

Localized Load-aware Geographic Routing in Wireless Ad Hoc Networks

Xu Li, Nathalie Mitton, Amiya Nayak, Ivan Stojmenovic

► **To cite this version:**

Xu Li, Nathalie Mitton, Amiya Nayak, Ivan Stojmenovic. Localized Load-aware Geographic Routing in Wireless Ad Hoc Networks. International Conference on Communications - Wireless Networks Symposium -(IEEE ICC-WN 2012), Jun 2012, Ottawa, Canada. hal-00658270

HAL Id: hal-00658270

<https://hal.inria.fr/hal-00658270>

Submitted on 11 Jul 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Localized Load Balancing for Geographic Routing in Wireless Ad Hoc Networks

Xu Li*, Nathalie Mitton*, Amiya Nayak[†], and Ivan Stojmenovic[†]

*INRIA Lille - Nord Europe, Univ Lille Nord de France, USTL, CNRS UMR 8022, LIFL, France

[†]School of Information Technology and Engineering, University of Ottawa, Canada

Email: {xu.li, nathalie.mitton}@inria.fr, {anayak, ivan}@site.uottawa.ca

Abstract—In this paper, we propose to apply the concept of cost to progress ratio (CPR) in geographic routing for load balancing. The resultant routing protocol is named CPR-routing. It is a localized parameterless approach optimizing both geographic progress and nodal load during hop selection. Through extensive simulation, we compare it with an existing parameter-based localized solution α -routing. Our simulation results indicate that CPR-routing outperforms α -routing in both load balancing and packet delivery ratio.

Keywords—Load balancing, geographic routing, localized algorithms, wireless ad hoc networks

I. INTRODUCTION

Wireless ad hoc networks are collections of wireless nodes (usually battery powered) that communicate with one and another using radio frequency without a fixed infrastructure or centralized control. They are widely reorganized to have great potentials in a wide range of applications in disaster relief, conference, and battlefield environments, for example. One instance of such networks is wireless sensor networks. These networks as a whole improve the accuracy of information or computation provided by individual computationally weak sensors, greatly extend our ability to monitor and control the physical world, and hold the promise to enable an information-rich ubiquitous computing environment. In wireless ad hoc networks, a message sent by a node is received by all nodes located within the node's transmission range. Because the limited transmission range, two communicating nodes that are far apart from each other have to rely on some intermediate nodes relaying their messages. Under this circumstance, routing is a primary challenge.

Geographic routing employs non-flooding based route discovery and offers scalability [16]. It has been studied for years in wireless ad hoc networks. It requires that nodes know their own positions. This requirement may not be a strong assumption because in some cases nodal location information is readily available. For example, in most wireless sensor networks, nodes know their location for providing meaningful sensory data and supporting proper event response. A deep discuss about the rationality of location awareness can be found in [16]. Location awareness may be achieved by attaching each sensor a GPS device or running some non-GPS based localization algorithm when GPS is not available. In addition to their own location, nodes also need to know the position of the destination in order to engage geographic routing. When

the destination is fixed, the information can be preprogrammed in the nodes; when the network size is small, it may be dynamically obtained by a broadcast task. In the general case that the destination can be any node and the network size is large, a location service algorithm [11] is needed.

Routing must be done with caution in order not to overload resource-constrained wireless nodes. Load can be defined differently [19], for example as energy consumption, delay, traffic size, link quality, activity, etc. or the combination of them, depending on the resource scarcities and/or the optimization objectives. Load balanced routing for wireless ad hoc networks has been studied in literature. Each node may estimate its own load and attach the information to the control packets during route discovery, and destination is then able to select the route with least load overall [6,9] when multiple paths are discovered. Alternatively, each intermediate node may simply ignore a routing message if it considers itself overloaded [2,22] (with respect to a load threshold or priorities, for example). Multi-path routing has also been suggested for load balancing purpose [14,21], but its effect is limited unless there is a very large number of paths available [5]. It is recently proposed as a load balancing technique to map the 2D network to a 3D sphere and route along the surface of the sphere [12,15]. The logic is that the 3D sphere offers larger space stretch and thus more possibilities to spread load. This method however requires a pre-defined projection function for the mapping.

In this paper we propose to employ the cost over progress ratio (CPR) concept [17] to achieve load balancing in geographic routing. We combine CPR and distance-based greedy forwarding [4] to obtain a simple localized load-aware routing protocols, named CPR-routing. It optimizes both geographic progress (toward the destination) and nodal load during hop selection, without requiring any input tuning parameter. CPR-routing can replace the greedy routing part of complex greedy-face combined routing protocols [1,8] (known for guaranteed delivery) and equip them with load balancing capabilities. Extensive simulation is conducted to evaluate its performance in comparison with an existing parameter-based load-aware routing protocol [20], referred to as α -routing by us in this paper. Simulation results indicate that CPR-routing outperforms α -routing in terms of both load balancing and delivery ratio. In the following, we present α -routing and CPR-routing in Section II and report our simulation study in Section III. We conclude the paper in Section IV.

II. LOCALIZED LOAD-AWARE GEOGRAPHIC ROUTING

In distance-based greedy forwarding [4], the node C that is currently holding a packet picks the neighbor with the shortest distance to the destination D among all neighbors that are closer to D than itself as the next forwarding node. If no neighbor is closer to D than C , the packet reaches a dead end, and greedy routing fails. Greedy failure may be recovered by FACE routing technique [1,8]. We see that distance is the only metric for next hop selection in greedy forwarding. The logic of a load balancing technique for geographic routing is to use certain utility function of both distance and load to guide next hop selection so as to ensure progress (toward the destination) and avoid routing through loaded areas. The technique will reduce the risk that some nodes reach their load capacity limits soon and finally drop packets. To circumambulate heavily loaded area, nodes have to know the latest load information of itself and the neighbors around it. They are able to keep track of their own load by counting their computation and communication activities such as sending, receiving and processing. Neighboring nodes obtain each other latest load information by periodic hello message exchanges [10].

Ideally, a load balancing technique is able to select a neighbor with minimal load and maximal progress to the destination. However, the two factors cannot be minimized simultaneously in most cases. The goal is thus to seek solutions to balance the contribution of load and geographic progress to routing decision. In the following, we first introduce an existing protocol α -routing that relies on an input parameter α for making hop selection decisions. After that, we propose CPR-routing.

A. Definition of Load

Generally speaking, load implies the percentage of time in which a node is busy from certain concerned perspectives. It is then given a specific definition in accordance with the resource scarcities. For instance, it can be reflected by the CPU usage when computation is a major concern, or available storage space when memory is limited. Here, we refer to load as the energy consumption caused by wireless communications. Different from other resources, energy can not be recovered/released once occupied.

In [20], load is defined as the percentage of time that a node is busy by transmitting, receiving, or overhearing a transmission from a neighboring node. Each node scans the wireless medium at regular intervals and calculates its current load by summing up all measured medium states (busy or idle) and averaging them over a certain period of time. Load balancing therefore aims to avoid generating communication hot spots in the network. This load definition does not distinguish the energy consumption by transmitting and receiving, and thus will not lead to balanced power usage among nodes.

Hence, in this paper we define load in a different way. Assume that packets have constant size and nodes transmit packets using maximum power. Each node C has a load limit M , representing the maximum energy level that can be consumed. Initially, its load is 0. Whenever a packet is

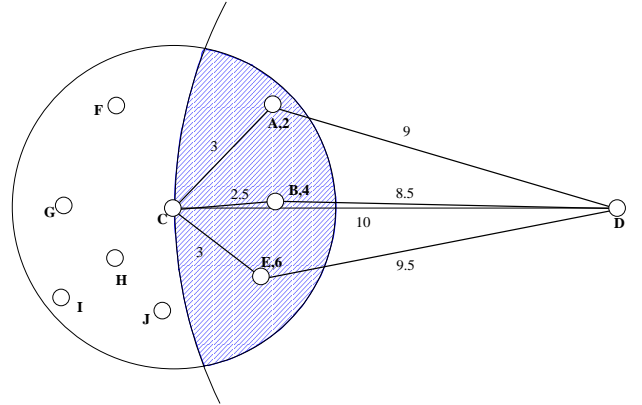


Fig. 1. Average hop count.

transmitted from C to a neighbor A , load is increased by a units (transmission power usage) for C and by b units (reception power usage) for all nodes neighboring C .

B. α -routing

Transier et al. [20] proposed a dynamic load balancing method, which is referred to as α -based technique by us here in this paper. In this method, routing next hop is selected based on an objective function:

$$F_{\alpha}(A) = \alpha \text{Load}(A) + (1 - \alpha) \frac{|AD|}{|CD|},$$

where α is a pre-defined system parameter in range $[0, 1]$, D the destination, C the current node and A a node neighboring C . The neighbor A that minimizes the objective function is chosen as next hop.

When $\alpha = 0$, this method is equivalent to greedy routing. For $\alpha = 1$, the routing decision completely relies on load, while distance is considered only for pruning neighbors farther from the destination than current node. In this case, packets may be forwarded to certain neighbor closer to the destination, thus there is still chance to succeed in packet delivery. The authors suggested that an α value between 0.2 and 0.5 would lead to a good tradeoff between delivery ratio and load balancing.

C. CPR-routing

We proposed a localized Cost over Progress Ratio (CPR) framework [17] for designing network layer protocols. This framework may be applied in geographic routing to optimize cost as follows. Suppose that each edge has a cost measure. Node C , currently holding the packet, will forward it to neighbor A , closer to destination D than itself, which minimizes the objective function (the ratio of cost over progress):

$$F_{CPR}(A) = \frac{\text{Cost}(CA)}{|CD| - |AD|}.$$

While progress clearly measures the advance of A toward D in the CPR framework, the cost measure can be hop count (from A to D), power, reluctance, power_reluctance, delay,

and expected hop count, etc., depending on the assumptions and metrics used. In the context of load balancing, we have $Cost(CA) = Load(A)$. The CPR framework well generalizes different optimization objectives and provides a uniform solution to them. However, due to its generalized nature and wide range of application possibilities, its performance is not well evaluated. In particular, there is no performance evaluation in load balancing setting.

III. PERFORMANCE EVALUATION

In this section, we evaluate CPR-routing and α -routing through extensive simulation. Because we are interested in load-aware routing on the networking layer, we ignore the complexity of the lower layers and conduct simplified, yet effective, simulations. Unlike traditional simulation study, our simulation places the emphasis on the average node degree (i.e., the average number of neighbors of each node) as the independent variable rather than the transmission radius or network size. This allows us to evaluate the performance for sparse, medium degree and dense networks with clear and precise values of the independent variable. We choose to use the following performance metrics:

- Average load per node: the average hop of each node at certain time instant.
- Average hop count: the average number of hops along each established route.
- Overall Success Rate: the ratio of packets delivered to packets sent.
- Non-greedy Success Rate: the ratio of packets delivered to packets sent and not dropped due to greedy failure but because load limit has been reached.

A. Simulation setup

We implemented the greedy routing protocol [1] with the two load balancing techniques using a custom C simulator. We considered a wireless ad hoc network randomly deployed in a 100×100 region, where each node has a load limit 100. We fix the network size $N = 100$ and vary the average node degree d from 6 to 99. For each d , we generate 300 routing tasks between randomly selected nodes and run the plain greedy routing protocol and the two versions of load-aware greedy routing protocols over 30 randomly generated connected unit disk graphs (UDGs) to get average simulation results. Note that for α -routing, we only present the simulation results with $\alpha = 0.2$. This is because when $\alpha = 0.2$, the routing protocols achieve better or similar performance compared with other α values, as discussed in [20] and confirmed by our own simulation in Figure 2.

1) *Load increment policy*: The policy can be defined according to the nature of the work load. Different definitions will result in different load distribution among the network nodes. Here, we aim to use load to reflect energy consumption. Based on the statistics on real-life sensors, the energy consumption of transmitting is about twice as that of receiving. Therefore, we set $a = 2$ and $b = 1$. We are interesting in the

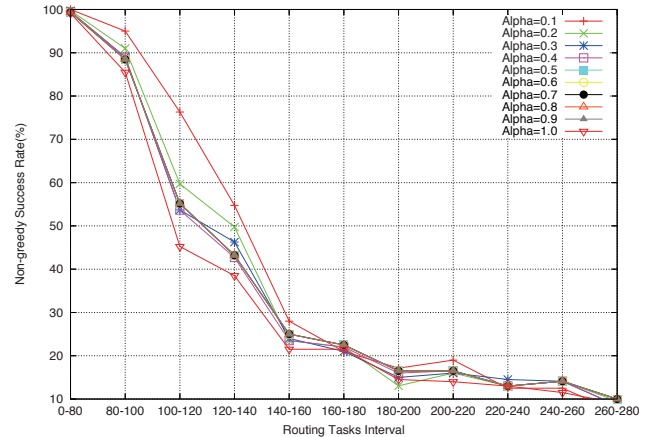


Fig. 2. α -routing with different α values.

performance difference of α -routing and CPR-routing under this load increment policy.

2) *Random connected UDGs*: One way to construct random connected UDGs is introduced in [18]. This method first places all the N nodes randomly in the deployment field and then select a proper communication range to satisfy the degree requirement. It is the most unbiased way of generating random connected UDGs in that it chooses any random connected UDG among all possible random connected UDGs having the desired N and d with equal probability. We observed that in most cases the resultant UDGs are not evenly distributed in the deployment field. Such graphs may contain connectivity-critical nodes that have to be always used in routing and result in inevitable unbalanced load distribution.

To minimize measurement noise and clearly show the behavior of the load balancing techniques, evenly distributed UDGs are desired in our simulation study. Therefore, we choose to use the more advanced UDG generate method, named MAX-DPA [13]. The basic idea is to place node incrementally in the deployment field and impose additional restrictions during the node placement in order to improve network connectivity and functionality. Two main restrictions are: (1) the distance between any pairs of nodes has to be greater than d_0 , and (2) the degree of any existing node has to be less than $d_{max} + d$, where d is the required average node degree of the graph.

By MAX-DPA, a large-valued d_0 and a small-valued d_{max} will lead to a more evenly distributed graph than a small-value d_0 and a large-value d_{max} . So, these two parameters need to be set carefully. In our simulation, we set $d_0 = 5$ and $d_{max} = 3$.

B. Simulation results

Figure 3 illustrate the average load per node in relation with node degree d . The results are taken at the moment that the 100-th routing task is initiated. We see that it goes up as d increases. This phenomenon is reasonable. Nodes obtain additional load by overhearing transmissions from neighbors. When node degree is high, nodes have more neighbors, and routing therefore impose load on more nodes and increase the

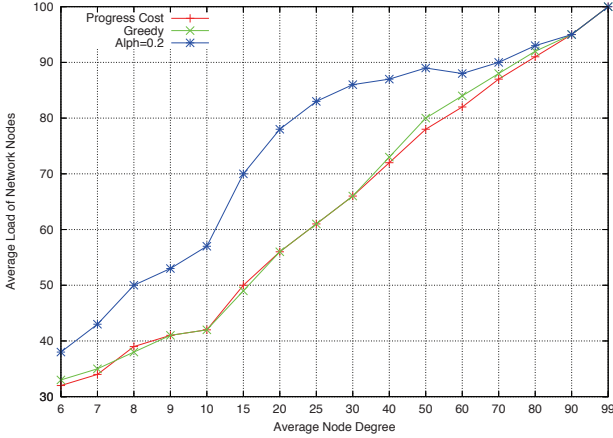


Fig. 3. Per node load at the moment of the 100-th routing task being initiated.

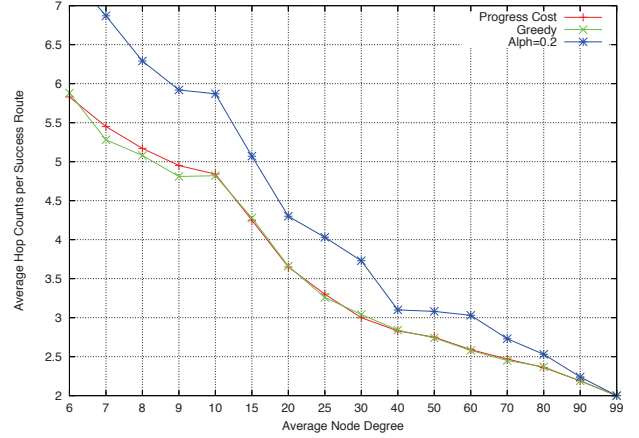


Fig. 5. Average hop count.

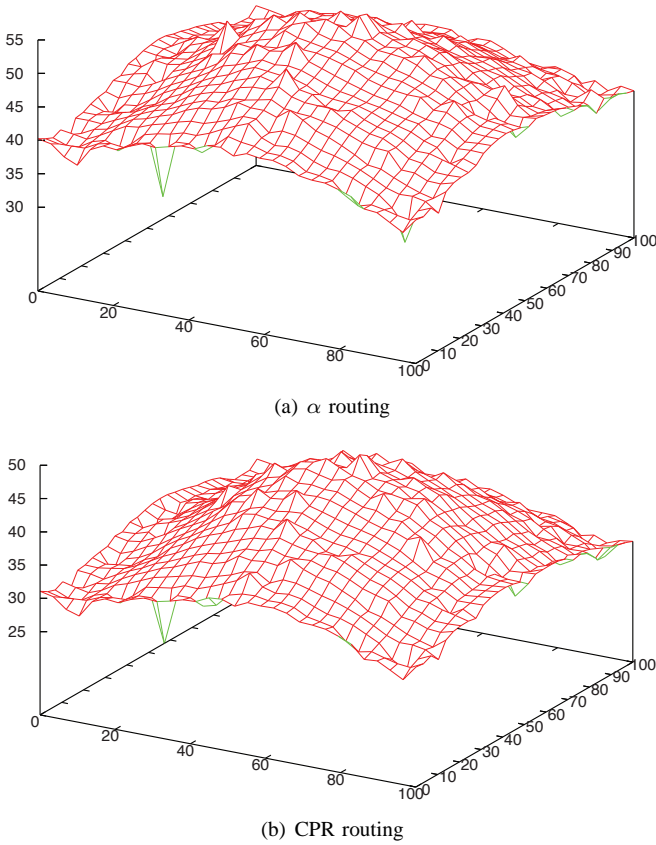


Fig. 4. Load distribution when the 100-th routing task is initiated ($d = 7$).

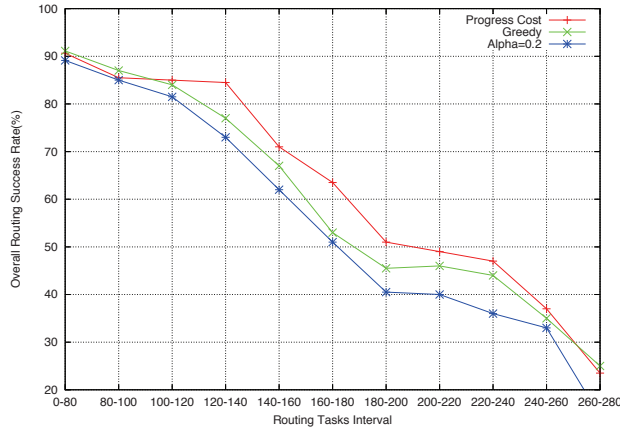
average per node load as a result. We also observe that the CPR routing has comparable per node energy consumption as greedy routing, and exhibit better performance than α -routing. Figure 4 shows the load distribution throughout the network with $d = 7$ at the moment that the 100-th routing task is generated. Compared with greedy routing, both α routing and CPR routing result in a more even load distribution throughout the the network.

Figure 5 depicts how average hop count is impacted by node

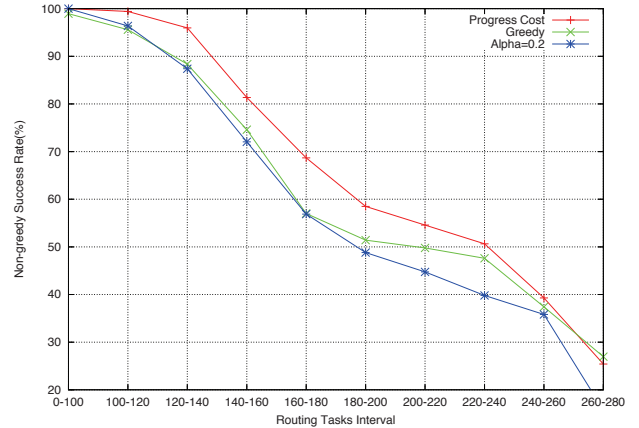
degree d . As a general rule, the smaller the network degree, the more difficult to construct straight line path, and the longer the routes eventually built. We observe that in the same setting, the length of the routes constructed by CPR routing is comparable to that by greedy routing, and shorter than that of α routing. In the sequel, we will study the packet delivery rate of the protocols. Relevant simulation results for $d = 7, 15$ are plotted in Figure 6; results for other d values are consistent them and thus omitted here for space limit.

Observe Figure 6. In all the three protocols, SR (whether overall or Non-greedy) drops as the number of routing tasks initiated increases. The phenomenon can be easily explained by the ever increasing loads on each node that eventually reaches the limit. When a node has maximum load, it will not participate any routing process. A similar decreasing trend of SR appears when the average node degree d . This is because the greater the network degree, the denser the network, and the faster the load increases on individual nodes (due to the overhearing more transmissions). According to our simulation, in the extreme case of $d = 99$ (roughly a complete graph), each node's load increase at least by 1 during a routing process, and SR drops from 0 from 100 when the number of tasks initiated is about $100 \sim 120$.

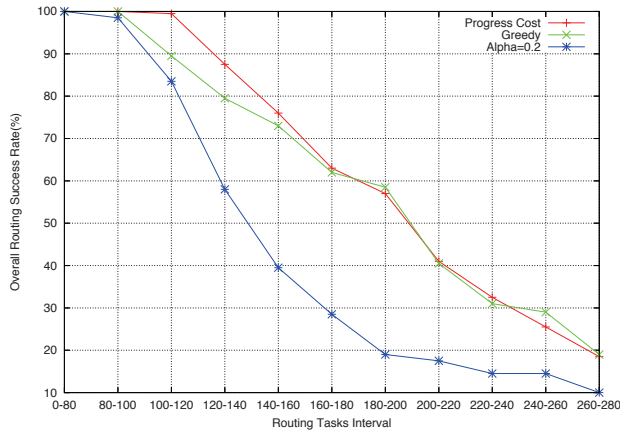
One promising observation from our simulation is that CPR routing outperforms both greedy routing and α routing in almost all the scenarios. α routing ranks below greedy routing. However, when $d > 70$, the three routing produce more and more similar results. When $d = 99$, their results are exactly the same. Another interesting phenomenon is that: when $6 \leq d \leq 10$, Overall SR is not as high as Non-greedy SR; when $d \geq 10$, overall SR tends to Non-greedy SR and eventually becomes the same as Non-greedy SR after d reaches 30. This is because that greedy routing fails very often when d is smaller (in sparse networks), which drags down the overall SR. When network d approaches 30, greedy failure no longer appears or appear very rarely, and consequently overall SR and Non-greedy SR are almost the same.



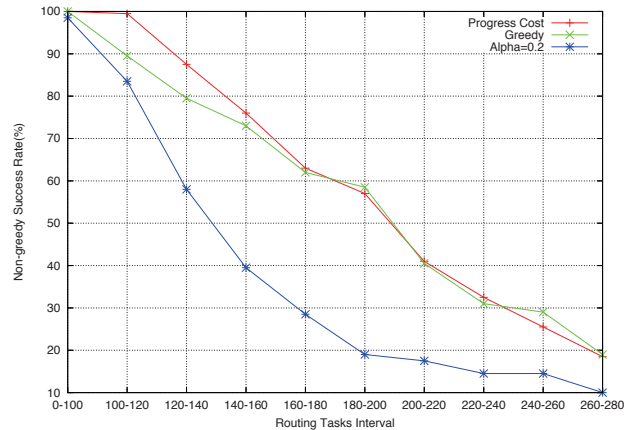
(a) Overall SR ($d = 7$)



(b) Non-greedy SR ($d = 7$)



(c) Overall SR ($d = 15$)



(d) Non-greedy SR ($d = 15$)

Fig. 6. Success rate (SR)

IV. CONCLUSIONS

In this paper, we have proposed to apply the Cost to Progress Ratio concept in greedy routing by considering load as cost and achieve load-aware feature. We compared the resultant CPR routing with the existing α routing through extensive simulation. Simulation results are encouraging. They imply that CPR routing is able to save more energy and build better routes than α routing. In the future, we will study more complex definition of cost to improve the protocol's performance. For example, it can be defined as $Nbr(A) * Load(A)$, where $Nbr(A)$ is the number of nodes neighboring A . This definition will favor nodes with less load and less neighbors. Further, the two contributing factors could be given different powers and show different importance. By tuning the powers, we will have fine control over the routing process.

ACKNOWLEDGEMENT

Part of this work was supported by CPER Nord-Pas-de-Calais/FEDER Campus Intelligence Ambiante.

REFERENCES

- [1] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks," *Wireless Networks*, vol. 7, no. 6, pp. 609-616, 2001.
- [2] M. Chen, V.C.M. Leung, S. Mao, and T. Kwon, "Receiver-oriented Load-balancing and Reliable Routing in Wireless Sensor Networks," *Wireless Communications and Mobile Computing*, vol. 9, no. 3, 2009
- [3] C. Diallo, M. Marot, and M. Becker, "Link Quality and Local Load Balancing Routing Mechanisms in Wireless Sensor Networks," In *Proc. of AICT*, pp. 306-315, 2010.
- [4] G.G. Finn, "Routing and addressing problems in large metropolitan-scale internetworks," Technical Report ISI/RR-87-180, Information Sciences Institute (ISI), 1987.
- [5] Y. Ganjali and A. Keshavarzian, "Load balancing in ad hoc networks: single-path routing vs. multi-path routing," In *Proc. of IEEE INFOCOM*, 2004.
- [6] H. Hassanein and A. Zhou, "Routing with load balancing in wireless ad hoc networks," In *Proc. of ACM MSWIM*, 2001.
- [7] M.E. Hesari, L. Narayanan, and J. Opatry, "New Routing Algorithms to Balance Traffic Load," In *Proc. of IEEE WCNC*, 2011.
- [8] F. Kuhn, R. Wattenhofer, and A. Zollinger, "Worst-case optimal and average-case efficient geometric ad-hoc routing," In *Proc. of ACM MobiHoc*, 2003.
- [9] S.-J. Lee and M. Gerla, "Dynamic load-aware routing in ad hoc networks," In *Proc. of IEEE ICC*, 2001.
- [10] X. Li, N. Mitton, and D. Simplot-Ryl, "Mobility Prediction Based Neighborhood Discovery for Mobile Ad Hoc Networks," In *Proc. of IFIP NETWORKING, LNCS 5793*, pp. 138-151, 2011.

- [11] X. Li, A. Nayak, and I. Stojmenovic, "Location Service in Sensor and Mobile Actuator Networks," *Wireless Sensor and Actuator Networks: Algorithms and Protocols for Scalable Coordination and Data Communication*, Wiley, 2010.
- [12] F. Li and Y. Wang, "Circular Sailing Routing for Wireless Networks," In *Proc. of IEEE INFOCOM*, 2008.
- [13] F.A. Onat, I. Stojmenovic, and H. Yanikomeroglu, "Generating random graphs for the simulation of wireless ad hoc, actuator, sensor, and internet networks," *Pervasive and Mobile Computing*, 4(5): pp. 597-615, 2008.
- [14] M.R. Pearlman, Z.J. Haas, P. Sholander, Z.J. Haas Peter Shol, and S.S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," In *Proc. of ACM MobiHoc*, 2000.
- [15] L. Popa, A. Rostamizadeh, R.M. Karp, C.H. Papadimitriou, and I. Stoica, "Balancing traffic load in wireless networks with curveball routing," In *Proc. of ACM MobiHoc*, 2007.
- [16] I. Stojmenovic, "Position Based Routing in Ad Hoc Networks," *IEEE Communications Magazine*, vol. 40, no. 7, pp. 128-134, 2002.
- [17] I. Stojmenovic, "Localized network layer protocols in sensor networks based on optimizing cost over progress ratio," *IEEE Network*, vol. 20, no. 1, pp. 21-27, 2006.
- [18] I. Stojmenovic and X. Lin, "Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks," *IEEE Trans. Parallel and Distributed Systems*, vol. 12, no. 10, pp. 1023-1032, 2001.
- [19] C.K. Toh, A.-N. Le, and Y.-Z. Cho, "Load Balanced Routing Protocols for Ad Hoc Mobile Wireless Networks," *Wireless Communications Magazine*, vol. 47, no. 8, pp. 78-84, 2009.
- [20] M. Transier, H. Fubler, M. Mauve, J. Widmer, W. Effelsberg, "Dynamic load balancing for position-based routing," In *Proc. of ACM CoNEXT*, 2005.
- [21] Z. Wang, E. Bulut, and B.K. Szymanski, "Energy Efficient Collision Aware Multipath Routing for Wireless Sensor Networks," In *Proc. of IEEE ICC*, 2009.
- [22] Y. Yoo and S. Ahn, "A simple load-balancing approach in cheat-proof ad hoc networks," In *Proc. of IEEE GLOBECOM*, 2004.