



Location Dynamic Tabu Routing Protocol for MANETs

Armir Bujari¹ · Luigi De Giovanni² · Claudio E. Palazzi¹ · Daniele Ronzani¹

Accepted: 9 March 2021 / Published online: 23 April 2021
© The Author(s) 2021

Abstract

Mobile Ad-Hoc Networks are currently experiencing a second youth in terms of research interest as well as providing benefits to our society. Clearly, this has been fostered by the wide range of applications that have become actually feasible thanks to the pervasive and increasing presence of smartphones, drones, sensors and other small devices with communication and sensing capability. Aiming to deploy solutions able to self-organize without the need for any infrastructure support, packet routing remains an open and critical research problem. Several routing solutions have been conceived to guarantee delivery and low overhead in this context. A promising approach, trying to limit the information needed and stored in the network, is represented by *stateless* solutions, a class of solutions not relying on topology state information. However, since they exploit local knowledge, the achieved performance in terms of packet delivery and latency is not always up to expectations. To this aim, we propose a dynamic routing protocol based on a *tabu search* approach, relying on local network knowledge and in-packet short-term memory to alleviate the local minima problem. A through experimental assessment is conducted, measuring protocol performance under different configurations and profiles, evidencing its benefits.

Keywords Tabu search · Routing protocol · MANET · Network performance

1 Introduction

Wireless networks have been extensively explored and plenty of research effort has been devoted to devise optimal and reliable routing strategies for wireless Mobile Ad-hoc Networks (MANETs), which, nowadays, still remains a major issue. Indeed, the widespread popularity of devices with networking capability such as smartphones, drones, sensors, vehicles, and even bicycles is creating a scenario with many nodes that are currently connected through the

Internet, thus elongating the network path and, even worse, augmenting the overhead burden on the infrastructure [1–3]. Instead, direct (multihop) communication may represent a better choice especially with applications that involve local generation, consumption and propagation of information [4–8].

Involved application scenarios may regard a vast plethora of innovative services improving the wellness of our society in many ways, including agriculture 2.0, industry 4.0, environmental sensing, smart transportation, and smart cities in general [9–14].

In this context, *stateless* routing protocols represent a promising approach for MANETs in which no session information is retained by routers or packets for data packet routing. In contrast to stateful routing protocols, e.g., Ad-hoc On-demand Distance Vector (AODV), stateless solutions handle each forwarding decision as an independent transaction, without the need of storing past events or other state information [15–17]. Therefore, stateless design does not require any memory, which would otherwise be used to store such information [18]. One class of such solutions are position-based routing protocols, which rely on location information of nodes to forward the data. In specific, each node is capable of obtaining its geographic position (e.g., by using an embedded GPS or another geo-location

✉ Armir Bujari
armir.bujari@unibo.it

Luigi De Giovanni
luigi@math.unipd.it

Claudio E. Palazzi
cpalazzi@math.unipd.it

Daniele Ronzani
dronzani@math.unipd.it

¹ Department of Computer Science and Engineering, University of Bologna, Bologna, Italy

² Department of Mathematics, University of Padua, Padua, Italy

system), taking the forwarding decision relying solely on the position of the destination and of its neighbors. However, it has been demonstrated that stateless routing protocols, as strategies based on local knowledge, are limited in terms of packet delivery and latency performance. The major issue still nowadays, regardless whether we consider classic MANETs or specific versions including vehicles or even drones, is represented by the *local minima* phenomenon [19–21]. Greedy approaches are a representative class of stateless protocols, relying on local information (e.g., in terms of minimum distance) forwarding the packet toward a destination.

In this context, intermediate next hop routing decisions are based on selecting the neighbor which has the closest distance, compass setting, or some other measure quantifying the forwarding progress. This approach has the advantage of not requiring maintenance of routing tables or route construction prior to or during the forwarding process [22]. Furthermore, the forwarding process is by design able to adapt to changes in the topology of the network, selecting the next best hop among those actually available in that very moment, without the need for route construction, table maintenance, or any other routing/topology information dissemination. Therefore, greedy solutions generally embody stateless, low overhead and low latency routing solutions for MANETs. On the other hand, in the case no neighbor is associated with a progress towards destination, the packet gets stuck into a local minimum and the algorithm fails. Recovery strategies are typically adopted to escape from local minima; yet, the delivery is still not guaranteed [23, 24].

In this paper, we consider a general MANET, which may include sensors, vehicles, drones, mobile users, etc. and propose the Location Dynamic Tabu (LDT) routing protocol, a tabu-based routing approach to support position-based strategies. Our solution relies on limited additional information stored in the data packet to take past actions into account in order to avoid local minima. Basically, the data packet includes a list of recently visited nodes. The algorithm design follows the tabu search principle, introduced by [25] in the context of combinatorial optimization as a strategy to escape from local optima, in which every past state in the list is considered *tabu* (i.e., prohibited) and hence not chosen for the next state change. The length of this tabu list defines the number of visits that are considered recent and hence the nodes that are considered as tabu. As soon as new nodes enter in the list, old ones are eliminated thus becoming again eligible forwarders. In this article, considerations on the network overhead of the solution and an experimental assessment are discussed, evidencing the benefits of the approach.

The article is organized as follows: in Section 2, we provide a synthetic overview of prior work in the

domain. Section 3 introduces the network model and some notations, while Section 4 details some design choices behind the proposed routing scheme. Section 5 presents the performance evaluation and comparison of our approach to prior work. Finally, Section 6 draws the conclusions and outlines some future work.

2 Related work

The focus on the network overhead for packet forwarding is not recent. In [26], a Depth First Search (DFS) strategy for position-based routing is proposed. Each node, upon receiving a packet, sorts all its neighbors according to a distance (to the destination) criteria and forwards the packet to the first node. The method makes sure the same packet will no longer be managed by that node by storing a list of received packet identifiers in every node. In our approach, in contrast, we reduce the total memory usage by storing only the tabu list, and only one copy of it maintained in the packet itself.

Other proposals try to hybridize geo-routing with classic solutions (e.g., AODV). In [27], the authors discuss Greedy-Closer Request (GCR) an attempt to combine the benefits of both approaches. Whenever the greedy forwarding process stalls, GCR recovers by employing a topology, reactive-based approach such as AODV. The idea is to design a routing protocol able to change its behavior according to the local condition of the network. In particular, the protocol switches between two modes: greedy-mode and creq-mode (closer request-mode). GCR starts with the former, forwarding the packet always to the closest node to the destination. The greedy-mode remains active until a local minimum is reached. Then, the protocol switches to creq-mode, with the aim of looking for a node in the network that is closer to the destination than the local minimum. Several works apply tabu search as a centralized optimization technique to solve different problems arising in MANETs, like network configuration, traffic routing, etc. [28, 29]. Concerning packet forwarding, a tabu-based routing protocol for sensor networks is proposed in [30]. The choice of the next sensor node to route a packet is based on a cost function value. The cost function is determined for each neighbor according to the node's remaining energy and the distance to the sink. The strategy proceeds by forwarding the packet to the neighbor node that optimizes the cost function. A tabu flag associated with each node is set to true when the node is visited by the packet, and reset to false when the packet arrives to the sink. When a node is visited by the packet, its identifier is included in the tabu list. When the packet arrives to the sink, all the flags are reset. Differently from our solution, no decentralized tabu list is maintained and the

mechanisms to exchange the tabu information, as well as the length of the tabu list, are not discussed. Furthermore, no mechanism is considered to deal with the case when all the neighbors are tabu, where the algorithm fails.

Another work that uses tabu search for routing is [29], which proposes a way to find the optimal path from source to destination using a meta-heuristic strategy. The algorithm analyzes the network state and performs an optimization process to minimize an objective function. Differently from our work, this approach requires the existence of an initial solution (a valid path from source to destination) and the complete knowledge of the network by every node. Clearly this is significantly different from our proposal, as we start from not having a path, the process is initiated from the source node, and where each node relies only on local network information, beyond the position of the destination node, as usual in position-based routing.

3 System model and notation

In the aforementioned scenario, we consider a general connected MANET, in which a path from any node to any other one exists. We have considered a network model where nodes have unique IDs and a position. On top of this, a location-based approach is then used for routing. Considering this, the discussed solution is then independent of the specific network protocol that could be employed.

A beaconing scheme, which allows nodes to know the locations of their neighbors, is assumed [31, 32]. Each beacon is a tiny packet containing the node's identifier (e.g., its MAC address) and location, in terms of 2D/3D coordinates. A *neighbor table* maintains the list of all the nodes from which a beacon has been received. By knowing their neighborhood, nodes can forward packets with a specific location forwarding criteria. In this work, we do not explore the beaconing process; we simply assume it exists, as usual for position-based routing protocols, and we keep it in its basic form i.e. a periodic process through which nodes notify their presence to their neighborhood.

3.1 Greedy Routing

The *greedy* algorithm is a single-path, stateless forwarding strategy, appearing in network routing in terms of *progress forwarding*. Using a greedy approach, a packet is forwarded by the current node to a neighboring node which makes a (or the best) progress towards the destination node. Starting from the source node, at each step, the packet is forwarded to the best available neighbor, until the destination node is reached or no other neighbor node can be explored. The term *best* is related with the notion of *progress*; typically, it is defined according to the distance to a destination node.

Hence, the best progress is made by the closest node to the destination. If none of the nodes is closer than the current one, greedy fails, as it is not able to find any progress. The node in which the packet gets stuck is called a *local minimum*. Greedy represents a good solution for high density networks, since it tends to reach the destination in few hops, but suffers from the local minima problem, that can lead to a low packet delivery rate [33].

3.2 Tabu Search

Tabu search is a combinatorial optimization technique based on local search and exploiting memory to escape from local optima [25]. The basic tabu search scheme requires an initial solution, a method to quickly obtain new *neighbor* solutions by modifying an initial one, and a tabu list. The search starts from the initial solution and explores and evaluates its neighbor solutions. The process iteratively moves to the best neighbor, in the attempt of obtaining better and better solutions according to an objective function. To overpass local optima, tabu search allows visiting non-improving neighbors, with the risk of getting stuck in a cycle. To address the problem, the last l visited solutions are stored in a *tabu list* and considered tabu: this excludes cycles of length up to l and, empirically, relatively small values of l prevent cycling at all.

In the context of packet routing, we consider each network node as a solutions of the tabu search scheme, the nodes in the neighbor table as the neighbor solutions, and the notion of progress adopted in the greedy routing (e.g., the minimization of the distance to the destination) as the objective function.

4 Location dynamic tabu routing

Our proposed forwarding algorithm is based on a greedy strategy. During the forwarding process, a tabu list is stored within the data packet containing a list of node identifiers. Every time a node is traversed by the packet, its identifier is added to the tabu list. When a packet is forwarded, nodes included in the tabu list are not considered as possible next-hops toward the destination, and the best available remaining neighbor node is chosen instead.

Generally, the protocol works as mentioned; yet, the following exceptions must be considered. If a packet arrives to a node having all its neighbors marked as tabu, the packet gets stuck in that node, as no progress can be attained. Another issue is the possibility to fall into a cycling path when the tabu list is not long enough to contain all the nodes in that cycle, allowing those no longer *tabu*, to be again selected as next nodes (whereas they should not). Explaining our proposal, we describe a series of improving

heuristics, designed to solve these issues and enhance the performance of the routing algorithm.

4.1 Clearing the tabu list

In case a packet arrives at a node having no eligible neighbors as a valid next hop (i.e. all the neighbors are in the tabu list), the algorithm would fail. To avoid this, we add the following rule: if the neighborhood is empty, then the tabu list is cleared and the forwarding can resume. Figure 1 illustrates an example: the source node *a* starts to forward a data packet, which travels through intermediate nodes towards the destination *n*. While travelling, every node's identifier is stored into the packet's tabu list, until node *e*, a local minimum, is reached (Fig. 1a). Since *d* is a tabu node, no non-tabu neighbor is available and the algorithm would fail; according to our solution, the tabu list is cleared and the greedy approach can resume to forward the packet through a new path, finally reaching *n* (Fig. 1b).

4.2 Dynamic tabu list

Adopting a fixed tabu list length may lead to two extreme cases. A long list, if only few nodes are actually needed to

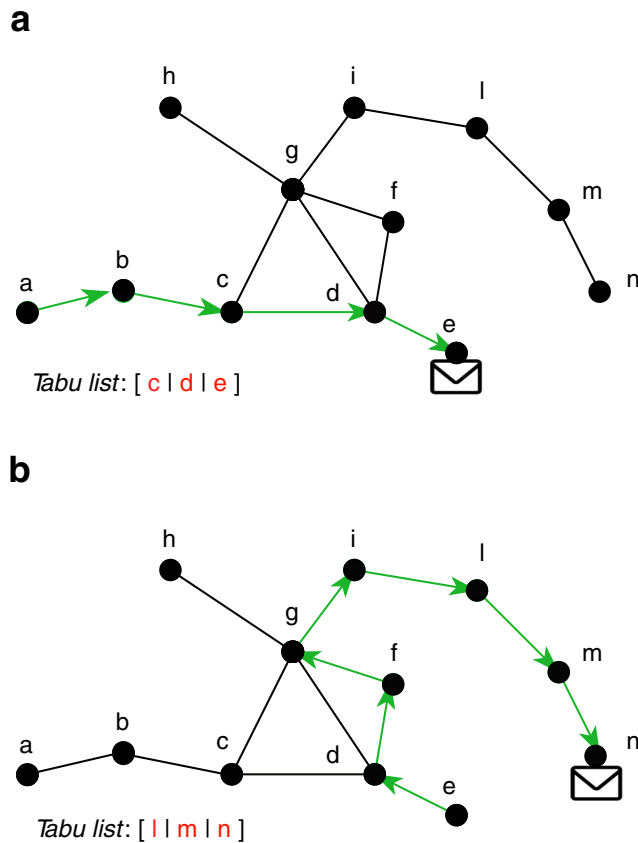


Fig. 1 Tabu Search execution example; a packet is sent by the source node *a* towards the destination node *m* ($l = 3$)

be tagged as tabu, is a waste of memory and causes network overhead. On the other hand, with a short list, we may not have enough space to store tabu nodes such that we prevent packets from getting stuck in a cycle. For example, if we consider Fig. 2, which shows the case $l = 2$, since node *d* is closer to *n* than *i*, the packet will get stuck in the cycle $e - d - f - g - d - e$. To address this issue, we define a dynamic tabu list, which increases or decreases its length when needed. To determine when we need to increase the tabu list's size, we count the number w of successive forwards that did not lead to an improvement. If, after X forwards, the packet ($w = X$), we assume that l is not large enough to deal with this local minimum; hence, we increase l by S , S being the initial tabu list length, and we reset w . Once the packet gets again closer to the destination than the local minimum, the list length is reset to its initial value to only store the S last visited nodes. Algorithm 1 illustrates our LDT routing protocol as executed at each node. The protocol starts at the source node *s* with a tabu list *tl* of initial size S , typically a small value (e.g., 3), stored into the data packet *p*. Other variables are stored in *p*: the initial global distance gd between *s* and the destination *d*, corresponding to the up-to-date minimum distance reached; a counter w (i.e., harmful forwards counter) initialized to 0, counting how many hops have been successively performed without getting closer to *d* with respect to gd ; a *tll* field, accounting for the total number of hops the packet has traversed, initialized to a *TTL_MAX* value and decremented after each forward. The algorithm terminates either when the destination is reached or *tll* becomes 0. As soon as the packet reaches a global progress towards *d*, gd is updated with the new distance and w goes back to 0. Every time a forwarded packet does not reach a shorter distance to *d* than gd , w is incremented by 1. Clearly, in this case, the current gd value represents the distance between the current local minimum *m* and *d*.

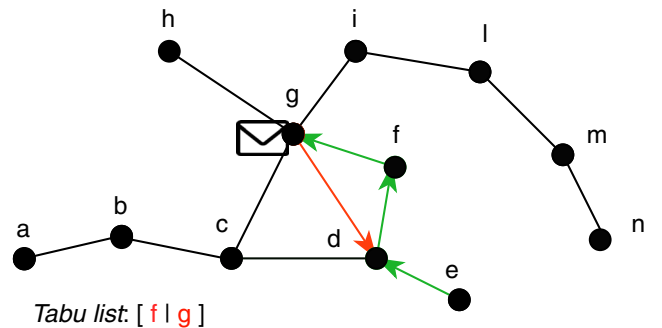


Fig. 2 Tabu Search execution with $l = 2$. The node *g* selects *d* as next node, leading the system into a cycle

Algorithm 1 Location dynamic tabu routing scheme.

```

1: Node  $c$  receives a packet  $p$ 
2: if  $c$  is source node  $s$  then
3:    $p.d =$  packet's destination
4:    $p.gd \leftarrow dist(s, p.d)$ 
5:    $p.w \leftarrow 0$ 
6:    $p.tl =$  empty tabu list of initial size  $S$ 
7:    $p.ttl = TTL\_MAX$ 
8: end if
9: if  $c = p.d$  then
10:   $p$  delivered to destination
11:  exit
12: end if
13: if  $p.ttl \leq 0$  then
14:  discard packet
15:  exit
16: end if
17: if  $c$ 's neighborhood is empty then
18:  erase  $p.tl$ 
19: end if
20: add  $c$  into  $p.tl$ 
21:  $u \leftarrow$  closest  $n_i$  to  $p.d$  in  $c.N$  not in  $p.tl$ 
22: if  $dist(u, p.d) < p.gd$  then
23:    $p.m \leftarrow u$ 
24:    $p.gd \leftarrow dist(u, p.d)$ 
25:    $p.w \leftarrow 0$ 
26:   reset  $p.tl$  size to  $S$ 
27: else
28:    $p.w \leftarrow p.w + 1$ 
29:   if  $p.w > W$  then
30:     increase size of  $p.tl$  by  $S$ 
31:   end if
32: end if
33:  $p.ttl = p.ttl - 1$ 
34: send  $p$  to  $u$ 

```

5 Performance evaluation

In this section, we evaluate the performance of our proposal against classic routing solutions. Following, is a second set of experiments aimed at comparing the scheme employing a dynamic tabu list approach against the fixed size strategy. In both cases, the considered metrics include the delivery ratio and incurred overhead. Simulations have been conducted in a custom Java-made simulator (code available in [34]), specifically implemented for this study. Our simulator allows to specify nodes' position and movements as well as the rules for packet forwarding. We have preferred to use our simulator instead of existing ones such as, for instance, NS2 or NS3, as we wanted to focus

only on the forwarding process without the details about the wireless physical model/layer, the transport/application protocols, etc. Indeed, these details could have hidden properties, behaviours or problems specifically related to the considered protocol mixing routing aspects with unrelated ones.

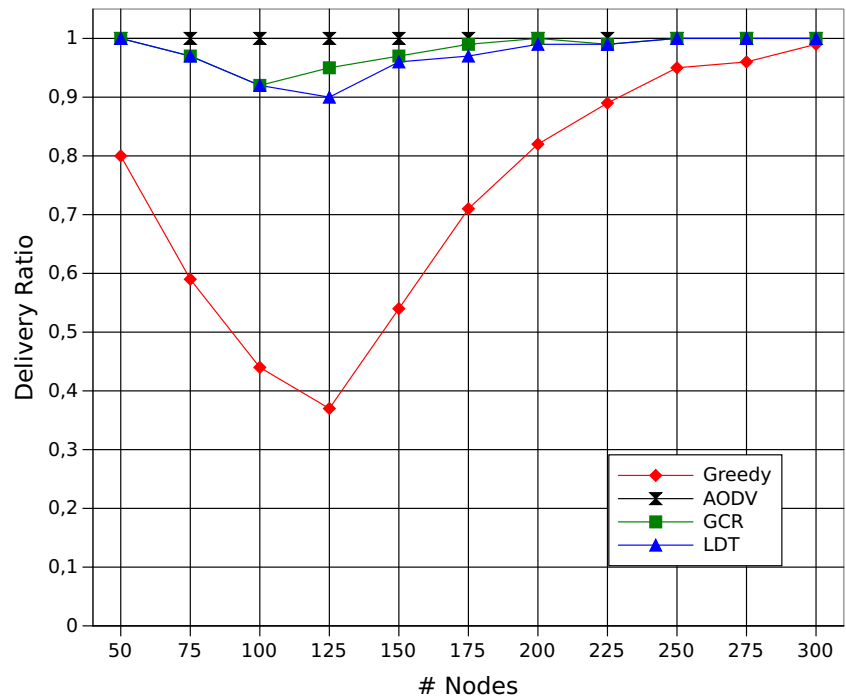
Our aim is that of studying the behavior of analyzed routing protocols in a general MANET, considering a wide range of application scenarios which may include sensors, vehicles, drones, mobile users, etc. in smart homes, campuses, cities, farms and industries. We believe that this generality is crucial when evaluating a new routing mechanism. To this aim we have considered randomly generated topologies with different node density, thus covering several possible use cases and going beyond basic scenarios. Of course, future work may include the analysis of specific use cases.

5.1 Comparison with classic solutions

To demonstrate the benefits of employing LDT, we compared its performance against a classic topology-based routing algorithm, AODV, and two position-based ones: Greedy and GCR. We consider a 2D scenario characterized by a square area of $750m \times 750m$ and a number of nodes varying from 50 to 300 with an increment step of 25. Nodes are placed randomly while the placement procedure makes sure the network is connected. For each number configuration, we perform a total of 500 runs, each employing a topology. Every run simulates the transmission of a single packet from a random source to a random destination. The transmission range is set to 100m. The tabu size is set to 3 and TTL_MAX to 100.

Results show the performance in terms of delivery ratio, number of hops and number of nodes involved by the transmission of a packet, number of routing packets transmissions and total overhead. Each value in the charts is the average computed over the total number of runs. As seen in Fig. 3, LDT is able to reach a delivery ratio of almost 90% even in the worst, low network density, case (125 nodes). This is a very good result as it shows that our solution achieves delivery ratios similar to protocols that intensively use memory (e.g., AODV). Although the average number of hops needed to reach the destination is slightly higher in the worst case, as shown in Fig. 4, from 200 nodes on, the average number of hops is similar to the other solutions. In terms of involved nodes (Fig. 5), with respect to AODV and GCR, LDT does not need to involve many intermediate nodes, since it does not transmit any control packet, e.g., Request Packets (RREQ) in case of AODV. This is also reported in Fig. 6, showing that the amount of control packets used by LDT is always 0, as for Greedy. This makes

Fig. 3 Packet delivery ratio with varying node density



LDT scalable for high density networks: indeed, the higher the network density, the higher the delivery ratio, up to 100% (see Fig. 3), without increasing the control overhead and number of traversed hops.

The routing overhead reported in Fig. 7 corresponds to the amount of bytes sent during the routing process to deliver a single data packet. For the sake of generality, we just take into account the routing information strictly

needed to the single packet routing (e.g., reserved bits field of the RREQ packet is not considered when computing the overhead). Results show that LDT’s performance is better or comparable to those of other state-of-the-art protocols, as one may expect, since LDT relies only on the information in the tabu list for its forwarding process.

The only exception is Greedy, which, however, is the worst in terms of delivery, as seen in Fig. 3. As expected,

Fig. 4 Average number of hops traversed

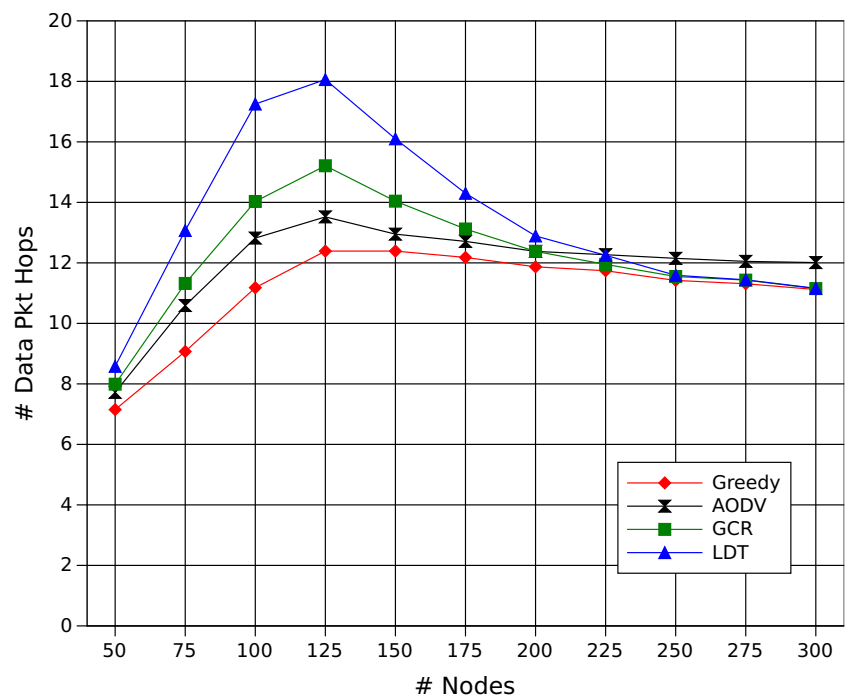
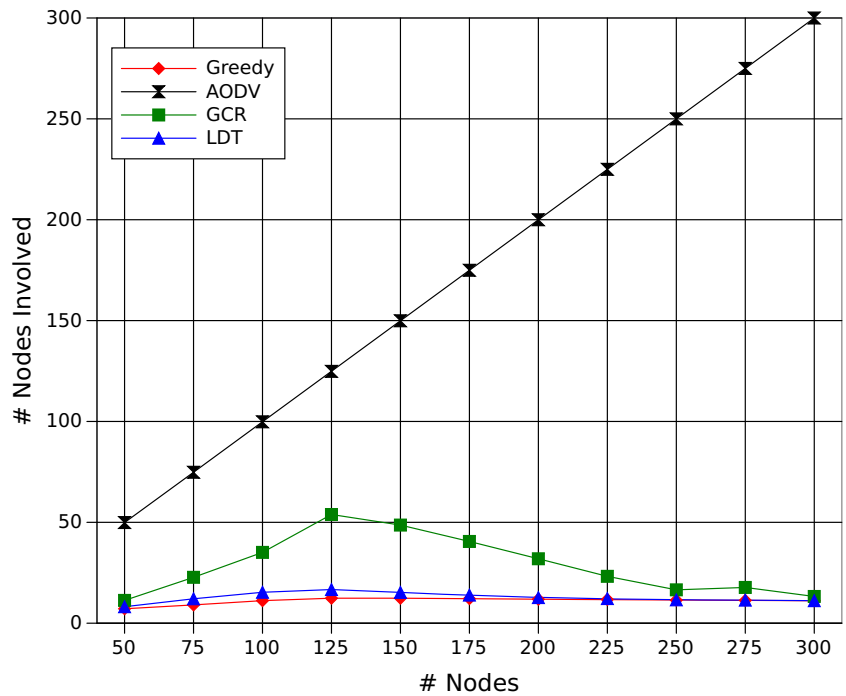


Fig. 5 Number of nodes involved at least once in forwarding the data packet



AODV produces significant routing traffic, which increases proportionally with the number of nodes due to its broadcast phase.

5.2 Static vs dynamic

In order to demonstrate the effectiveness of the dynamic feature in LDT, a comparison with similar routing methods,

but using a fixed tabu list length, is presented. In this case we consider five different tabu list sizes: 5, 10, 15, 20, 25; each instance is performed in topologies of 50, 100, 150 and 200 nodes, respectively. Finally, the dynamic version (LDT) with an initial tabu size of 3 nodes is considered. The simulation results are shown in Fig. 8 (delivery ration), Fig. 9 (average number of hops), and Fig. 10 (average tabu list length).

Fig. 6 Average number of control packets sent

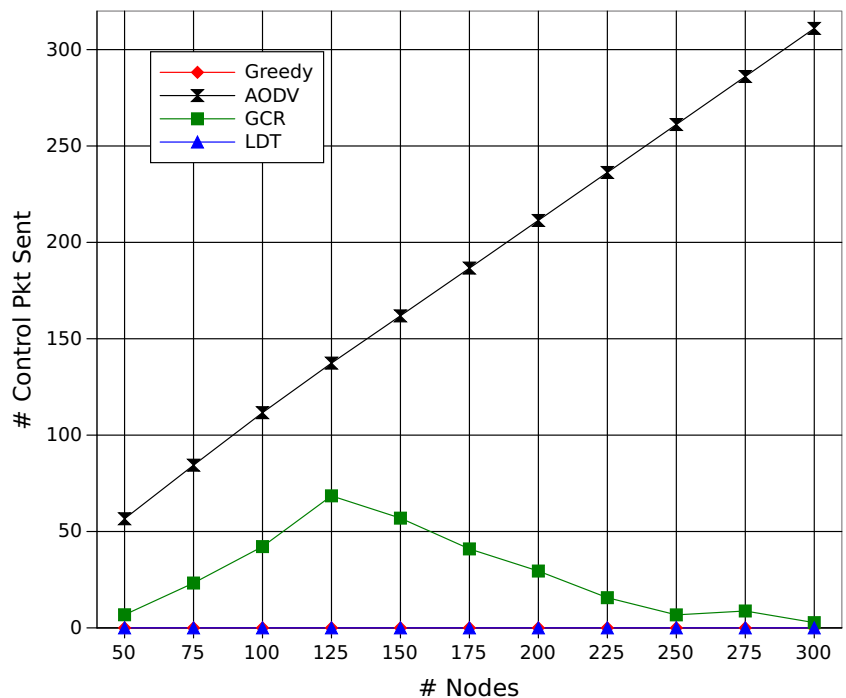
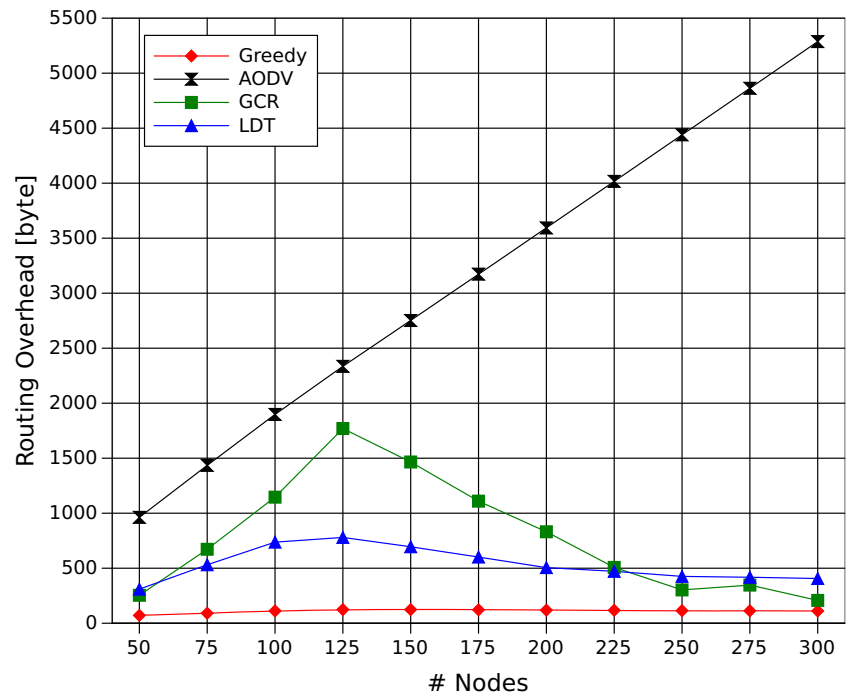


Fig. 7 Routing overhead measured in terms of bytes



As expected, LDT is always able to provide the highest data delivery when compared to the strategy employing a fixed length (Fig. 8). Indeed, any choice of length gets associated sometimes to a local minimum that can instead be solved with the use of a longer tabu list, as achieved by LDT thanks to its adaptability.

On the other hand, in order to address some local minima, LDT has to try different tabu list lengths, in sequence, until finding one that allows to actually proceed towards destination. The ability to overcome a local minimum depends on the ability to attempt longer explorations of nodes around and this is ensured by longer tabu list lengths.

Yet, finding a way to reach a destination around complex local minima implies the use of long, tortuous, paths with a high number of hops. This is clearly visible in Fig. 9, where using a longer tabu list or a dynamic one (even a very long one if needed) as LDT is associated to a higher average number of hops: the average is, in fact, affected by those cases where some complex local minima was solved thanks to a long tabu list.

Furthermore, LDT has another advantage over the fixed tabu list alternative: it employs on average a shorter list length (Fig. 10). Indeed, LDT uses a very short tabu list (only three elements as initial setting) with the exception of

Fig. 8 Packet delivery ratio with varying node density

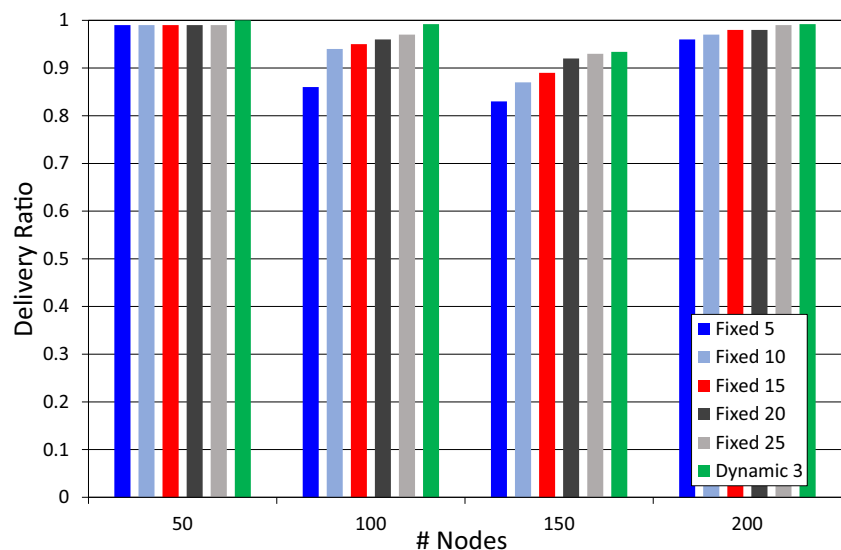
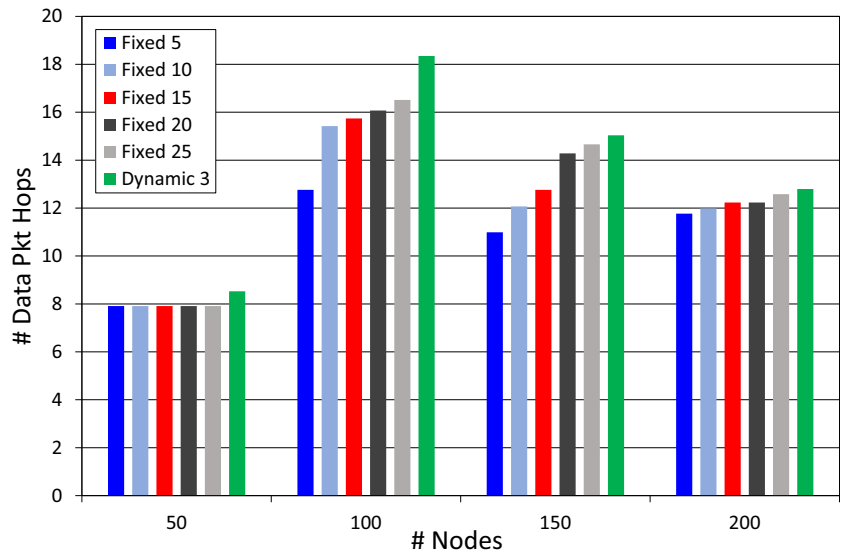


Fig. 9 Average number of hops traversed by the data packets



those cases where a longer one is useful to overcome local minima. Hence, using a dynamic tabu list provides a good tradeoff solution between data delivery and low overhead. Yet, an upper limit could of course be set to prevent an excessive growth of the list and, as a consequence, of the packet size.

6 Conclusion

We have designed and implemented a tabu-based routing scheme for MANETs, named LDT, introducing a heuristics to complement the local knowledge of the greedy algorithm. Comparing our strategy with other state-of-the-art solutions, we can see significant benefits in terms of routing overhead and generated traffic, at the cost of slightly extending, in

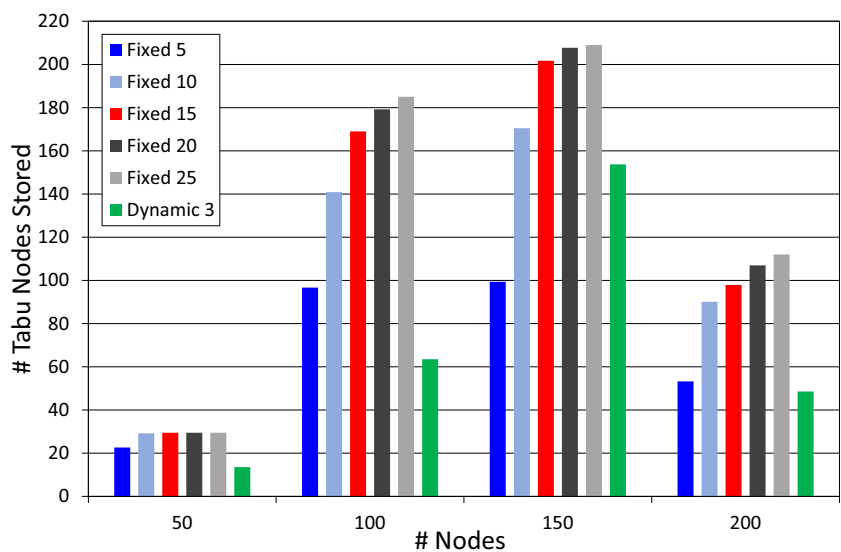
some cases, the travelled path towards destination. Actually, for high density networks, the performance improves without any overhead drawback.

Moreover, significant improvements in terms of routing overhead (involved nodes and forwarded bytes) are obtained for the critical case of low density networks, where LDT provides an interesting trade-off between overhead and delivery ratio.

In conclusion, we have demonstrated that adopting a decentralized light memory-based heuristic (with a little information, the tabu list, added only in the packet), bound with a typical position-based routing protocol, may be a good trade-off between opposite philosophical approaches and a successful strategy for packet routing in MANETs.

This work can of course be expanded in several directions. To this aim, we intend to compare our solution

Fig. 10 Total number of forwarded node IDs per configuration, in average



with alternative protocols and testing it in specific case studies (e.g., smart homes, campuses, cities, farms and industries [35, 36]) as well as applications (e.g., sensing, tourism, gaming [37–39])).

Acknowledgements This work is partially funded by the Department of Mathematics of the University of Padua, Italy, through the BIRD191227 project. Responsibility for the presented content resides with the authors.

Funding Open access funding provided by Alma Mater Studiorum - Università di Bologna within the CRUI-CARE Agreement.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Mostafa A et al (2018) A V2X-based Approach for Reduction of Delay Propagation in Vehicular Ad-Hoc Networks. In: Proceedings of IEEE International Conference on ITS Telecommunications, St Petersburg, Russia
- Amoroso A, Marfia G, Roccetti M (2011) Going Realistic: Optimal A Distributed Multi-hop Broadcast Algorithm for Vehicular Safety. *Comput Netw* 55(10):2504–2519
- Aguiari D et al (2018) Canarin II: Designing a Smart E-bike Ecosystem. In: Proceedings of IEEE Consumer Communications & Networking Conference (CCNC), Las Vegas
- De Francesco C, De Giovanni L, Palazzi CE (2018) The Interference-aware Drone Ad-hoc Relay Network Configuration problem. *Electron Notes Discret Math* 69:317–324
- Anjum SS, Noor RM, Anisi MH (2015) Survey on MANET based communication scenarios for search and rescue operations. In: Proceedings of International Conference on IT Convergence and Security (ICITCS), Kuala Lumpur
- Mirri S, Prandi C, Salomoni P, Monti L (2016) Social Location Awareness: A Prototype of Altruistic IoT. In: Proceedings of IFIP International Conference on New Technologies, Mobility and Security (NTMS), Larnaca
- Bujari A, De Giovanni L, Palazzi CE (2018) Optimal configuration of active and backup servers for augmented reality cooperative games. *Concurr Comput Pract Exper* 30(20):e4454
- Marfia G, Roccetti M, Amoroso A, Gerla M, Pau G, Lim J-H (2011) Cognitive cars: Constructing a cognitive playground for VANET research testbeds. In: Proceedings of International Conference on Cognitive Radio and Advanced Spectrum Management (CogART), Barcelona
- Alvear Ó, Calafate CT, Cano J-C, Manzoni P (2018) Crowdsensing in smart cities: overview, Platforms, and Environment Sensing Issues. *Sensors* 2:18
- De Rango F, Potrino G, Tropea M, Santamaria AF, Fazio P (2019) Scalable and Lighthway Bio-inspired Coordination Protocol for FANET in Precision Agriculture Applications. *Comput Electr Eng* 74:305–318
- Bononi L, Donatiello L, Furini M (2009) Real-Time Traffic in Ad-Hoc sensor networks. In: Proceedings of IEEE International Conference on Communications (ICC), Dresden
- Bujari A, Conti M, De Francesco CC, Palazzi E (2019) Fast Multi-hop Broadcast of Alert Messages in VANETs: An Analytical Model. *Ad Hoc Netw* 82:126–133
- Bedhief I, Foschini L, Bellavista P, Kassar M, Aguiari T (2019) Toward Self-Adaptive Software Defined Fog Networking Architecture for IIoT and Industry 4.0. In: Proceedings of IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)
- Prandi C, Monti L, Ceccarini C, Salomoni P (2020) Smart campus: Fostering the community awareness through an intelligent environment. *Mob Netw Appl* 25(3):945–952
- Perkins CE, Royer EM (1999) Ad-hoc On Demand Distance Vector Routing. In: Proceedings of IEEE Workshop on Mobile Computing Systems and Applications, New Orleans
- Perkins CE, Belding-Royer EM, Das SR (2002) Ad hoc On-Demand Distance Vector (AODV) Routing, draft-ietf-manet-aodv-10.txt IETF MANET working group
- Bujari A, Gaggi O, Palazzi CE, Ronzani D (2018) Would Current Ad-Hoc Routing Protocols be Adequate for the Internet of Vehicles? A Comparative Study. *IEEE Internet Things J* 5(5):3683–3691
- Tropea M, Santamaria AF, De Rango F, Potrino G (2019) Reactive flooding versus link state routing for FANET in precision agriculture. In: Proceedings of IEEE Consumer Communications & Networking Conference (CCNC), Las Vegas
- Khan IU et al (2020) Smart IoT Control-Based Nature Inspired Energy Efficient Routing Protocol for Flying Ad Hoc Network (FANET). *IEEE Access* 8:56371–56378
- Ghaffari A (2020) Hybrid opportunistic and Position-Based routing protocol in vehicular ad hoc networks. *Springer J Ambient Intell Human Comput* 11:1593–1603
- Kaliyamurthi B, Palanisamy A (2021) Geographic routing with hybrid firefly algorithm and galactic swarm optimization for efficient 'Void' handling in mobile ad hoc networks. *Wiley Int J Commun Syst* 34:e4690
- Lemmon C, Lui SM, Lee I, Forwarding Geographic (2009) Routing for Ad-hoc Wireless Network A Survey. In: Proceedings of International Joint Conference on INC, IMS and IDC, Seoul
- Karp B, Kung HT (2000) GPSR: Greedy Perimeter Stateless Routing for Wireless Networks. In: Proceedings of the Annual International Conference on Mobile Computing and Networking, pp 243–254
- Rao SA, Pai M, Boussedjra M, Mouzna J (2008) GPSR-L: greedy perimeter stateless routing with lifetime for VANETS. In: Proceedings of IEEE International Conference on ITS Telecommunications
- Glover F (1989) Tabu search-Part i. *ORSA J Comput* 1:190–206
- Stojmenovic I, Russell M, Vukojevic B (2000) Depth First Search and Location Based Localized Routing and QoS Routing in Wireless Networks. In: Proceedings of ICPP, pp 173–180
- Ronzani D, Bujari A, Palazzi CE (2018) A Hybrid Reactive and Position-based Approach to Packet Routing in 3D Topology Networks. In: Proceedings of IFIP Wireless Days (WD), pp 9–14
- De Giovanni L, Palazzi CE (2013) Optimal Client-Server configuration of mobile Ad-Hoc networks. *Electron Notes Discret Math* 41:495–502
- Jang K-W (2011) A Tabu Search Algorithm for Routing Optimization in Mobile Ad-hoc Networks. *Telecomm Sys - TELSIS* 51:11

30. Semchedine F, Bouallouche-Medjkoune L, Bennacer L, Aber N, Aïssani D (2012) Routing protocol based on tabu search for wireless sensor networks. *Wirel Pers Comm* 67:105–112
31. Alsaqour R et al (2015) Dynamic Packet Beaconing for GPSR Mobile Ad Hoc Position-based Routing Protocol Using Fuzzy Logic. *J Netw Comput Appl* 47:32–46
32. Choi J, Khaled Y, Tsukada M, Ernst T (2008) IPv6 Support for VANET with Geographical Routing. In: *Proceedings of International Conference on ITS Telecommunications*, pp 222–227
33. Bujari A, Palazzi CE, Ronzani D (2018) A Comparison of Stateless Position-based Packet Routing Algorithms for FANETs. *IEEE Trans Mob Comput* 17(11):2468–2482
34. Mobilelab UNIPD, Location Dynamic Tabu Routing Protocol for MANETs [online]. Available: <http://mobilelab.math.unipd.it/sw-geo-tabu.html> Last Accessed: January 2021
35. Vegni AM, Little TDC (2010) A message propagation model for hybrid vehicular communication protocols. In: *Proceedings of IEEE, IET International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, Northumbria University, Newcastle
36. Sahoo S, Gupta Y, Nath V, Pal S (2021) Ad Hoc Network Using UAVs in Indian Farms: A Review. *Nanoelectron Circ Commun Syst* 692:765–770
37. Gaggi O (2013) Discovering Local Attractions from Geo-tagged Photos. In: *Proceedings of the 28th Annual ACM Symposium on Applied Computing (SAC)*, Coimbra
38. Bujari A, Ciman M, Gaggi O, Marfia G, Palazzi CE (2015) Paths: Enhancing Geographical Maps with Environmental Sensed Data. In: *Proceedings of ACM MobileHealth 2015/MobiHoc 2015*, Hangzhou
39. Hamdi MM et al (2020) A Review of Applications, Characteristics and Challenges in Vehicular Ad Hoc Networks (VANETs). In: *Proceedings of the International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, Ankara

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.