, Morgan Kaufmann,Los Altos,CA.
[8] Kennedy J and Elberhart R (1995), Particle swarm optimization,In proceedings of IEEE International conference on Neural networks, pp. 1942-1948.
[9] S .Das et al (2008): Particle Swarm Optimization and Differential Evolution Algorithms: Technical Analysis, Applications and Hybridization Perspectives, studies in Computational Intelligence(SCI) 116, 1-38
[10] K.P. Wong and C.C (1993). Fung, "Simulation Annealing based Economic Dispatch Algorithm",proc.Inst. Elect.Eng. Ge. Trans. Vol. 140, no. 6, pp. 509-515.
[11] Idris Skloul Ibrahim, Peter J.B King, Robert Pooley(2009) "Performance Evaluation of Routing Protocols for MANET" Fourth International Conference on Systems and Networks Communications , IEEE 2009 pp. 105-112.
[12] Narendra Singh, R.P. Yadav, "Performance Comparison and Analysis of Table-Driven and On-Demand Routing Protocols for Mobile Ad-hoc Networks" International Journal of Information Technology Volume 4 Number 2 pp. 101-109.
[13] C.K. perkins and E. M. Royer, "Ad Hoc On- Demand Distance Vector Routing," Proceedings of IEEE Workshop on Mobile Computing Systems and Application 1999, February 1999, pp.90- 100.
[14] S. Murthy and J.J. Gracia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks" ACM Mobile Networks and

Applications Journal, Special issue on routing in Mobile communication Networks, Vol. 1, no. 2, October 1996, pp. 183-197.

[15] C.K. perkins and P. Bhagat, "Highly Dynamic Destination-Sequenced Distance-Vector Routing(DSDV) for Mobile Computers" In Proceedings of ACMSIGCOMM 1994, August 1994, pp. 234-24

[16] K.W. Jang "A tabu search algorithm for routing optimization in mobile ad-hoc networks" springer(2011

should include special cases such as high density, high mobility of nodes.