

Table-driven geographical routing using aggregate areas

Mandimby Ranaivo Rakotondravelona*, Bruno Baynat†, Fanilo Harivelo* and Pascal Anelli*

*Laboratoire d'Informatique et de Mathématiques (LIM)

Université de la Réunion

Email: {mandimby.ranaivo, fanilo.harivelo, pascal.anelli}@univ-reunion.fr

†Laboratoire d'Informatique de Paris 6 (LIP6)

Email: bruno.baynat@lip6.fr

Abstract—Geographical routing protocols scale well in large ad-hoc and sensor networks but fall short with some topologies. This causes packet losses and a drop in network performance. In this paper, we propose to introduce topological information aiming to improve the quality of routing decision while keeping the protocol scalable. Every node maintains precise information for nearby nodes and aggregated information for farther nodes by means of aggregate areas. Evaluation demonstrates the scalability of the proposition and its efficiency compared to a pure geographical routing protocol.

Index Terms—Ad hoc networks, geographical routing, table-driven routing, aggregation.

I. INTRODUCTION

Ad-hoc and sensor networks involve nodes interacting with each other over multihop connections without central management in an infrastructure-less environment [1]. One of the main issues in such self-organized networks is scalability in terms of routing. With the numerous potential applications of ad-hoc and sensor networks (Internet of Things, disaster recovery, military application ...) scalability is an important property. It allows the network to grow in scale without quick performance loss. For instance, a scalable routing protocol still ensures high packet delivery rate even with increased number of nodes and also without excessive cost.

Geographical routing approaches are known to scale well in large ad-hoc and sensor networks. They require nodes to maintain existing links to their one-hop neighboring nodes. But additionally, geographical information are needed to perform forwarding. The nodes have to be aware of their location by means of GPS or alternative techniques [2]. Forwarding can be summed up as selecting one of the one-hop neighbors which is geographically closer to the destination. Routing decision are purely local. Because the list of existing links to be maintained by each node is small and network size-independent, these approaches are scalable. However, due to lack and/or limited use of topological information, geographical routing performs poorly with some network topologies. Nodes ignore or are not aware of the availability of some links in the network. Chosen paths are not necessarily optimal. In extreme cases, forwarding fails even if routes exist to the destination. One noticeable consequence of such failure is the appearance of dead-ends called *local maxima*. A local maximum is a node

which has not any neighbor geographically closer to the destination, hence forwarding can make no more progress. Several techniques allows to circumvent these dead-ends as described in [3], among them, flooding, geometric-based or planarization. Nonetheless, they have limitations such as the need of bidirectional links or nodes with equal radio range [4].

We investigate the possibility to enrich a position-based routing with topological information. With an exclusive topology-based approach, each node has a more or less complete knowledge of every other nodes in the network [5]. More precisely, a list of existing links in the network is maintained at any time (proactive protocols) or determined when needed (reactive protocols). Forwarding decisions are taken based on these link information. One of the advantages of these approaches is the precision as routing is performed only on known links. But they face scaling problems. As a matter of fact, in large networks, the list of existing links can be huge and too costly to maintain.

On one hand, we aim to improve the quality of routing decisions. On the other hand, we would like to keep the scalability of the overall protocol. We provide routing with information about links, and, thus, existing nodes. But not all information is necessary for a given node. Far destinations don't require high precision routes to be reached as routing decision is taken on every hop. Let's imagine a traveler who wants to join his booked hotel room in a foreign country. First, he has to travel to the destination country. It doesn't require very detailed information so far. On arrival, he must find the town where his hotel is located by consulting, for instance, a map of the country. As he gets closer, he now needs more detailed information. And so on, arrived in the town, he will need very sharp details in order to find the quarter and the street where his hotel is located. By analogy, routing needs only more and more details as the destination comes closer. Based on this principle, we propose a scalable approach as it gives more precision to localized routing information while summarizing the global ones. For a node, the list of nearby nodes is maintained as reachable hosts in the routing table. Farther nodes are gradually aggregated into larger and larger areas as they are located away from the considered node. We aggregate nodes by their geographical coordinates.

Our proposition relies on the following components:

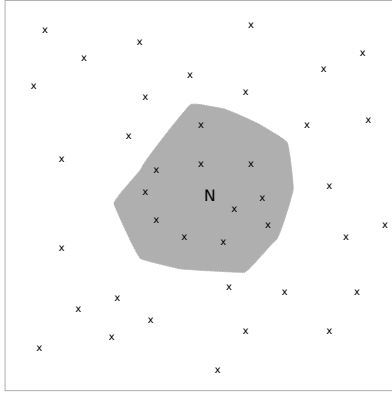


Fig. 1. Local area

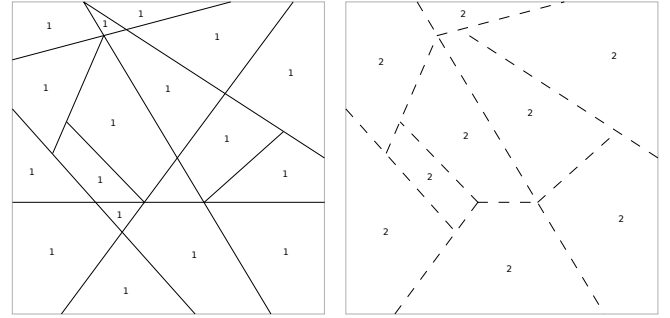
- At some point, we need to aggregate some nodes into area. Aggregation is done on location-basis. We, then, partition the whole deployment space into geographical areas, that we call primary aggregate partitions or areas, shared and known by all nodes in the network. Primary partitions can be aggregated, in their turn, to form larger partitions to make it possible to refer to faraway nodes. They are uniquely identified and known by all nodes in the network, and so on;
- Known nodes and areas are maintained in a routing table by every node;
- Nodes exchange periodically routing information to keep their routing table up-to-date;
- An unified forwarding strategy consists in sending a packet to the gateway of the entry (either a node or an area) closest to the destination.

Mixing topology-based routing and location-based routing was already proposed in the literature [6], [7]. We make differentiation by using only one type of routing key (the destination location), a tight integration of topology-based and location-based components (forwarding decision relies solely on the routing table) and more flexible and user-defined settings to tune the tradeoff between precision and cost related to the maintenance of routing information thanks to aggregation. We demonstrate through simulations that our proposition makes larger use of existing paths while keeping the routing table size low.

II. TABLE-DRIVEN GEOGRAPHICAL ROUTING

A. Local area

Each node has its own local area. It is a limited geographical area surrounding a node. Its size can be defined in terms of hop counts, i.e., it is the area containing nodes at a given maximum number of hops from the considered node. The size can also be limited by geographical distances. In this case, the local area is defined by $d(N, P) \leq l$ where d is a distance function, N is the considered node, P is a point in the network zone and l is the maximum allowed distance from N to be part of the local area.



(a) Order-1

(b) Order-2

Fig. 2. Example of aggregate areas

A node retains link information to all the nodes inside its local area. On the Fig.1, the local area of the node N is colored in grey.

B. Aggregate areas and network zone partitioning

Link information to nodes inside the network are aggregated by means of geographical areas called aggregate areas. We use the following principles to design aggregate areas:

- We define primary aggregate areas that can be uniquely identified;
- Aggregate areas can be, in turn, aggregated into larger aggregate areas that are also uniquely identified;
- The aggregation level of each area is known.

We define multilevel aggregate areas: order- n areas are aggregated into order- $n + 1$ where n is the aggregation level. The smallest aggregate areas (primary or order-1) result from a global partition of the network zone.

On Fig.2a, the network zone has been partitioned into order-1 aggregate areas. Then on Fig.2b, order-1 areas have been, in turn, aggregated into order-2 ones. The result is a global partition of the network into order-2 aggregate areas.

C. Table structure

From a node point of view, the routing table reflects its knowledge of, on one hand, the link information to nodes inside its local area and, on another hand, aggregated link information to nodes inside aggregate areas. Thus there are two parts inside the routing table:

- Local area part: containing route to nodes inside the local area. Each line in this part indicates how to join a specific node inside this area;
- Aggregate areas part: containing information about aggregate areas known to contain at least one node. Each line of this part gives information on how to join a specific order- n aggregate area.

The Table I sums up the structure of a routing table.

- *Area (A)* indicates the nature of the route, either leading to a node (local) or an order- n aggregate area (order- n aggr.). N is the highest order;
- *Destination (D)* tells the ID of the destination of the route which can be a node or an aggregate area;

TABLE I
ROUTING TABLE STRUCTURE

Area(A)	Destination(D)	Gateway(GW)	Cost(C)
Local	Node ID	Node ID	...
Local
Order-1 aggr.	Node ID Area ID	Node ID Node ID
Order-1 aggr.
Order-2 aggr.	Area ID	Node ID	...
Order-2 aggr.
Order-2 aggr.	Area ID	Node ID	...
Order-2 aggr.
Order-N aggr.	Area ID	Node ID	...
Order-N aggr.
Order-N aggr.	Area ID	Node ID	...
Order-N aggr.

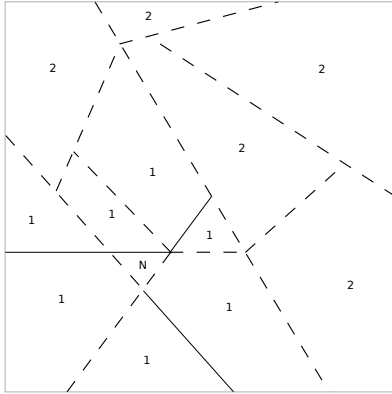


Fig. 3. Aggregate areas of N

- *Gateway (GW)* is the ID of the next hop node through which the destination can be reached;
- *Cost (C)* indicates the cost of going from the current node to the destination according to a specific metric (e.g. hop count).

It is important to note that these information are the minimal necessary to be retained in case of a static network. Additional information are required to handle mobility (which is out of the scope of this paper) such as sequence numbers [8].

D. Updating tables

Because of the table-based nature of our protocol, one of the key points is the updating process of the table content. We propose an updating mechanism which is similar to what is used in DSDV [8]. Every node periodically broadcasts its table content to all of its one-hop neighbors. Restraining the broadcast to the one-hop neighbors saves bandwidth yet allow the routing information to spread across the network.

Aggregation decisions. As stated earlier, a node's routing table reflects its knowledge of, one one hand, the link information to nodes inside its local area and, on another hand, aggregated link information to nodes inside aggregate areas. In other terms, information outside the local area are aggregated. In order to optimize the scalability of our solution, we pose the following principle: with great distance comes high level

of aggregation. More precisely, from a node perspective, the further is the node corresponding to a link information, the higher is the order of the aggregate area containing it. The order of the aggregate area is function of the distance between the node receiving the link information and the node concerned by the latter:

$$order = \begin{cases} 1 & \text{if } 0 < d(N, P) \leq x_1 \\ 2 & \text{if } x_1 < d(N, P) \leq x_2 \\ \dots & \\ n & \text{if } x_{n-1} < d(N, P) \leq x_n \end{cases}$$

where N is the center node (owner of the table), P is a point in the network zone, d is a distance function and $x_1 < x_2 < \dots < x_n$ are reference distances.

The Fig.3 shows the aggregate areas (with their respective order) considered by the node N.

Initialization. At start up, a node adds its own information to its table: the *area* column is filled with local, the *destination* with its own ID, the *gateway* also with its own ID and the *cost* with zero.

Receiving updates. Let N a node receiving a routing message for a destination D from one of its neighbors noted GW. The routing table of N will be referred as $T(N)$. N will follow the next algorithm in order to update its routing table given this new information. In the following, C' is an update of the cost C after routing update processing.

- If D is a node:
 - If D is located inside the local area:
 - * If there is not yet route to D inside $T(N)$, add the route {local, D, GW, C' };
 - * If there is already a route to D inside $T(N)$, it will be replaced if the new route offers a lower cost.
 - If D is located outside the local area:
 - * Identify the order-1 aggregate area containing D;
 - * Restart the algorithm with the latter area as entry D.
- If D is an aggregate area:
 - Find the smallest aggregated area known by N containing D, let A the resulting aggregate area whose order is n :
 - * If there is not yet route to A inside $T(N)$, add the route {Order- n aggr., A, GW, C' };
 - * If there is already a route to A inside $T(N)$, it will be replaced if the new route offers a lower cost.

Example. On the Fig.4 we have a representation of the local area and aggregate areas of a node N. We assume that N's table has just been initialized so contains only one entry (its own). Now, N receives the following routing messages {Destination, C} from a node that will be referred as GW:

- {A, C}: A is situated inside the local area, so N adds the line {local, A, GW, C' } to its table;

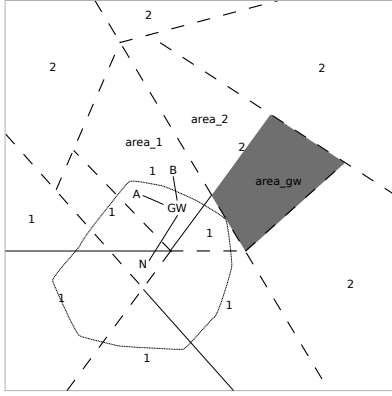


Fig. 4. Receiving routing information

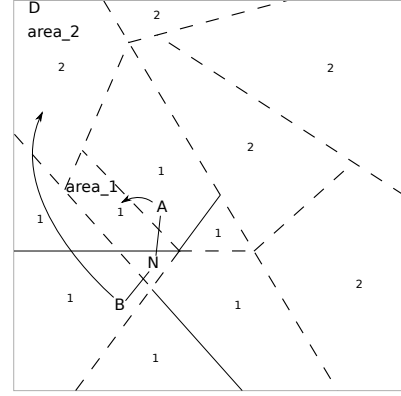


Fig. 5. Example of forwarding

- $\{B, C\}$: B is outside the local area but inside $area_1$. Thus, the routing table is updated with $\{Order-1\ aggr.,\ area_1,\ GW,\ C'\}$;
- $\{area_gw, C\}$: $area_gw$ is an order-1 aggregate area. At this distance, N doesn't retain any order-1 area but order-2. So $area_gw$ is aggregated into order-2 area that is $area_2$. Then the added entry is $\{Order-2\ aggr.,\ area_2,\ GW,\ C'\}$.

E. Forwarding

Forwarding decisions rely on information contained in the routing table.

When a node receives a packet, first it checks if it is the destination. If not, it has to choose a node to which the packet will be forwarded. This process is repeated by every node until the packet reaches the final destination.

We recall the assumption that the sender node is aware of the geographical location of the destination either inferred from its ID or from a location service [9]. Also, each aggregate area has a defined geographical location corresponding to that of its *reference point*. The reference point of an aggregate area can be defined as its geometric centroid.

We propose a greedy-like forwarding process. In pure greedy routing [4], the next hop for a forwarded packet is one of the one-hop neighbors nodes that is closer to the destination. In our protocol, the next hop is chosen as the gateway to the node or the area (regarding its reference point) that is closest to the destination. We use euclidean distances for proximity comparisons.

Algorithm. Let N a node receiving a packet whose final destination is D.

- Select all the gateways offering a node or an area closer to D:
 - If no gateway is found, forwarding fails;
 - Else, among the selected gateways, choose the one offering the node or the area closest to D.

Pure greedy routing fails when no neighbor is at a smaller distance to the destination (local maximum). In our solution, forwarding fails when all the areas reachable through the

gateways are at further distances from the destination than the source. But the failure rate is expected to be lower. There are two main reasons for that. First, the reachable areas are not restrained to the immediate neighborhood of a node. They can be located far from the farthest one-hop neighbors. Thus chances are high to find an area closer to the destination. Secondly, the one-hop broadcast and non geographical nature of the the updating mechanism allows the routing information to circumvent local maxima. Thus some forwarding candidates areas are made reachable even through gateways that are at greater distance from the destination than the source.

Example. On Fig.5, the node N has to forward a packet whose final destination is D. In its routing table it has the following entries among others:

- $\{Order-1\ aggr.,\ area_1,\ A,\ C\}$: A is the gateway to reach the area $area_1$;
- $\{Order-2,\ area_2,\ B,\ C\}$: B is the gateway to reach the area $area_2$.

Because $area_2$ is closer to D than $area_1$, the packet will be forwarded to B even if B is farther from the destination than N.

F. Discussion

In this section we evaluate qualitatively our solution by discussing its performance according to some metrics.

1) *Table size*: The routing table size (the number of entries inside tables) has a direct impact on the amount of memory consumed by every node in order to run the protocol properly. In classical table-based routing protocols, for a connected network, the table size is linear with the number of nodes in the network as ultimately each node will trigger an entry. Our proposition reduces the size of tables thanks to the concept of areas which aggregates node inside them. Furthermore, smaller tables leads to smaller route updates meaning reduced overhead.

2) *Convergence time*: Given a static network where all nodes start at the same time, the convergence time is the necessary amount of time for each node's routing table to stabilize. Compared to classical table-driven routing [10], our solution needs shorter time to converge. In fact, a routing

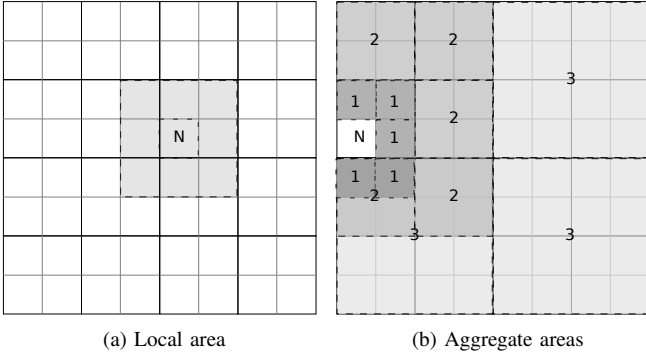


Fig. 6. Local area and aggregate areas

update leads to table modification in a limited range of nodes. As seen previously, when a node receives a route update, it looks for an aggregate area for the destination. The further the destination is, the higher the probability that the aggregate area is a big one. In this case, chances are high that a route with the same aggregate area has already been received, so no update occurs.

3) *Packet delivery ratio (PDR)*: In the case of static network, our proposition is expected to be a tradeoff between a full table-driven routing protocol and a geographical one. More precisely we are improving the table-driven aspect by reducing the table size and bandwidth consumption (as seen before) with the cost of lower PDR because of routing information aggregation. But the use of routing tables can circumvent some of the local maxima making our solution more efficient than pure geographical routing protocol.

4) *Mobility*: The study of mobility needs extensive simulation and is out of the scope of this paper. However we wanted to highlight a potential advantage of our solution. Thanks to the aggregate areas and routing tables having areas as entries, we tolerate some local mobility of distant nodes. In fact, for a given node, mobility of other nodes that are confined inside an aggregated area don't trigger any update. In other words, they mostly generate updates at a local level.

III. SIMULATION

A. Objectives

The main goals are to show that our proposition is more efficient compared to pure geographical routing while having good scaling capabilities in terms of routing table size and convergence time. We measured table size, convergence time and packet delivery ratio (PDR) in regard to nodes number and nodes density.

B. Local area and aggregate areas implementation

In order to partition the network zone, we chose the hierarchical area partitioning proposed in [11]. Their algorithm consists in a recursive subdivision of the whole network zone into square areas: 4 order- n areas form one order- $n + 1$ area. The size of the primary areas (order-1) is chosen such as two nodes inside the same order-1 area can reach each other.

The local area of a node is implemented as the geographical region formed by the order-1 aggregate area containing the node and all its adjacent order-1 areas (Fig.6a).

Order- n aggregate areas are implemented as order- n areas adjacent to the order- n area containing the node. Fig.6b shows examples of aggregate areas with their respective orders.

C. Simulation parameters

We conducted our simulation using *ns-3* (ns-3.24 release). Each node implements the 802.11b model with 2 Mbps rate and 250 m range. The nodes are randomly deployed in the network zone and remain static. They are powered on at the same time. For routing table size and convergence time measurements, we simulated 20 to 300 nodes with constant density of 1 node per 9000 m². For PDR scenario, we simulated 200 nodes with density varying from 0.25 node per 9000 m² to 1.25 nodes per 9000 m². The nodes are deployed in variable size square regions to reflect the density. 10 nodes are selected randomly to generate 2 kbps CBR flows with 64 bytes packets sent to other 10 random nodes. The flows are started after the convergence of the routing tables and last for 10 seconds. There is no loss due to interference between flows as they are not concurrent. We chose an interval of 15 s between routing periodic updates. A simulation ends 10 s after the last sent packet.

D. Results

1) *Routing table size*: The Fig.7a shows the average number of entries in the routing tables with different network sizes. As we can see, the average number of routing table entries for increasing nodes number has a logarithmic shape. This shows the scalability of our proposition as the average routing table size doesn't explode with large number of nodes in the network.

2) *Convergence time*: The convergence time is the necessary amount of time for all the nodes tables to stabilize after start-up for a static network. Before convergence, the tables have incomplete routing information resulting in poor forwarding decisions. Thus the convergence time has an impact on how fast a network can be deployed. It is interesting to see how this metric evolves with the size of the network as short deployment time even with high number of nodes can be considered as a criteria of scalability. The Fig.7b shows that the convergence time grows slowly with the number of deployed nodes.

3) *Packet delivery ratio (PDR)*: The PDR is the ratio between the number of packets successfully received by the destination and the total number of packets sent by the source. The Fig.7c shows the average PDR with different node densities. As we can see, with a large range of node densities, our proposition brings substantial improvement to the PDR compared to pure greedy routing. It demonstrates its capacity for circumventing the local maxima, reducing forwarding failures.

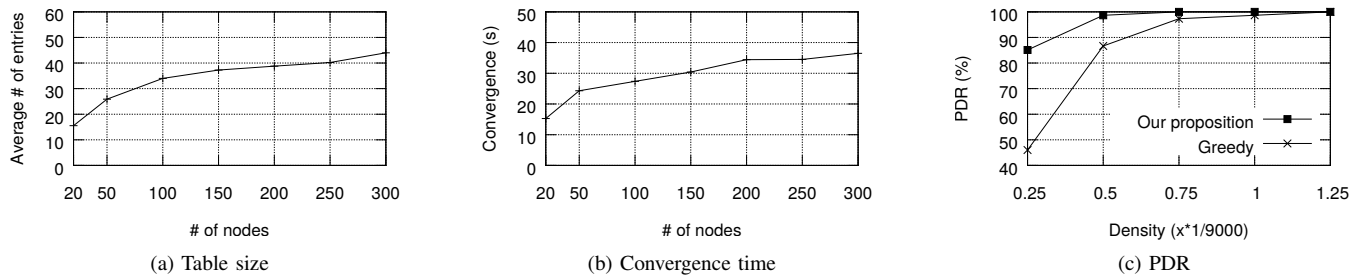


Fig. 7. Simulation results

IV. RELATED WORK

In [10] Mohseni et al. propose a comparative study of topology-based routing protocols. For the position-based approaches, a comprehensive survey can be found in [4]. To avoid local maximum handling, some geographical routing protocols construct landmarks [12] or spanning tree-like structure [13]. The idea of using areas for routing purpose has been already proposed. ZRP [14] defines two areas for applying respectively reactive and proactive topology-based routing protocol. Another protocol using areas not in a geographical way is DART [15] which uses its own addressing scheme to define logical hierarchical areas. In a more geographical fashion, [16] proposes region summaries of nodes with routing on them. Regarding variable routes details, FSR [17] takes into account the proximity of nodes to each other to adapt routing updates precision.

V. CONCLUSION

We propose in this paper, a geographical routing protocol provided with topological information. Every node maintains a routing table containing hosts entries for closer nodes and areas entries aggregating further nodes. To make it possible for the node to aggregate and exchange nodes information, the deployment area is splitted into partitions. Routing decisions are taken solely on routing table contents: a packet is sent to the gateway of entry nearest to the destination.

Evaluation compares the proposition performances with those of a pure geographical routing. It shows higher Packet delivery ratio validating that the proposed protocol makes a better use of link information by reducing forwarding failures. The size of the routing table and the convergence time increase slowly in logarithmic shape with the number of nodes, demonstrating a good scale factor stability.

The way our proposal introduces topological information in location-based routing, can be seen as widening the possibility for a node to choose the next hop. While on classical geographical routing, the next hop is a node, in our proposition, the next hop can be considered as an area: geographical routing is then extended to areas. Future work will study quantitatively the impact of mobility on protocol performance.

ACKNOWLEDGMENTS

This research was supported by Région Réunion and European Union (ERDF).

REFERENCES

- [1] F. Dressler, "Self-Organization in Ad Hoc Networks: Overview and Classification," University of Erlangen, Dept. of Computer Science 7, Technical Report 02/06, March 2006.
- [2] S. A. R. Zekavat, S. Kansal, and A. H. Levesque, "Wireless positioning systems: Operation, application, and comparison," in *Handbook of Position Location*. John Wiley & Sons, Inc., 2011, pp. 3–23.
- [3] D. Chen and P. K. Varshney, "A survey of void handling techniques for geographic routing in wireless networks," *IEEE Communications Surveys Tutorials*, vol. 9, no. 1, pp. 50–67, First 2007.
- [4] F. Cadger, K. Curran, J. Santos, and S. Moffett, "A survey of geographical routing in wireless ad-hoc networks," *IEEE Communications Surveys Tutorials*, vol. 15, no. 2, pp. 621–653, Second 2013.
- [5] M. Mauve, J. Widmer, and H. Hartenstein, "A survey on position-based routing in mobile ad hoc networks," *IEEE Network*, vol. 15, no. 6, pp. 30–39, Nov 2001.
- [6] L. Blažević, S. Giordano, and J.-Y. Le Boudec, "Self organized termination routing," *Cluster Computing*, vol. 5, no. 2, pp. 205–218, 2002.
- [7] D. D. Couto and R. Morris, "Location proxies and intermediate node forwarding for practical geographic forwarding," MIT Laboratory for Computer Science, Tech. Rep., June 2001.
- [8] C. E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," in *SIGCOMM '94 Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234–244.
- [9] S. Das, H. Pucha, and Y. Hu, "Performance comparison of scalable location services for geographic ad hoc routing," in *INFOCOM 2005 24th Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 2, March 2005, pp. 1228–1239.
- [10] S. Mohseni, R. Hassan, A. Patel, and R. Razali, "Comparative review study of reactive and proactive routing protocols in MANETs," in *4th IEEE International Conference on Digital Ecosystems and Technologies*, 2010, pp. 304–309.
- [11] M. N. Ranaivo Rakotondravelona, F. Harivelo, and P. Anelli, "Hierarchical area-based address autoconfiguration protocol for self-organized networks," in *Ad Hoc Networks - 7th International Conference, AdHoc-Nets 2015, Proceedings*, 2015, pp. 173–184.
- [12] J. Na and C. Kwon Kim, "GLR: A novel geographic routing scheme for large wireless ad hoc networks," *Computer Networks*, vol. 50, no. 17, pp. 3434 – 3448, 2006.
- [13] B. Leong, B. Liskov, and R. Morris, "Geographic routing without planarization," in *NSDI 2006 3rd Symposium on Networked Systems Design and Implementation*, vol. 6, 2006, p. 25.
- [14] B. Patel and S. Srivastava, "Performance analysis of zone routing protocols in mobile ad hoc networks," in *NCC 2010 National Conference on Communications*, Jan 2010, pp. 1–5.
- [15] J. Eriksson, M. Faloutsos, and S. Krishnamurthy, "DART: Dynamic address routing for scalable ad hoc and mesh networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 119–132, Feb. 2007.
- [16] P.-H. Hsiao, "Geographical region summary service for geographical routing," *SIGMOBILE Mobile Computing and Communications Review*, vol. 5, no. 4, pp. 25–39, Oct. 2001.
- [17] A. Iwata, C.-C. Chiang, G. Pei, M. Gerla, and T.-W. Chen, "Scalable routing strategies for ad hoc wireless networks," vol. 17, no. 8, pp. 1369–1379, 1999.