Impact of Mobility on User-centric Routing

Namusale Chama, Rute C. Sofia
SITI/University Lusofona, Lisbon
Email: {namusale.chama, rute.sofia}@ulusofona.pt

Susana Sargento
University of Aveiro, Aveiro
Email: susana@ua.pt

***Abstract*—This paper presents a summary of work aimed at making current multihop routing in mobile ad hoc networks more sensitive to node mobility and ultimately improve routing performance in the face of node mobility. We discuss the various existing mobility tracking parameters, how they capture node mobility and also their shortfalls. We propose some mobility tracking schemes(heuristics) based on one of the mobility tracking parameters(Link Duration) and also show the performance of one of the proposed heuristics.**

***Index Terms*—multihop routing, link duration, node mobility**

I. Introduction

The most recent paradigms in wireless architectures describe environments where nodes present a somewhat dynamic behavior (e.g. *Mobile Ad-hoc Networks*, *MANETS*) or even a highly dynamic behavior (e.g., *User-provided Networks*, *UPNs*). These nodes are wireless devices that are carried by humans and hence the nodes portray human mobility patterns in their respective networks. Studies have shown that human mobility is characterized by short but frequent movements[6], hence the resultant network will bear highly mobile nodes too. In such environments, due to limited node transmission ranges, data transmission is based on multihop routing i.e., based on *single-source shortest-path* approaches. This implies that the most popular multihop routing approaches normally rely on static link cost metrics such as hop count. The result is that when facing movement of nodes, multihop routing has its own shortfalls e.g., the need to recompute paths frequently if nodes exhibit high variability in movement of the nodes. In other words, current multihop approaches lack sensitivity in what concerns nodes movement. The cost function of single shortest-path approaches does not take into consideration metrics that automatically adjust to node speed, movement pattern, acceleration, or even direction. Hence, a topology change is merely interpreted as a trigger to perform path re computation. There are, however, cases where node movement may actually not represent a link breakage. Or, instead a link change due to node movement may be so subtle that in fact it would not require any update to the topology, from a routing perspective[3].

Mobility of nodes affects the performance of routing protocols in any type of mobile network. Such performance is affected in several ways. To give a concrete example, when adjacent nodes move, they affect some properties of the link, either the link becomes longer but does not break or it becomes longer and breaks. Figure 1 illustrates node mobility and and how routing responds to different types of node movements. A node that is on a routing path may be making confined movements around its space as illustrated in Figure 1(i). Depending on whether a link between node A and B is short or long, how a routing protocol performance will be affected differs. In case of a short link, displacement that causes link elongation will reduce the link capacity and may not lead to a link break. On the other hand, if it where a long link, the same displacement would lead to a link break and ultimately, path re computations. Another type of node mobility is when a node moves to and fro about a particular position as shown in Figure 1(ii). Such kind of mobility will lead to continuous link breaks when node B moves away with an alternative path A-D-E-C being used. Because node B lies on the shortest path, once it returns to its original position, the path A-B-C will be used. With such kind of mobility, continuous path re computations will occur and weigh down the performance of the routing protocol in form of unnecessary control overhead generation during path re computations. It may also happen that node B moves away and never returns as hown in figure 1(iii), a path recomputation in this case is justified if the link breaks.

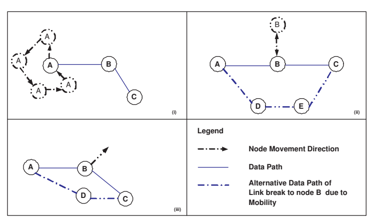

Figure 1. Node Mobility

On another note, and giving the example of hop count as the routing metric applied, when two nodes move apart and despite the fact that the link capacity may change, a routing protocol may be agnostic to such change. Therefore, depending on the operation of the routing protocol and algorithm followed, the performance of the network may be significantly affected and for different reasons, as protocol react to link breakages and perform route discovery differently.

The remaining sections are organised as follows; section II covers related work, section III provides a discussion on mobility tracking parameters. Link Durations Heuristics are covered in section IV. Preliminary results are in section V and we conclude in section VI.

II. Related Work

A number of approaches have been undertaken to detect and/or counter the effects of node mobility in routing.

978-1-4577-1394-1/11/$26.00 ©2011 IEEE

The first category comprises of approaches proposed to find stable links at route discovery and/or route maintenance phases. Bilgin, Khan and Al-Fuqada, on *Ad hoc On-Demand Distance Vector* (AODV), developed a mechanism that detects weaker links by checking a received packet and the weak links are replaced by alternative paths through path recomputations[2]. Hu, Wang and Wang have employed *Link Expiration Time*(LET) to discover and maintain stable links. If the route LET falls below a specified threshold, path recomputation is triggered[4]. On the other hand, Li, Mitton and Simplot-Ryl have used mobility prediction to acquire up-to-date network topology. Based on node location and direction changes, the next node location is predicted. If the prediction is wrong, a HELLO message is sent to communicate of the new location[5]. Node mobility that may cause weak links and shorter LET may be temporal as is the case in confined or repetitive motions represented in figure1(i) and 1(ii) respectively. Paths recomputations are unnecessary for a link that will form shortly or may be even break at all. While mobility prediction is a good feature, in this case, predictions based on node speed changes and direction changes only, do not capture mobility patterns which are important. If a link is known to reform after a short while or node displacement is temporal, path recomputations or location change notications to other nodes are not required.

A second category of work that tries to make routing more sensitive to node mobility relates to throughput variation measurement as a way to determine node mobility. For instance, Suyang and Evans have used the slope of change of throughput in a link vs. the link load to estimate topology changes [7]. Based on history, through throughput monitoring, a decrease, according to Suyang and Evans, means that there is a true change in the physical topology. They have attributed the changes to increase in the node distance and increase in interference. Even with good attributes of avoiding interference and detecting mobility collectively, node mobility individually has attributes that have not been addressed. Nodes may exhibit movement that does not necessarily impact the route stability.

Also Bilgin, Khan and Al-Fuqaha proposed on AODV a "shrink" mechanism where optimal paths are discovered during route maintenance phase through a shrink packet that propagates a route[1]. While route optimality is a good feature, a short lived optimal path is a cost to routing as path recomputations will occur when the route breaks and routing performance is affected. If mobility pattern was taken into consideration , then discrete optimal paths would improve routing in AODV.

All of the mentioned approaches have in common the aspect that they fail in being able to distinguish between a topology change that is long-lasting, and a topology change that is so short that in fact it should not result in route re-computation. Our belief is that by defining a multihop routing metric more sensitive to node mobility, multihop routing can become more robust and better adjusted to the current wireless dynamic scenarios.

III. Mobility Tracking Parameters

A number of mobility tracking parameters have been devised to detect node mobility. Some of the mobility tracking parameters are *link duration*, *pause time, average number of link breaks(ALB)*and *node degree stability (NDS).*

A. Pause Time

Pause Time the period of time that the node is stationary. As a mobility parameter, a link whose node(s) have low pause time is considered not stable and is avoided in routing. *Pause Time* captures link stability only among stationary nodes, but link stability that exists among mobile nodes due to spacial correlation is not captured.

B. Average Number of Link Breaks

This is the average number of link breaks a node incurs in a specified period of time. A node with high number of link breaks is undesired for routing. As a mobility parameter, link breaks that result from repetitive motions are not distinguished and results alsso in path recomputation for the affected route even as the link break is a temporal one.

C. Link Duration

Link duration (LD) or *lifetime* is a parameter that is tightly related to the movement of nodes and is also as of today one of the parameters that is most popular in terms of tracking node mobility. However, the current LD does not capture the case where a node jumps between its original position and a second position with a frequency that is not significant in terms of the potential delay it causes. Such movement will trigger repeated recomputation, which brings in more delay than if such frequent hopping would simply be disregarded. In as far as mobility patterns are concerned, LD captures link stability of nodes that do not reach their link break threshold. However, cannot distinguish between a temporal and permanent link break.

IV. Link Duration Heuristics

From the study of mobility tracking parameters[3], *Link Duration* is a parameter that is more sensitive to mobility than others and we have devised heuristics based on link duration to better capture node mobility where native Link Duration fails. Three heuristics have been developed.

A. Relaxing the Link Duration Definition: LD with a Tolerance Interval

The notion behind this heuristic is that when a link breaks, we wait for a period of t seconds before triggering path re-computation. If a link re-forms within the specified period t, then path re-computation will be avoided and no control overhead would result as such. On the other hand, if the link does not re-form, then some delay will be introduced. Our belief is that the metric would perform well in environments where repetitive motions is highly prevalent.

B. Spacial Stability-based Link Duration

The heuristic is a combination of *node degree*(NDS) and *average number of link breaks*(ALB) parameters. To provide an example of how this can be achieved, we define *Spatial Stability-based Link Duration (SS LD)* as an extension of LD based on a correlation between ALB and NDS. It should be noticed that the work here provided is intended to be initial and hence we do not provide a concrete instantiation of a formula that may represent SSLD. Instead, we explain the rationale for this heuristic.

Let us consider a node i that has a high NDS and a high ALB. In terms of node mobility and its impact on routing, links related to such node are expected to be less robust, given that there is a strong movement associated to the perspective of such node. Hence what SSLD can provide is a way to, at an early instant in time, discard successors of a node because they exhibit some mobility variability. As mentioned before, the node may be static and yet, the result are less stable paths (because most neighbors may be moving). These aspects have been previously debated[3].

C. Promoting More Robust Links: Link Duration Prime(LDP)

In this metric, we propose to compute a link robustness weight that is inversely proportional to the number of times that such link breaks, based on a periodical check up. The line of thought for this third heuristic is to consider that while a link is stable, no penalty is provided to the link. Based on a time-window mechanism (variable or fix) and upon a verification that the link is stable, then the link robustness weight is incremented. We highlight that at this stage such weight is simply based on increments; more sophisticated formulas for the weight computation are expected to be considered in the future.

For a link i the robustness weight r_i is incremented each time there is a verification for the link status, and the link is considered to be active. This is however, a deterministic behavior which we intend to further improve. Hence, we opted to consider a simplistic approach assuming that a link, upon verification, is broken.

LDP (and its variations) is expected to perform the best upon the occurrence of motion that exhibits some form of repetitive pattern. On the other hand, in case of non- repetitive motions, some delay is expected.

V. Preliminary Results

The link duration heuristics that have been proposed are to be implemented in AODV and *Optimized Link State Routing*(OLSR) in NS2. The third heuristic(Link Duration Prime) has been implemented in OLSR and we give details of the simulation environment. To obtain preliminary results, a 100 second simulation was performed, using Random Waypoint mobility model for 20 nodes with maximum speed of 10m/s in an area of 600 m by 600m squared. In comparison to its performance, is shortest path OLSR and link quality based OLSR (ETX).

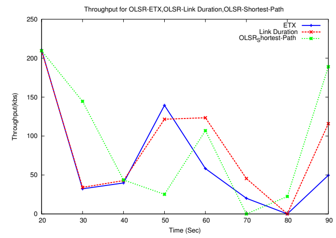

Figure 2. Throughput for different OLSR flavors

Figure 2 shows the performance of the different flavors of OLSR including link duration heuristic proposed. From the graph, we can deduce that OLSR-ETX does not perform well in mobile environment while the link duration heuristic in form of OLSR-Link Duration outperformed shortest path in some cases. This is to say that some element of spacial correlation do exist in the topology and was captured OLSR-LD and that shortest path was employing unstable links. On the other hand, OLSR-shortest path also did outperform OLSR-LD too in some cases, this is to say that stable links that were used by OLSR-LD were long links.

VI. Conclusions

We have discussed node mobility sensitivity and how it impacts multihop routing. We have also discussed works that have been proposed to capture node mobility in form of mobility tracking parameters and how they do not fully capture mobility. We did propose some three heuristics which are aimed at improving sensitivity to mobility. One of the heuristic has been tested in OLSR and not fully outperformed shortest path OLSR but in some instances it has. Our belief is that with further fine tuning of the metric in the time window,it would perform better in all mobility circumstances. As, for the future works, we intend to implement the remaining heuristics also and continue fine tuning the one that has already been implemented.

References

[1] Zeki Bilgin, Bilal Khan, and Ala Al-Fuqaha. Only the short die old: route optimization in manets by dynamic subconnection shrinking. In *Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, IWCMC '10, pages 681–685, New York, NY, USA, 2010. ACM.

[2] Zeki Bilgin, Bilal Khan, and Ala I. Al-Fuqaha. Using connection expansion to reduce control traffic in manets. In *IWCMC'10*, pages 534–538, 2010.

[3] Namusale Chama and Rute Sofia. A discussion on developing multihop routing metrics sensitive to node mobility. *Journal of Communications*, 6:56–67, 2011.

[4] Xi Hu, Jinkuan Wang, and Cuirong Wang. Mobility-adaptive routing for stable transmission in mobile ad hoc networks. *Journal of Communications*, 6(1):79–86, 2011.

[5] Xu Li, Natalie Mitton, and David Simplot-Ryl. Mobility predictibility based neighborhood discovery in mobile ad-hoc networks. https://ece.uwaterloo.ca/ x279li/publications/networking11.pdf, 2011. accessed:14/7/2011, 2011.

[6] C. A. Hidalgo M. C. Gonzalez and A.-L. Barabasi. Understanding individual human mobility patterns. *Nature*, 453, no. 7196:779–782, june 2008.

[7] Ju. Suyang and J.B. Evans. Mobility-aware routing protocol for mobile ad-hoc networks. In *Communications Workshops, 2009. ICC Workshops 2009. IEEE International Conference on*, pages 1–6, 2009.