
A differential evolution-based routing algorithm for multi-path environment in mobile ad hoc network

Anju Sharma* and Madhavi Sinha

Birla Institute of Technology,
27, Malviya Industrial Area Jaipur Campus,
Jaipur, India
Email: anjusharma@bitmesra.ac.in
Email: madhavisinha@bitmesra.ac.in
*Corresponding author

Abstract: Optimisation is a procedure through which the best possible ways of decision variables are obtained under the given set of constraints and in accordance to a selected optimisation objective function. Over the last decade, differential evolution (DE) algorithms have been extensively used in various problem domains and succeeded effectively in finding the optimal solutions. The present paper emphasis on a DE algorithm-based ad hoc on-demand multi-path distance vector (DE_AOMDV) protocol for mobile ad hoc network (MANET). The proposed DE_AOMDV routing protocol has better performance and improves the quality of the MANETs. The main objective of this research paper is to find the optimal path from available multiple paths between source and destination to be used in route recovery process. DE_AOMDV provides backup paths to avoid reroute discovery in the case of link failure between nodes. Simulation based results and data analysis shows DE_AOMDV protocol is better than AOMDV.

Keywords: differential evolution; DE; mobile ad hoc networks; MANETs; routing strategy; AOMDV; strategy adaptation.

Reference to this paper should be made as follows: Sharma, A. and Sinha, M. (2019) 'A differential evolution-based routing algorithm for multi-path environment in mobile ad hoc network', *Int. J. Hybrid Intelligence*, Vol. 1, No. 1, pp.23–40.

Biographical notes: Anju Sharma is currently working as an Assistant Professor in the Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Jaipur Campus India since 2005. She received her BSc in Science in 1997 at the B.R. Ambedkar University, India. She received her MSc in Mathematics in 1999 at the B.R. Ambedkar University, India. She received her MTech in Computer Science in 2003 at the Birla Institute of Technology, Mesra, Ranchi, India. She is pursuing her PhD in Computer Science. She has over 12 years teaching experience for UG and PG courses. Her current research interests are distributed computing, networking and ad hoc network.

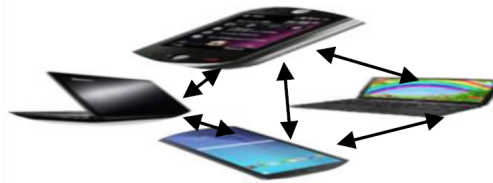
Madhavi Sinha is currently working as and Associate Professor and Head in Department of Computer Science and Engineering, Birla Institute of Technology, Mesra, Jaipur Campus India since 2007. Her current research interests are artificial intelligence, genetic algorithms, real time communication, machine translation, and ad hoc network. She has over 30 papers published in reputed international journals and conference

proceedings. She is member of BOS, research committee of many universities and given invited talks in many national and international conferences. She earned her MSc in Computer Science in 1990 and obtained PhD from the Banasthali Vidyapith University, India in 1998 in Computer Science.

1 Introduction

A wireless mobile ad hoc network (MANET) 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. Usually these nodes act as end systems and routers at the same time as in Figure 1. Each node is considered in this dynamic network as a mobile router, to transfer the message (Riad et al., 2013).

Figure 1 Mobile ad hoc network (see online version for colours)



Ad hoc routing protocols can be divided into two main categories: topology-based and position-based (Jayakumar and Ganapathy, 2007). *Topology-based* routing protocols use the information about the links to perform packet forwarding that exists in the network. *Position-based* routing protocols are used in the geographical position of nodes to take routing decisions that may result in improving efficiency and performance. Topology-based routing can be further divided into two approaches: *proactive* and *reactive* approach (Singh and Yadav, 2007). *Proactive* routing protocols in ad hoc network periodically broadcast control messages in an attempt to have each node always know a current route to all destinations. But it maintains routing information about the available paths even if these paths are not currently used. Some *proactive* routing protocols are destination sequenced distance vector (DSDV) (Perkins and Bhagat, 1994), source-tree adaptive routing (STAR) and wireless routing protocol (WRP). But reactive routing protocols maintain only the routes that are currently in use thereby reducing the burden on the network are appropriate for wireless environments because they initiate a route discovery process when data packets need to be routed. They include the dynamic source routing (DSR) protocol, ad hoc on-demand distance vector (AODV) protocol (Perkins and Royer, 1999), ad hoc on-demand multi-path distance vector (AOMDV) protocol (Al-Nahari and Mohamad, 2016; Marina and Das, 2001) and the temporally ordered routing algorithm (TORA) protocol (Royer and Toh, 1999).

Due to node mobility and unpredictable behaviour of nodes in MANET between source and destination, data delivery is very challenging. Data delivery may be done through single or multiple paths (Paulose and Paulose, 2016; Vanaja and Umarani, 2011).

In this paper we focus on ad hoc on-demand multipath distance vector (AOMDV) routing protocol (Ibrahim et al., 2009). AOMDV provides multiple loop free disjoint paths. Each route discovery is associated with high latency and overhead. This inefficiency can be avoided by having multiple redundant paths available. AOMDV can be used to find node-disjoint or link-disjoint paths. Node-disjoint paths are also known as totally disjoint routes (Huang and Fang, 2009). These paths have no nodes or links in common while link-disjoint paths do not share any link. Node-disjoint paths are desirable where node resources are scarce or when nodes are susceptible to failure. On the other hand, link-disjoint paths are preferred where link resources are scarce or when only links are susceptible to failure. Using link-disjoint paths, one can address the issue of fault-tolerance. However, link-disjoint paths cannot be used for simultaneous data transfer because doing so will overload the nodes that are common in more than one path.

Another important requirement for MANET routing protocol is a time-constraint service to determine a path from a source to a destination since the topologies of MANETs are more frequently changed than those of other types of networks (Elhoseny et al., 2017a). In order to solve this problem, some algorithms (Kennedy and Elberhart, 1995; Kennedy et al., 2001) are implemented to find the optimum and efficient routing path. In this paper, we adapt and investigate differential evolution (DE) (Das et al., 2008) to solve the ad hoc routing optimisation problem by considering the linear equality solutions to optimisation problems using three basic operations: which are mutation, crossover and selection.

Rest of this paper is organised as follows. Section 2 reviews related work on mobile ad hoc network. Section 3 describes about problem description and major issues. Section 4 describes the proposed DE for optimum routing in AOMDV. Section 5 is devoted to performance criteria and finally Section 6 contains the conclusions. The proposed design described in this paper is capable of obtaining optimal solutions efficiently.

2 Related work

Currently, there are many network protocols for MANET with their own advantages and disadvantages. Das et al. (2016) evaluated performance of node-disjoint multipath method in AOMDV with link-disjoint multipath. These protocols investigated with varying node density and pause time. They proposed node-disjoint multipath in AOMDV performs better than link-disjoint method. For the performance analysis they have computed average end to-end delay, routing overhead and throughput. The results are computed from several simulation runs.

Wang (2003) proposed a braided multipath routing, magnetic-field-based multipath routing in mobile ad hoc multipath no fixed network. Braided multipath routing is not flexible to geographically localised and correlated failures. Its shortcoming is that in terms of mobility, each route in MFR has set of nodes; therefore, the node membership for each route can change dynamically without breaking the multiple routes. Mueller et al. proposed (2012) a multipath routing technique which uses braided multi-paths. Braided multipaths relax the requirement for node disjointness. Multiple paths in a braid are only partially disjoint from each other and are not completely node-disjoint. These partially disjoint paths are usually shorter than node-disjoint multi paths and thus

consume less energy resources; alternate paths should consume an amount of energy comparable to the primary path.

Li et al. (2009) proposed the performance of the proposed algorithm, a multipath routing algorithm-based on traffic prediction (MRATP) with Ant-based routing and AODV based on bandwidth optimisation (BW-AODV) for average end-to-end delay, which reflects the ability of convergence. They evaluated that the average end to-end delay of MRATP is slightly less than that of BW-AODV. However, when the same amount of data is sent to the destination end, the total time taken by MRATP is much less than that of BW-AODV because MRATP adopts the method of multi-path streaming mechanism based on traffic prediction.

Daru and Bingbing (2008) proposed not only combined the adaptation, robustness, distributed and decentralised merit of ant colony optimisation (ACO) but also used simulation annealing (SA) to remedy the weakness of Ant algorithm and increase the convergence rate of ACO (Holland, 1975).

Yang et al. (2010) proposed that immigrants and memory-based GAs can quickly adapt to environmental changes (i.e., the network topology changes) and produce high-quality solutions after each change (Ali and Torn, 2004; Yuan et al., 2017). This paper proposed an algorithm to use GAs with immigrants and memory schemes to solve the dynamic SP routing problem in MANETs.

To solve the problems above, a multipath routing algorithm based on traffic prediction (MRATP) is proposed by Li et al. (2009) for wireless mesh networks. MRATP consists of three modules including an algorithm on multipath routing built, a congestion discovery mechanism based on wavelet-neural network and a load balancing algorithm via multipath. Simulation results show that MRATP has some characteristics, such as better scalability, flexibility and robustness (Lee and Gerla, 2000).

Suresh Kumar et al. (2011) proposed a Secure and Adaptive backup routing protocol for MANET using genetic algorithm (Goldberg, 1989; Deb, 2000; Elhoseny et al., 2017b). It provides alternative path or optimal backup path to avoid reroute discovery in the case of link failure or node failure.

3 Problem description and major issues

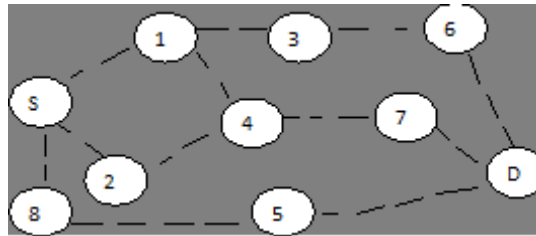
The efficiency of multipath routing depends on the construction and physical distribution of paths. Path selection decides the performance of the multipath routing. If the selected paths are independent of each other or cooperative, multipath routing could improve the performance. Thus, node-disjoint path routing protocols are proposed to avoid interference among paths

Multipath routing protocols can provide fault tolerance by having redundant information routed to the destination via alternative paths. Using node-disjoint paths (Abbas et al., 2004), one can address fault tolerance as well as load sharing. Finding node disjoint paths in case of ad hoc networks is a challenging task. This is due to the fact that the protocol in ad hoc environment should be localised and distributed. If the topology of the network is known, there exist standard methods to discover node-disjoint paths between a given pair of nodes (Sidhu et al., 1991; Suurballe, 1974). A protocol may or may not identify all node-disjoint paths between a given pair of nodes.

First problem in AOMDV is path diminution (Abbas and Jain, 2004), which means that the number of node-disjoint paths discovered by a protocol is less than the number of

node-disjoint paths that exist in the network between a given pair of nodes. We can use a heuristic to possibly compute a maximal set of node-disjoint paths at the destination. Consider a situation as shown in Figure 2 there exist multiple node-disjoint paths from a source s to a destination d . Initially, the communicating pair is not aware of these paths. Suppose node s initiated a route discovery by sending RREQ to its neighbours 1, 2, 8. Node 1 forwards the RREQ to its neighbours before nodes 2 and 8. At node 4, the RREQ transmissions of node 2 are discarded due to the fact that these are considered as the duplicate copies of the same RREQ which node 4 has already processed. Node 4 forwards the RREQ to its neighbours before nodes 5 and 6. The duplicate RREQs at other nodes are simply discarded. Therefore, the path formed is $(s, 1, 3, 6, d)$. It should be noted that this path has blocked the disjoint paths $(s, 2, 4, 7, d)$ and $(s, 8, 5, d)$ which might have found in the network.

Figure 2 Disjoint paths in MANETs



Another problem in AOMDV (Baburaj and Vasudevan, 2008) is path selection. Also, because of node mobility and node failure, the topology changes dynamically. Therefore, an algorithm that requires the topology to be known (and fixed) cannot be used in a MANET. A badly selected path at an early stage can block other disjoint paths. This means that one might not be able to find disjoint paths, even if they exist in the network. In this paper, we proposed DE-based AOMDV (DE_AOMDV) which is very appealing due to the great convergence characteristics that it presents when compared to other algorithms from evolutionary computation. In DE_AOMDV, the mutation operator generates mutant vector (noisy replicas) of the current population inserting new parameters in the optimisation process. The crossover operator used to generate the trial vector by combining the mutant vector with the parameters of a parent vector selected from the population. The trial vector competes against the parent vector in the Selection operator and the one with better performance advances to the next generation. Several generations must be repeated until resulting in an evolution of the population to an optimal value (Lee and Jeon, 2015). Intelligent path selection can be used to enhance the performance of multipath routing.

4 Proposed protocol (DE_AOMDV)

To improve the routes stability and to provide more optimal paths from alternative paths in AOMDV, this paper proposed an optimised routing path from source to destination in MANET. To find the optimal paths from the available multiple paths, we use DE (Gundry and Kusyk, 2012). A Multipath routing protocol based on DE algorithm is

proposed in MANETs to reduce the time in case of route failure. DE design-based protocol increases the performance (packet delivery ratio, throughput).

In this section, we first present our network model and then formulate the DE_AOMDV. We consider a MANET operating within a fixed geographical region. We model MANET as an undirected and connected topology graph $G(V, E)$ where V represents the set of wireless nodes (i.e., routers) and E represents the set of communication links connecting two neighbouring routers falling into the transmission range. A communication link (i, j) cannot be used for packet transmission unless both node i and node j have a radio interface each with a common channel. In addition, message transmission on a wireless communication link will incur remarkable delay and cost.

Here, we summarise some notations that we use throughout this paper:

$G(V, E)$	initial MANET topology graph
$G_i(V_i, E_i)$	MANET topology graph after the i^{th} change
s	source node
d	destination node
$P_i(s, d)$	path from s to d on the graph G_i
dl	transmission delay on the communication link l
cl	cost (distance) on the communication link l
$\Delta(P_i)$	total transmission delay on the path P_i
$C(P_i)$	the total cost of the path P_i .

Total transmission delay on path P_i is calculated by:

$$\Delta(P_i) = \sum_{l \in P_i(s, d)} dl \quad (1)$$

Total cost of the path P_i is calculated by:

$$C(P_i) = \sum_{l \in P_i(s, d)} cl \quad (2)$$

4.1 Route discovery

In AOMDV, source node needs a route discovery process by generating route request packets RREQs. Since the RREQs are flooded network-wide, a node may receive several copies of the same. All duplicate copies are examined in AOMDV for potential alternate reverse path, but RREQ reverse paths are formed only using those copies that preserve loop-freedom and disjointness among the resulting set of paths to the source. When an intermediate node receives a reverse path via a RREQ copy, it checks whether there are one or more valid forward paths to the destination. If so, node generates a route reply packet RREP and sends it back to the source along the reverse path; the RREP includes a

forward path that was not used in any previous RREPs for this route discovery. The intermediate nodes do not propagate the RREQ further. Otherwise, the node re-broadcasts the RREQ copy if it has not previously forwarded any other copy of this RREQ and this copy resulted in the formation/update of a reverse path.

When destination d receives RREQ copies, reverse paths are formed as in the same way as intermediate nodes. The destination generates a RREP in response to every RREQ copy that arrives via a loop-free path to the source even though it forms reverse paths using only RREQ copies that arrive via loop-free and disjoint alternate paths to the source. In RREQ flooding mechanism, a node receives a route advertisement using equation (3).

$$\begin{aligned} advertised_hop_count_s^d &:= \max\{hop_count_s^d\}, s \neq d \\ &:= 0, otherwise \end{aligned} \quad (3)$$

4.1.1 Path diminution

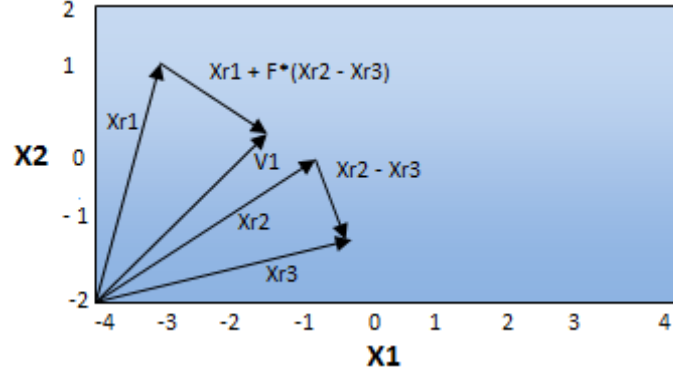
The number of disjoint paths between a communicating pair depends on the degrees of source and destination. The maximum number of disjoint paths is: $K_{\max} = \min(d_s, d_d)$, where d_s is the out-degree of source and d_d is the in-degree of destination.

The forwarding rule in AOMDV is that at any intermediate node the RREQ is forwarded only once. The duplicate copies coming later are simply discarded. We can call this policy early discard scheme. The RREQs are discarded prematurely before setting up source to destination paths or in other words before computing the end-to end disjointness. For this problem, we can introduce a policy for RREQ forwarding as follows. Upon receiving a copy of an RREQ, an intermediate node checks the previous hopID in PathTraversed of the RREQ. An intermediate node forwards the copy of the RREQ if and only if the previous hopID does not match with the previous hopID of any of the copy of the RREQ stored in its RREQCache. In this technique, every node transmits at most one copy per neighbour (OCN). As a result, the number of RREQ transmissions is twice the number of edges in the network $O(n)$. This technique is known as *at most OCN*.

After that, the destination collects all copies of an RREQ before the expiry of a timeout. Upon expiry of the timeout, the destination computes the maximum number of node-disjoint paths between the source and the destination. The overhead acquired in computing a maximal set of node-disjoint paths at the destination is $O(zn \log n + z^2 n + z^2)$ where z is the number of copies of an RREQ received by the destination.

4.2 Implementation of DE

In this section we describe the design of the DE for the SP problem (Storn and Price, 1997). DE is one of the most recent population-based stochastic evolutionary optimisation techniques. DE is a heuristic method for minimising non-linear and non-differentiable continuous space functions. DE (Qin and Suganthan, 2005; Qin et al., 2009) is a powerful algorithm due to convergence characteristics and few control parameters as in Figure 3.

Figure 3 DE in MANET's (see online version for colours)

The design of the DE has components like population initialisation, fitness function, mutation, crossover and selection scheme. A routing path contains the sequence of nodes in network. The DE algorithm is applied to node disjoint paths that is been obtained from the route discovery phase. A routing path is encoded by a string of positive integers that indicating the IDs of the nodes in the network. The length of the string should not be more than the number of nodes present in the network.

4.2.1 Initialisation

DE starts with the population of NP D-dimensional search variable vectors. We present subsequent generations in DE by discrete time steps like $t = 0, 1, 2, \dots, t, t + 1$, 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., $G = 0$) as

$$\overline{X}_{i,G} = [x_{i1,G}, x_{i2,G}, x_{i3,G}, \dots, x_{iD,G}] \quad (4)$$

These vectors are referred in literature as ‘genomes’ or ‘chromosomes’. In DE, each chromosome represents a potential solution and this can contain more than one solution initially. The node disjoint paths obtained from route discovery phase are considered as initial chromosomes.

For an obtained solution we should be able to evaluate its quality accurately which can be done with the help of fitness function. In this problem, we obtain the shortest path and lowest transmission delay time as the primary concern. So fitness function can be defined as

$$f(X) = \sum_{l \in P(s,d)} cl + dl \quad (5)$$

Equation 5 is obtained from equation 1 and equation (2). Where cl represents the cost of the path and dl represents the transmission delay taken by the each chromosome from source to destination.

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

initialised somewhere in their feasible numerical range. If the j^{th} parameter of the given problem has its lower and upper bound as x_j^L and x_j^U respectively, then we can initialise the j^{th} component of the i^{th} population members as

$$x_{i,j}(0) = x_j^L + \text{rand}(0, 1) \cdot (x_j^U - x_j^L) \quad (6)$$

where $\text{rand}(0, 1)$ is a uniformly distributed random number lying between 0 and 1.

4.2.2 Mutation operation

After initialisation, DE employs the mutation operation to produce a mutation vector $v_{i,G}$, with respect to each individual $x_{r,G}$, so called target vector, in the current population. For each target vector $x_{r,G}$, (initially taken at random) at the generation G , its mutation vector $v_{i,G} = \{v_{i1,G}, v_{i2,G}, \dots, v_{iD,G}\}$ can be generated via certain mutation strategy as in Figure 3. For each target vector $x_{r,G}$ a mutant vector $v_{i,G+1}$ is generated according to the following:

$$v_{i,G+1} = x_{r1,G} + F \cdot (x_{r2,G} - x_{r3,G}) \quad (7)$$

The indices x_{r1} , x_{r2} and x_{r3} are mutually exclusive integers randomly generated within the range $[1, \text{NP}]$.

Equation (7) shows that we must have at least four node disjoint paths (population) after route discovery. This problem has been resolved by the path diminution. These indices are randomly generated once for each mutant vector. In mutation, the scaling factor F is a positive control parameter for scaling the difference vector.

For better local search ability and faster convergence rate, a mutant vector $v_{i,G+1}$ can also be generated according to the following:

$$v_{i,G+1} = x_{r1,G} + F \cdot (x_{b,G} - x_{w,G}) \quad (8)$$

where x_{r1} is a random chosen vector and x_b and x_w are the best and worst vectors in the entire population.

4.2.3 Crossover operation

After the mutation phase, crossover operation is applied to each pair of the target vector $x_{i,G}$, and its corresponding mutant vector $v_{i,G}$

$$\begin{aligned} U_{i,G+1} &= v_{i,G+1} & \text{if } \text{rand}(0, 1) < C_r, \\ &= x_{ij,G} & \text{else...} \end{aligned} \quad (9)$$

where C_r is a crossover factor and rand is a random decimal figure between $[0, 1]$. The binomial crossover operator copies the j^{th} parameter of the mutant vector $v_{i,G+1}$ to the corresponding element in the trial vector $U_{i,G+1}$ if $\text{rand}(0, 1) < C_r$ otherwise; it is copied from the corresponding target vector $x_{i,G}$. To keep the population size constant over subsequent generations, the next step of this algorithm calls for 'selection' to determine which one of the target vector and the trial vector will survive in the next generations at $G+1$.

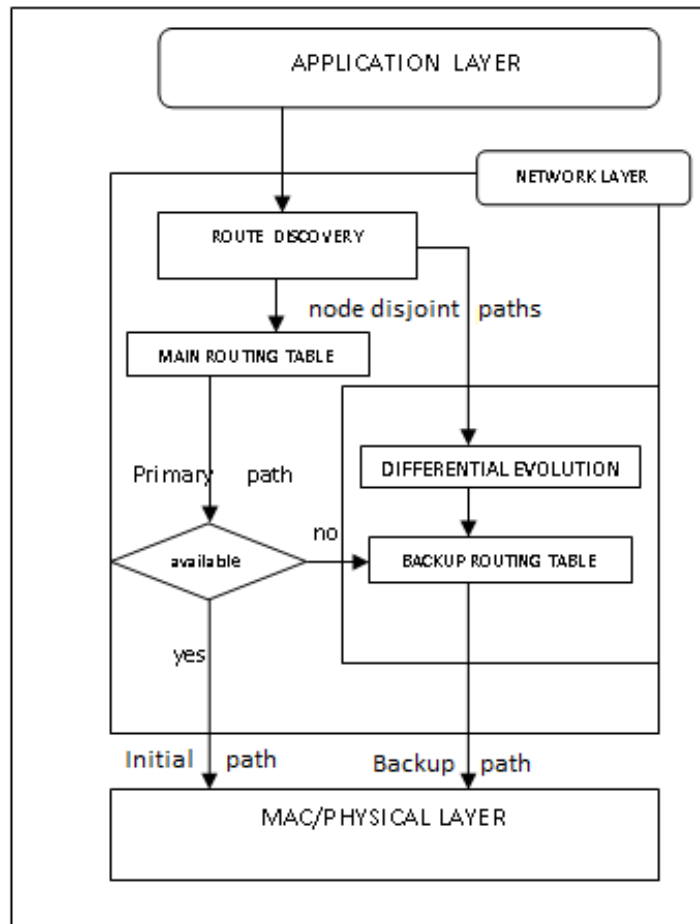
4.2.4 Selection operation

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

$$x_{i,G+1} = \begin{cases} U_{i,G+1} & \text{if } f(U_{i,G+1}) \leq f(x_{i,G}) \\ x_{i,G} & \text{otherwise} \end{cases} \quad (10)$$

where $f(X)$ is the function to be minimised. In DE, the selection operator plays two roles: guiding the individuals to advance towards the search space in which solutions have higher ranks and less crowded. If the new trial vector yields a better value of the fittest function, it replaces its target in the next generations. Hence the population gets better or constant.

Figure 4 Backup routing using DE in AOMDV



4.3 Backup routing using DE

In this section we discover backup paths from source to destination in case of primary path link failure. We proposed a new protocol to improve existing on-demand multiple path routing protocols by constructing multiple backup routes using DE algorithm in the last section. By using DE (Aswani et al., 2015), we obtain the optimal path from source to destination, and in the same phase we find an alternate path to be used in link failure. The alternate path will be next best path when compared to the optimal path. When a primary path fails we can recover the connection by utilising the backup paths using this method. This backup path routing contains two main functions: path discovery phase and path maintenance phase.

When the network topology changed, the proposed protocol (DE_AOMDV) could transmit data packets dynamically through backup routes. Backup routing had fewer control packages, lower routing packet overhead, and a higher receiving ratio than others.

4.3.1 Path discovery phase

The primary paths and backup paths are established during the route discovery phase itself when we find optimal paths by using DE. The backup paths are geographically closer to the primary paths and more optimal than alternate paths. And we have discussed enough about route discovery phase we will go now to path recovery phase.

Figure 5 Route discovery phase

Algorithm 1 Optimal Path Selection (N[])
<pre> // Input: N is the set of n mobile nodes, N=(N₁, N₂, N₃,.....N_n). //Input: N_s : Source Node, N_d: destination node //output: feasible paths from backup routing table </pre>
<ol style="list-style-type: none"> 1. Begin 2. Source node N_s in AOMDV broadcast route request (RREQ) message to the intermediate node N_i if they are reachable. // route discover process 3. Accumulate each intermediate node address in the RREQ message and forward the RREQ message till reached to destination node N_d. 4. N_s receives route reply (RREP) messages from N_d through accumulated intermediate nodes and create an entry to main routing table per path response packet of destination (node disjoint paths). 5. These paths are used to evaluate more feasible paths using Differential Evolution (DE) algorithm. 6. Apply Differential Evolution (DE) (Algorithm 2) process on node disjoint paths (population) P. 7. Feasible paths are stored in Backup routing table. //These paths are used by source node as backup routes in route recovery process. 8. End

4.3.2 Path recovery phase

The data packets are delivered via the primary path till the primary path is disconnected. When a node detects a link failure, it utilises the backup paths in place of primary path. This is done with the help of RERR message when a node faces the link or node failure it will send a RERR message to the source node that initiates the routing process. So the source node chooses the backup path from backup routing table in path recovery phase instead of finding a new route by reroute discovery process.

Figure 6 Route recovery phase

Algorithm 2	Differential Evolution based AOMDV (DE_AOMDV)
<pre> // Input: F is a scaling factor (0-1). //Input: Cr is Crossover probability (0-1). //output: feasible paths from node disjoint paths using Differential Evolution. 1. Evaluate the initial population P (node disjoint paths) of random individuals. 2. While stopping criteria not met, do //set max. iterations 3. for each individual parent P_i (i=1, 2,popSize) from P do 4. Pick at random 3 distinct individual from the current population P_{i1}, P_b, P_w from P where i₁, b, w are pair wise different, b is the best vector and w is the worst vector from the population. 5. Create mutant vector U = P_{i1}+F*(P_b-P_w). 6. Set candidate C (child) as the result of discrete recombination of U and P_i with crossover probability C_r. 7. If f(C)>=f(P(i)) then // f is an objective function to check feasibility 8. set the trial vector V(i)=P(i) // new created population 9. else 10. set the trial vector V(i)=C 11. end if 12. end for 13. for each individual P_i in the population do 14. set P(i) = V(i) 15. end for 16. end while 17. return feasible paths P(i). </pre>	

5 Performance criteria

The performance of the DE algorithm highly depends on the chosen trial vector generation strategy and associated parameter values used. In appropriate choice of strategies and parameters may lead to premature convergence or stagnation, which can be defined by Table 1.

Table 1 Performance evaluation of different strategies in DE

<i>Strategy</i>	<i>NP</i>	<i>CR</i>	<i>F</i>	<i>Convergence speed</i>
DE/rand/1	5D	0.1	0.5	Fast (multimodal)
DE/best/1	3D–8D	0.3–0.9	0.6	Premature (multimodal)
DE/rand-to-best/1	20D	0.9	0.5	Fast (unimodal)
DE/best/2	20 D	0.4	0.2	Premature (unimodal)
DE/rand/2	20D	1	0.5	Fast (both)

In this section, we compare the proposed DE algorithm-based AOMDV (DE_AOMDV) with the AOMDV protocol in MANETs. Each mobile node in the network starts its journey from a random location to a random destination with a randomly taken speed. Once the destination is reached, after a pause by the mobile node, another random destination is targeted. 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 DE with the population size: 10, 50, 100 and 200. In general, if the population size increases in the DE, the probability of finding the optimal solution increases.

For example, if we take a fixed value of population as 50, then the value of two parameters minimum routing cost and average execution time can be evaluated as shown in Table 2.

Table 2 Performance criteria for different routing protocol

<i>Algo.</i>	<i>AOMDV</i>	<i>DE_AOMDV</i>
<i>Parameter</i>		
Min. routing cost	600	500
Avg. time execution	0.271	0.240

Table 3 Simulation parameters

<i>Parameters</i>	<i>Value</i>
S-2 version	2.34
OS	Ubuntu 13.10
Simulation area	1,000 m × 1,000 m
No. of nodes	50
Channel	Wireless channel
Routing protocol	AODV
Simulation time	200 s
Traffic type	CBR, TCP
Packet size	512 bytes
Optimisation techniques	AOMDV, DE_AOMDV
Mobility model	Random way mobility

The result shows that the DE-based AOMDV (DE_AOMDV) protocol takes a shorter time than the AOMDV. Finally, for the routing problem in the MANETs, we observe that the proposed DE algorithm (DE_AOMDV) can efficiently solve this problem in terms of

routing cost and transmission delay. It is pertinent to solve the problem within a reasonable execution time.

NS-2 network simulator used to simulate behaviour of network on a simulator. Protocol is implemented over NS-2 Simulator as in Table 3. NS2 provides a way to design and evaluate new network protocols, comparing different protocols and simulating their corresponding behaviours.

Performance evaluated is based on the basis of these performance metrics namely: throughput, packet delivery ratio and end-to-end delay.

- a *Throughput (bits/s)*: throughput is the measure of the number of packets successfully transmitted to the destination per unit time as in Figure 7. Throughput of AOMDV is very low at less number of nodes but as we increase the number of nodes, throughput is very high in DE_AOMDV as compared to AOMDV.
- b *Packet delivery ratio*: it is the ratio of data packets delivered to the destination to those generated by means of the sources. PDR is calculated by dividing the number of packet received by destination through the number packet originated from source as in Figure 8. In DE_AOMDV, the packet delivery ratio is always high as compared to AOMDV because DE_AOMDV provided more optimal paths so the number of dropped packets is decreased.
- c *End-to-end delay*: end-to-end delay signifies the average time taken by packets to reach from one end to other as in Figure 9. DE_AOMDV takes less time to reach destination as compared to AOMDV.

A MANET protocol should function effectively over a wide range of networking contexts for small, collaborative, ad hoc groups to larger mobile and multi-hop networks. In summary, the networking opportunities for MANETs are intriguing and the engineering tradeoffs are very challenging.

Figure 7 Throughput in MANET's (see online version for colours)

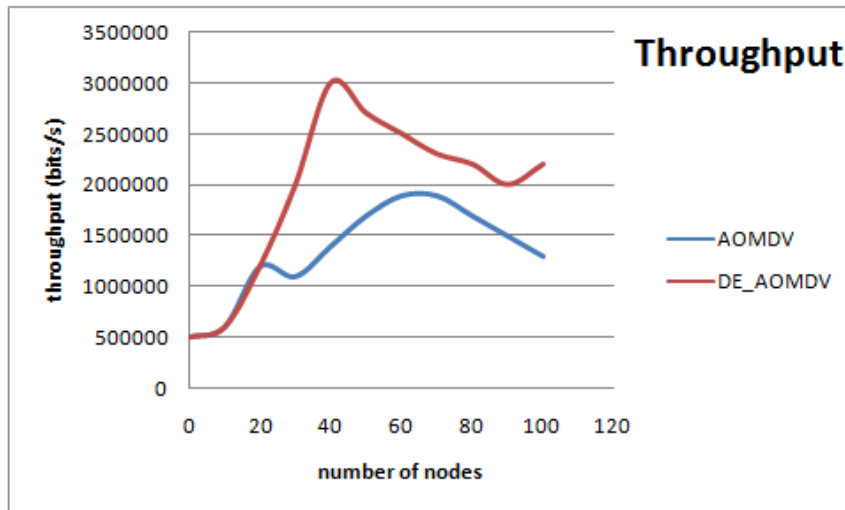


Figure 8 Packet delivery ratios in MANET's (see online version for colours)

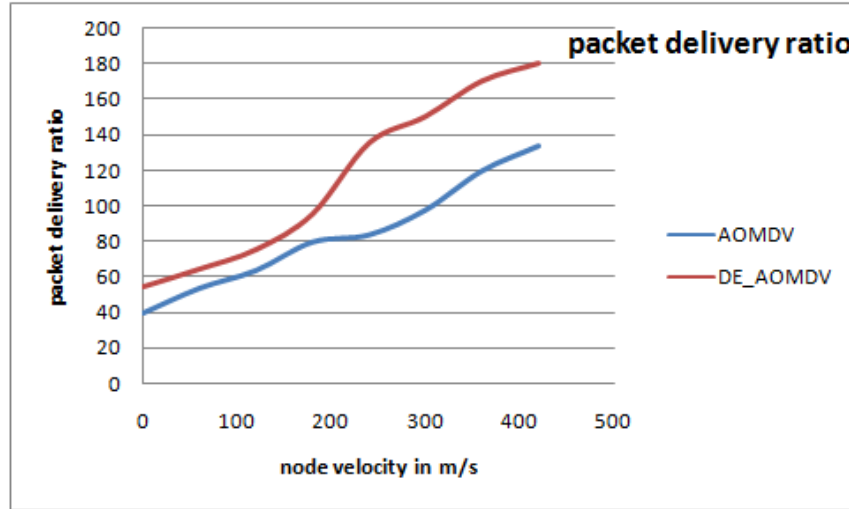
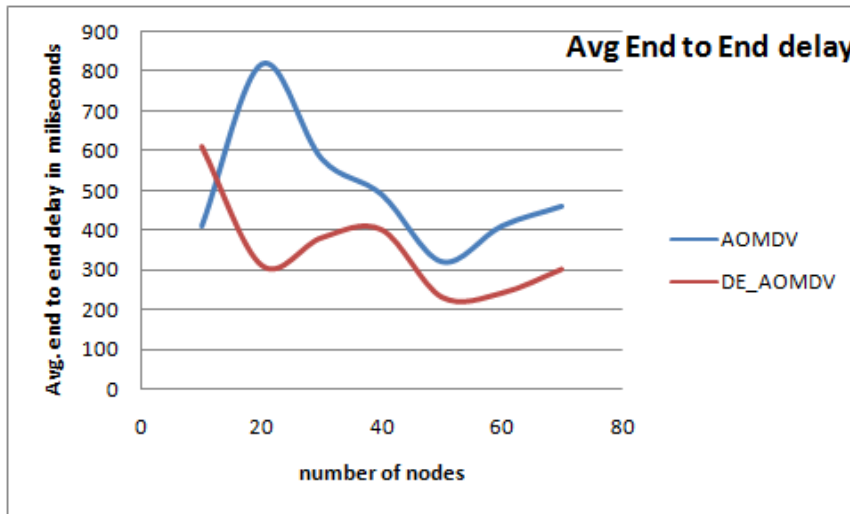


Figure 9 Average end to end delay in MANET's (see online version for colours)



6 Conclusions

In this paper, a DE-based AOMDV (DE_AOMDV) protocol for Mobile ad hoc network is proposed and the performance of the DE_AOMDV protocol has been compared to that of some other well known AOMDV protocol. From the simulation studies, it was observed that the convergence speed of the DE_AOMDV is significantly better than the AOMDV presented. Evaluation considered a range of realistic mobility models and included special cases such as high density, high mobility of nodes. The proposed routing

protocol provides backup routes to increase efficiency and avoid reroute discovery. The comparative study of DE_AOMDV with the AOMDV in terms of end-to-end delay, packet delivery ratio and throughput is described in this paper. Results based on simulation and data analysis proves that DE-based routing protocol is better than AOMDV.

References

- Abbas, A.M. and Jain, B.N. (2004) 'Path diminution in disjoint multipath routing for mobile ad hoc network', *IEEE*, pp.130–134.
- Abbas, A.M., Khandpur, P. and Jain, B.N. (2004) 'NDMA: a node disjoint multipath ad hoc routing protocol', *Proc. Fifth World Wireless Congress (WWC)*, Delson Group, San Francisco, Calif., pp.334–339.
- Ali, M.M. and Torn, A. (2004) 'Population set based global optimization algorithms: some modifications and numerical studies', *Computer Operation Research*, Vol. 31, No. 10, pp.1703–1725.
- Al-Nahari, A. and Mohamad, M.M. (2016) 'Receiver-based ad hoc on demand multipath routing protocol for mobile ad hoc networks', *PLoS ONE*, Vol. 11, No. 6, p.e0156670.
- Aswani, V., Praveen, V. and Thangavelu, S. (2015) 'Performance analysis of variants of differential evolution on multi-objective optimization problems', *Indian Journal of Science and Technology*, August, Vol. 8, No. 17.
- Baburaj, E. and Vasudevan, V. (2008) 'An enhanced tree based MAODV protocol for MANET'S using genetic algorithm'
- Daru, P. and Bingbing, X. (2008) 'An ant routing algorithm for wireless mesh network', in the *7th World Congress on Intelligent Control and Automation, (WCICA)*, pp.4595–4599.
- Das, I., Lobiyal, D.K. and Katti, C.P. (2016) 'An analysis of link disjoint and node disjoint multipath routing for mobile ad hoc network', *International Journal Computer Network and Information Security*, Vol. 3, pp.52–57.
- Das, S. et al. (2008) 'Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives', studies in *Computational Intelligence (SCI)*, Vol. 116, pp.1–38.
- Deb, K. (2000) 'An efficient constraint handling method for genetic algorithms', *Computer Methods in Applied Mechanics and Engineering*, Vol. 186, Nos. 2–4, pp.311–338, Elsevier, Netherlands.
- Elhoseny, M., Farouk, A., Zhou, N. et al. (2017a) 'Dynamic multi-hop clustering in a wireless sensor network: performance improvement', *Wireless Pers. Commun.*, Vol. 95, No. 4, pp.3733–3753.
- Elhoseny, M., Tharwat, A., Farouk, A. and Hassanien, A.E. (2017b) 'K-coverage model based on genetic algorithm to extend WSN lifetime', *IEEE Sensors Lett.*, August, Vol. 1, No. 4.
- Goldberg, D.E. (1989) *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley, Reading.
- Gundry, S. and Kusyk, J. (2012) 'Performance evaluation of differential evolution based topology control method for autonomous MANET nodes', *IEEE*, pp.228–233.
- Holland, J. (1975) *Adaptation in Natural and Artificial Systems*, Univ. of Michigan Press, Ann Arbour.
- Huang, X. and Fang, Y. (2009) 'Performance study of node-disjoint multipath routing in vehicular ad hoc networks', *IEEE Transactions on Vehicular Technology*, May, Vol. 58, No. 4.
- Ibrahim, I.S., King, P.J.B. and Pooley, R. (2009) 'Performance evaluation of routing protocols for MANET', *Fourth International Conference on Systems and Networks Communications*, IEEE, pp.105–112.

- Jayakumar, G. and Ganapathy, G. (2007) 'Performance comparison of mobile ad-hoc network routing protocol', *IJCSNS International Journal of Computer Science and Network Security*, Vol. 7, No. 11, pp.77–84.
- Kennedy, J. and Elberhart, R. (1995) 'Particle swarm optimization', in *Proceedings of IEEE International Conference on Neural Networks*, pp.1942–1948.
- Kennedy, J., Elberhart, R. and Shi, Y. (2001) *Swarm Intelligence*, Morgan Kaufmann, Los Altos, CA.
- Lee, H. and Jeon, D. (2015) 'A mobile ad-hoc network multi-path routing protocol based on biological attractor selection for disaster recovery communication', *ICT Express*, September, Vol. 1, No. 2, pp.86–89.
- Lee, S-J. and Gerla, M. (2000) *SMR: Split Multipath Routing with Maximally Disjoint Paths in Ad Hoc Networks*, Technical report, August, Computer Science Department, University of California, Los Angeles.
- Li, Z., Wang, R. and Bi, J. (2009) 'A multipath routing algorithm based on traffic prediction in wireless mesh networks', *Intern. Conf. on Natural Computation*, pp.115–119, Tianjin, China.
- Marina, M.K. and Das, S.R. (2001) 'On-demand multipath distance vector routing in ad hoc networks', in *Proc. IEEE ICNP*, November, pp.14–23.
- Matre, V. and Karandikar, R. (2016) 'Multipath routing protocol for mobile ad hoc networks', *Colossal Data Analysis and Networking*, DOI: 10.1109/CDAN.2016.7570946.
- Morrison, R.W. (2004) *Designing Evolutionary Algorithms for Dynamic Environments*, Springer-Verlag, Berlin, Germany.
- Mueller, S. and Tsang, R.P. and Ghosal, D. (2012) 'Multipath routing in mobile ad hoc networks: issues and challenges'.
- Paulose, N. and Paulose, N. (2016) 'Comparison of on demand routing protocols AODV with AOMDV', *International Journal of Science, Engineering and Technology Research (IJSETR)*, Vol. 5, No. 1, pp.181–184.
- Perkins, C.K. and Bhagat, P. (1994) 'Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers', in *Proceedings of ACM SIGCOMM*, August, pp.234–24.
- Perkins, C.K. and Royer, E.M. (1999) 'Ad hoc on-demand distance vector routing', *Proceedings of IEEE Workshop on Mobile Computing Systems and Application*, February, pp.90–100.
- Qin, A.K. and Suganthan, P.N. (2005) 'Differential evolution algorithm with strategy adaptation for numerical optimization', in *Proc. IEEE Congr. Evolut. Comput.*, September, Edinburgh, Scotland, pp.1785–1791
- Qin, A.K., Huang, V.L. and Suganthan, P.N. (2009) 'Differential evolution algorithm with strategy adaptation for global numerical optimization', *IEEE Transactions on Evolutionary Computation*, April, Vol. 13, No. 2.
- Riad, A.M., El-minir, H.K. and El-hoseny, M. (2013) 'Article: secure routing in wireless sensor networks: a state of the art', *International Journal of Computer Applications*, April, Vol. 67, No. 7, pp.7–12.
- Royer, E.M. and Toh, C.K. (1999) 'A review of current routing protocols for ad-hoc mobile wireless networks', *IEEE Personal Communications*, pp.46–55.
- Sidhu, D., Nair, R. and Abdallah, S. (1991) 'Finding disjoint paths in networks', *Proc. ACM Conference on Communications Architecture and Protocols (SIGCOM)*, ACM Press, Zurich, pp.43–51.
- Singh, N. and Yadav, R.P. (2007) 'Performance comparison and analysis of table-driven and on-demand routing protocols for mobile ad-hoc networks', *International Journal of Information Technology*, Vol. 4, No. 2, pp.101–109.
- Storn, R. and Price, K. (1997) 'Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces', *J. of Global Optimization*, December, Vol. 11, pp.341–359.

- Suresh Kumar, D., Manikandan, K. and Saleem Durai, M.A. (2011) 'Secure on-demand routing protocol for MANET using genetic algorithm', *International Journal of Computer Applications*, April, Vol. 19, No. 8, pp.0975–8887.
- Suurballe, J.W. (1974) 'Disjoint paths in a network', *Networks*, Vol. 4, No. 2, pp.125–145.
- Vanaja, K. and Umarani, R. (2011) 'An analysis of single path AODV vs. multipath AOMDV on link break using NS-2', *International Journal of Electronics and Computer Science Engineering*, pp.1080–1086.
- Wang, A-I.A., Kuenning, G.H. and Reiher, P. (2003) 'Multipath routing in ad hoc networks', in Makki, K., Pissinou, N., Makki, K.S. and Park, E.K. (Eds.): *Mobile and Wireless Internet*, Chapter 10, Springer, Boston, MA.
- Yang, S., Cheng, H. and Wang, F. (2010) 'Genetic algorithms with immigrants and memory schemes for dynamic shortest path routing problems in mobile ad hoc networks', *IEEE Transactions on Systems, man, and Cybernetics – Part C: Applications and Reviews*, January, Vol. 40, No. 1, pp.52–63.
- Yuan, X., Elhoseny, M., El-Minir, H.K. and Riad, A.M. (2017) 'A genetic algorithm-based, dynamic clustering method towards improved WSN longevity', *Journal of Network and Systems Management*, Vol. 25, No. 1, pp.21–46.