Contemporary Engineering Sciences, Vol. 9, 2016, no. 31, 1525 - 1533
HIKARI Ltd, www.m-hikari.com
https://doi.org/10.12988/ces.2016.69152

LeapFrog and Particle Swarm Optimization based Multipath Routing for VANETs

M. Bhagyavathi and V. Saritha

SCOPE, VIT University, Vellore, India

Abstract

The research is more focused on the application of the optimization algorithms only for single path routing. In this paper, a multipath routing designed using leapfrog algorithm and particle swarm optimization (LFPSOM) is proposed. The leapfrog algorithm is used to estimate the future position of the nodes and predetermine the link breakage. So, the paths which are expected to breakdown in the near future can be updated without losing the packets. The particle swarm optimization (PSO) is used to determine the optimum multiple paths for transmission. The update mechanism in this process is based on the leapfrog mechanism. The proposed algorithm enhances the performance in terms of the QoS parameters like throughput, packet delivery ratio and end-to-end delay. The comparison of the performance of the proposed algorithm is compared with Adhoc on Demand Multipath Routing (AOMDV) and the same are projected.

Keywords: Leapfrog algorithm, Particle Swarm Optimization, Multipath routing, optimization, VANET

1 Introduction

Vehicular Adhoc Network (VANET) is a wireless network in which the nodes are the vehicles with special devices installed. Nodes use their special devices in the transmission and reception of information. VANET require specially designed protocols as the vehicles move very fast when compared to the nodes in the other networks like MANET. There are many challenges in designing the routing protocols for VANET. Basically, there are two types of routing protocols – proactive and reactive routing protocols [1, 10]. The proactive routing protocols predetermine the paths and they are maintained in the form of routing tables.

Reactive routing protocols determine the paths when there is a requirement. These protocols do not maintain any routing table as the network topology will be very dynamic. To take the advantage of both the types of protocols, one can design hybrid protocols where the nodes move for some time and be motionless for some time or some nodes move and some nodes be motion less. Another classification of routing protocols is the single path routing and multipath routing. In single path routing, only one path is determined from the source to destination. The disadvantage of single path routing is more time consumption in route re-establishment. Special mechanisms need to be employed to make it fault-tolerant. Multipath routing means determine more number of paths from source to destination node and is fault-tolerant as the transmission can be carried out even though after a failure in one of the paths.

Multipath routing protocols are more advantageous for VANET when compared to single path routing protocols [8]. One more advantage of multipath routing is fast transmission of packets from source to destination but the disadvantage is the reception of the packets will not be in order. The multiple paths determined may be ‘n’ shortest paths, node disjoint paths or link disjoint paths, etc.

In VANET, the nodes move very fast. Hence there is a high probability of link breakage. So, if the link breakage could be estimated before, it will be very useful and helpful in reducing the packet drops. This is possible using leapfrog algorithm. Leapfrog algorithm [7] is a modified version of Verlet algorithm [6] which helps in predicting the position and velocity of the node during time ‘t + 2’ if the position of the node during time ‘t’ and ‘t + 1’ is known.

Particle swarm optimization [9, 14, 15, 17, 19] is based on the behaviour of birds or fishes. The possible solutions are considered as particles initially. They are divided into groups and the weak particle is identified. This weak particle will be updated and replaced with the better particle. There are local and global best solutions. During iterations, the local best solution may become the global solution. This process is repeated until the optimum global solution is obtained. This applied to determine the best multiple paths in this paper.

The rest of the paper is organized as follows: section 2 presents the related work which has been carried out in this area. The proposed algorithm is discussed in section 3 and section 4 presents the results and discussions. Finally, conclusions are made in section 5.

2 Related Work

In [15], the selection of route is done in two steps. In the first step, the selection is based on the grade. The grade determines the strength of the node. Particle swarm optimization technique is applied on the result of the first step to find the optimal path. A particle swarm optimization is used for routing in cognitive radio networks in [19]. The possible paths from the source to destination are considered as initial candidates. The fitness of the candidates is measured with respect to the throughput, transmit power and delay. Each time, when the nodes move from place to place, the paths are updated. In [17], particle swarm optimization technique is employed to determine paths in multicast routing and Steiner tree problem. The proposed algorithm is grounded on jumping PSO [14]. Problems can be classified as continuous and discrete. The basic PSO algorithm best suits for problems which are continuous in nature. It has been modified and presented in [9] to make suitable for discrete problems. Later many variations of discrete PSO are introduced to solve the problems like scheduling, travelling sales man, etc. [2, 16, 20]. Two heuristics are used for local search to obtain better solution. The PSO is combined with ENS [11] strategy and GRASP [13] algorithm to solve vehicle routing problem in [12]. GRASP is used to identify the initial candidates, PSO is used to update the fitness values of the candidates and ENS is used for local search.

In [4], the shuffled complex evolution approach and PSO are combined and proposed a shuffled leap frog algorithm. In this algorithm, the initial candidates are generated and given weight or rank. They are divided into groups based on the ranks and worst candidate is replaced with the better candidate in each group. Then the candidates in all groups are rearranged and finally the best ideal solution is obtained. This shuffled leap frog algorithm is modified in [3]. The paths from source to destination are considered as frogs. The modification is done in the update process. The number of hops, residual energy and congestion status are used to determine the fitness function of the path. The nodes that are eligible for forwarding the packets to the next hop are determined by defining a parameter called node forwarding satisfaction rate and this is depending on the congestion or the level of traffic at the node. Using this node forwarding satisfaction rate, path fitness value is calculated. The authors in [3] state that the shuffled leap frog algorithm proposed in [4] might experience the drawback of obtaining the optimum path locally only.

The modification in the update process is that the node level update is made in [3] whereas path level update is made in [4] and also two learning variants are introduced in [3] and the update process depend on these variants.

3 LeapFrog and Particle Swarm Optimization based Multipath Routing Protocol (LFPSOM)

The nodes in the VANET can be considered as vertices and the links among them can be considered as edges when VANET is represented as a graph. Let the source node be represented as ‘S’ and destination node as ‘D’. Then the initial candidates or particles in LFPSOM are the possible paths from node ‘S’ to node ‘D’. Depth first traversal is used to determine the possible paths and the scenario shown in Fig. 1 is considered for illustration.

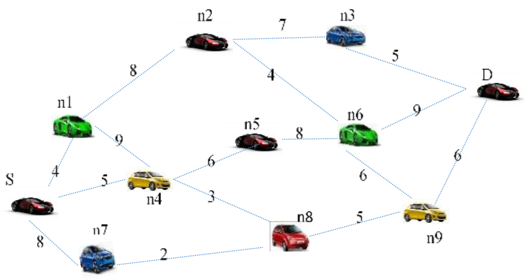


Fig 1: Sample network

Possible paths from source to destination

Path P1: S→n1→n2→n3→D	Path P2: S→n1→n2→n6→D
Path P3: S→n1→n4→n5→n6→D	Path P4: S→n1→n4→n5→n6→n9→ D
Path P5: S→n1→n4→n8→n9→D	Path P6: S→n4→n8→n9→D
Path P7: S→n4→n5→n6→n9→D	Path P8: S→n4→n5→n6→D
Path P9: S→n7→n8→n9→D	

The link stability of all the links of all determined multiple paths is calculated. The links which are weak will be eliminated and the corresponding paths are removed and the remaining paths are considered for the further process.
Consider there are ‘n’ links in the path
Let us assume the link stability of each link as ls_1, ls_2, ls_3, …, ls_n.
Then the stability of the path ‘P’ = min (ls_1, ls_2, ls_3, …, ls_n).
Determine the path stability of all the paths from source ‘S’ to destination ‘D’ using their corresponding link stabilities of Fig. 1.

Path P1: 4	Path P2: 4	Path P3: 4
Path P4: 4	Path P5: 3	Path P6: 3
Path P7: 5	Path P8: 5	Path P9: 2

The links which are weak are eliminated based on the threshold value. Here the threshold value is set as 3. Therefore, the links with link stability less than or equal to 3 are eliminated. Hence the links n4 → n8 and n7 → n8 are eliminated and so, the paths P5, P6 and P9 are not considered for the further process. The path stability is considered as the fitness function of the particle.
The particles are now distributed among different sets. For this, the paths are considered in the ascending order of their fitness value. The paths are distributed as the path with least fitness value to the first set, path with second least fitness value to the second set and so on. If ‘N’ sets and ‘M’ paths are there, where N < M, then after assigning N paths to N sets, N+1th path goes to the first set again, N+2th path goes to the second set and the process is repeated until all the paths are assigned to

the sets. Hence, the assignment of paths to various sets is done in a circular mode. The procedure of assigning paths P1, P2, P3, P4, P7 and P8 to various sets is shown in Fig 2. At each time, the best path among each set is used to transmit the packets. Hence, the multiple paths are determined for the transmission of packets. Hence, path P7 and P8 are considered for transmission.

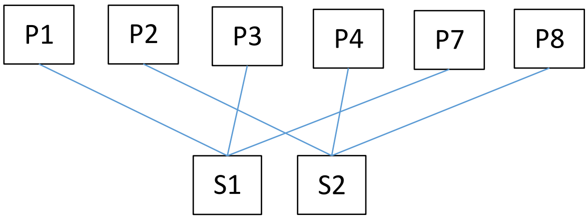
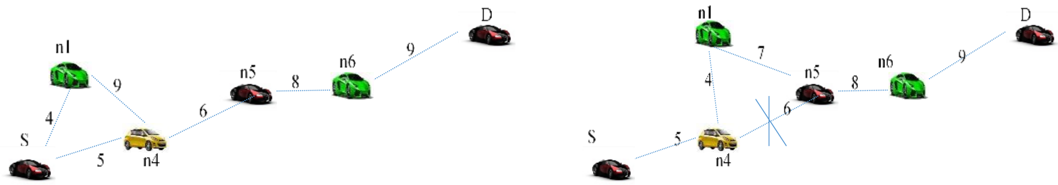


Fig. 2 Assignment of paths to Sets

Generally, the number of sets to be considered depends on the application. The number of sets determines the number of multiple paths that can be used for transmission. This can be optimally decided using learning algorithm which is considered as future work of this proposal. The stability of the link is estimated on beforehand using leapfrog algorithm. Leapfrog algorithm is used to find the position of the node at time 't+2' if the position of the node and velocity with which the node is moving is known at time 't' and 't+1'. The position and velocity of the nodes is obtained using GPS and the future position of the nodes is estimated using leapfrog algorithm. When the future position of the nodes is known, it can be determined that whether the corresponding two nodes will be connected or not at time 't+2'. If it is determined that the nodes get disconnected at time 't+2', then the path is updated by replacing the weak link with another strong link(s). Consider the path 8 from Fig. 1 and the scenario is shown in Fig. 3 (a) and assume that the nodes n4 and n5 disconnect in the path from 'S' to 'D' at time 't+2' and node n1 comes between n4 and n5.



(a) At time 't+1' (b) At time 't+2'

Fig 3. Path from Source node 'S' to destination node 'D'

Then the path from source node 'S' to destination node 'D' can be updated by replacing the link n4 → n5 with n4 → n1 and n1 → n5 as shown in the Fig. 3 (b).

Whenever the path is updated, the path stability is measured again. Now find the path stability for the updated path (path P8: 4). Shuffle all the paths from all the sets and redistribute among the sets again.

4 Results and Discussions

The proposed algorithm LFPSOM is simulated using NS-2 and verified for the performance in terms of packet delivery ratio, end-to-end delay and throughput. It is compared with AOMDV protocol and the results projected indicate that LFPSOM performs better. MOVE tool is used to develop the mobility model and is integrated with NS-2 where LFPSOM is simulated.
The two algorithms AOMDV and LFPSOM are compared and the results for the comparison in terms of end to end delay are shown in Fig 4. The delay is less in LFPSOM when compared with AOMDV. The links which are weak will be eliminated and the remaining paths are considered for the further process. Hence, the paths considered for the transmission are stable. This helps in reducing the packet drop.

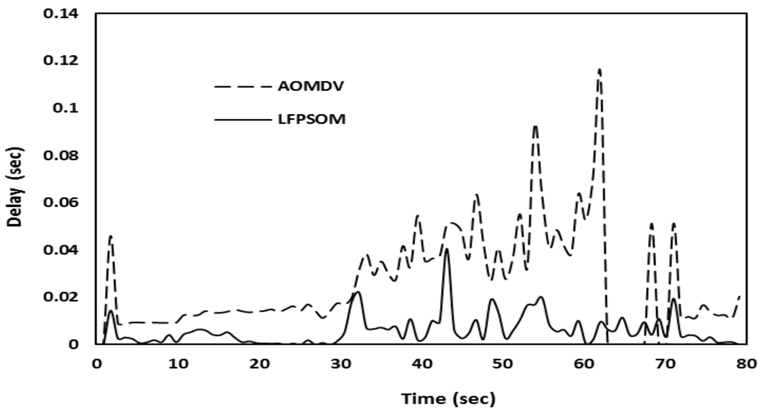


Fig 4. End-to-End Delay vs Time

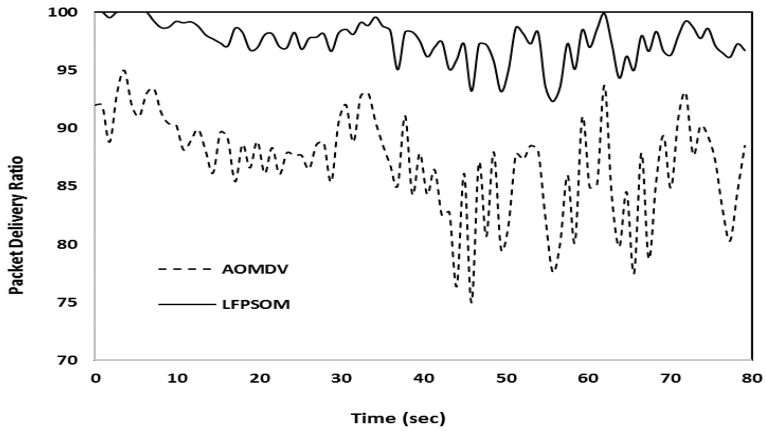


Fig 5. Packet Delivery Ratio vs Time

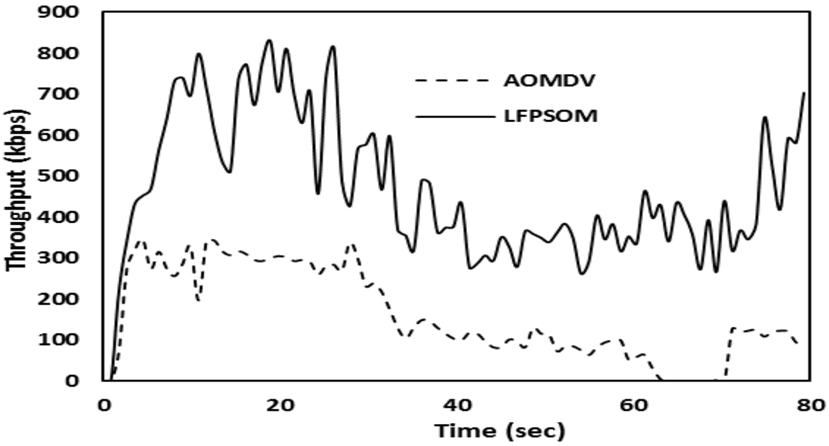


Fig 6. Throughput vs Time

It is further decreased by estimating the link break using the predetermined future position of the nodes. The comparison in terms of throughput is shown in Fig 6. The throughput of network is decreased with the vehicle node density increased; the throughput is better in LFPSOM when compared to AOMDV. The comparison of the packet delivery ratio in two algorithms, AOMDV and LEPSOM is shown in Fig 5. It can be observed that the packet delivery ratio is increased in LEPSOM when compared to AOMDV.

5 Conclusions

Designing a routing algorithm is a challenging and interesting task in VANET because of the high mobility. This paper proposed multipath routing protocol based on LeapFrog and Particle Swarm Optimization. Initially, the stable paths are determined using link stability. The paths are considered as the particles in PSO algorithm. LeapFrog algorithm is used during the update mechanism in PSO. The leapfrog algorithm is used to estimate the future position of the nodes and predetermine the link breakage. The weak links are replaced with strong link(s) in the path which needs to be updated. This helps in reducing the loss of the packets and improves the throughput of the system. The proposed protocol is simulated under a more equitable environment. The proposed algorithm LFPSOM outperforms when compared with AOMDV in terms of end-to-end delay, packet delivery ratio and throughput.

References

[1] Saif Al-Sultan, Moath M. Al-Doori, Ali H. Al-Bayatti, Hussien Zedan, A comprehensive survey on vehicular Ad Hoc network, *Journal of Network and Computer Applications*, **37** (2014), 380-392.
https://doi.org/10.1016/j.jnca.2013.02.036

[2] D. Anghinolfi, M. Paolucci, A new discrete particle swarm optimization approach for the single-machine total weighted tardiness scheduling problem with sequence-dependent setup times, *Eur. J. Oper. Res.*, **193** (2009), no. 1, 73–85. https://doi.org/10.1016/j.ejor.2007.10.044

[3] Jia Dongyao et al., Adaptive multi-path routing based on an improved leapfrog algorithm, *Information Sciences*, **367** (2016), 615-629. https://doi.org/10.1016/j.ins.2016.07.021

[4] Muzaffar Eusuff, Kevin Lansey and Fayzul Pasha, Shuffled frog-leaping algorithm: a memetic meta-heuristic for discrete optimization, *Engineering Optimization*, **38** (2006), no. 2, 129-154. https://doi.org/10.1080/03052150500384759

[5] Niels Gronbech-Jensen and Oded Farago, A simple and effective Verlet-type algorithm for simulating Langevin dynamics, *Molecular Physics*, **111** (2013), no. 8, 983-991. https://doi.org/10.1080/00268976.2012.760055

[6] http://www.fisica.uniud.it/~ercolessi/md/md/node21.html

[7] http://young.physics.ucsc.edu/115/leapfrog.pdf

[8] Xiaoxia Huang and Yuguang Fang, Performance study of node-disjoint multipath routing in vehicular ad hoc networks, *IEEE Transactions on Vehicular Technology*, **58** (2009), no. 4, 1942-1950. https://doi.org/10.1109/tvt.2008.2008094

[9] J. Kennedy, R.C. Eberhart, A discrete binary version of the particle swarm algorithm, *In: Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics 1997*, Piscataway, NJ, (1997), 4104–4109. https://doi.org/10.1109/icsmc.1997.637339

[10] Yun-Wei Lin, Yuh-Shyan Chen and Sing-Ling Lee, Routing Protocols in Vehicular Ad Hoc Networks: A Survey and Future Perspectives, *J. Inf. Sci. Eng.*, **26** (2010), no. 3, 913-932.

[11] Y. Marinakis, A. Migdalas and P. M. Pardalos, A hybrid genetic-GRASP algorithm using Langrangean relaxation for the traveling salesman problem, *Journal of Combinatorial Optimization*, **10** (2005), 311–326. https://doi.org/10.1007/s10878-005-4921-7

[12] Yannis Marinakis and Magdalene Marinaki, A hybrid genetic–Particle Swarm Optimization Algorithm for the vehicle routing problem, *Expert Systems with Applications*, **37** (2010), no. 2, 1446-1455. https://doi.org/10.1016/j.eswa.2009.06.085

[13] Yannis Marinakis, Athanasios Migdalas and Panos M. Pardalos, Multiple phase neighborhood Search—GRASP based on Lagrangean relaxation, random backtracking Lin–Kernighan and path relinking for the TSP, *Journal of Combinatorial Optimization*, **17** (2009), no. 2, 134-156. https://doi.org/10.1007/s10878-007-9104-2

[14] J.A. Moreno-Perez, J.P. Castro-Gutierrez, F.J. Martinez-Garcia, B. Melian, J.M. Moreno-Vega, J. Ramos, Discrete particle swarm optimization for the p-median problem, *In: Proceedings of the 7th Metaheuristics International Conference*, Montreal, Canada (2007).

[15] T.R. Gopalakrishnan Nair and Kavitha Sooda, Particle Swarm Optimization for Realizing Intelligent Routing in Networks with Quality Grading, *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM)*, (2011). https://doi.org/10.1109/wicom.2011.6040139

[16] G.C. Onwubolu, M. Clerc, Optimal path for automated drilling operations by a new heuristic approach using particle swarm optimization, *Int. J. Prod. Res.*, **42** (2004), no. 3, 473–491.

[17] Rong Qu et al., Particle swarm optimization for the Steiner tree in graph and delay-constrained multicast routing problems, *Journal of Heuristics*, **19** (2013), no. 2, 317-342. https://doi.org/10.1007/s10732-012-9198-2

[18] R.E. Wilde and S. Singh, *Statistical Mechanics, Fundamentals and Modern Applications*, John Wiley and Sons, Inc, Newyork, 1998.

[19] D. Simion, A. Graur, L. Alexandru, S. Stefan, A. H. Mahdi, An optimize Particle Swarm Optimization routing algorithm for data transmission in Cognitive Radio Networks, *2012 10th International Symposium on Electronics and Telecommunications (ISETC)*, (2012). https://doi.org/10.1109/isetc.2012.6408148

[20] H. Zhang, H. Li, C.M. Tam, Permutation-based particle swarm optimization for resource-constrained project scheduling, *J. Comput. Civ. Eng.* **20** (2006), no. 2, 141–149. https://doi.org/10.1061/(asce)0887-3801(2006)20:2(141)

Received: September 26, 2016; Published: November 7, 2016