

A Routing Metric to Improve Route Stability in Mobile Wireless Sensor Networks

Yi-Han XU, Yin WU, Jun SONG

College of Information Science and Technology, Nanjing Forestry University,
Nanjing, 210037 - China

[e-mail: {xuyihan, wuyin, songjun@njfu.edu.cn}]

*Corresponding author: Yi-Han XU

*Received October 20, 2015; revised December 30, 2015; revised March 7, 2016; accepted March 6, 2016;
published May 31, 2016*

Abstract

The hop count routing metric is widely used in routing protocols of Wireless Sensor Networks (WSNs) due to its simplicity and effectiveness. With a lower hop count route, fewer transmissions are required to send a packet from the source to the destination. This can improve the throughput of a network because fewer transmissions results in less channel contention and interference. Despite this, the hop count routing metric may not be ideal for mobile scenarios where the topology of a network changes constantly and rapidly. In this paper, we propose to increase route stability in mobile WSNs by discovering paths that are more stable during route discoveries using routing metrics. Two routing metrics were proposed, the true beauty of these routing metrics lies in the fact that they can even be used even without specialized hardware such as GPS and other sensors. We implemented the proposed routing metrics in the AODV routing protocol and found that they are highly effective and outperform other stability-based routing metrics and the hop count routing metric.

Keywords: Routing metric, Mobile wireless sensor networks, Stability-based, Node mobility

This project was supported by Introduction of high-level talents and overseas returnees scientific fund in Nanjing Forestry University (No.: GXL015) and 2015 University Students Practice Innovation Training Program (No.: 2015sjcx122)

1. Introduction

Wireless Sensor Networks (WSNs) are self-organizing, self-forming and infrastructure-less communication networks characterized by multi-hop transmissions and the use of wireless links. Many properties of WSNs can be attributed to its routing protocols. The hop count routing metric is one of the most widely used routing metrics in WSNs routing protocols due to its simplicity and effectiveness. However, the hop count routing metric only performs well in static to low mobility scenarios, the same cannot be said in higher mobility scenarios, in which nodes are moving at higher speeds and the network topology changes more rapidly. Besides, the hop count routing metric is also known to cause the border effect [1] in high node density scenarios, where links are formed through border nodes and have short lifetimes. A small movement from either one of the two nodes forming such a link could cause the link to break. When a link is broken, control packets are communicated among the nodes to deal with the link failure. These control packets increase channel contention and interference which reduces network throughput. To deal with the deficiencies of the hop count routing metric in mobile scenarios, stability-based routing metrics have been proposed. However, these routing metrics are usually either only marginally effective or incur additional cost by requiring the use of information from specialized hardware such as GPS sensor and compass. In this paper, we first investigate several methods for improving routing in mobile WSNs and ultimately propose two routing metrics for finding stable routes which can be used independent with specialized hardware. Analysis and simulation show that the proposed routing metrics are effective and outperform other stability-based routing metrics and the hop count routing metric. The contributions of this paper are summarized as follows:

1. The proposal of a routing metric that is based on the length of the links in a path.
2. The proposal of an extension to the above routing metric that also considers node mobility.
3. The proposal of a method to estimate the length of a link or the distance between two neighboring nodes without using specialized hardware.
4. The proposal of a method to estimate the mobility of a node without using specialized hardware.

The remainder of this paper is organized as follows. Section 2 gives a literature survey on the most relevant related works. Section 3 provides a high-level description and analysis of our proposed methods. Results and discussion are provided in Section 4. Finally, we conclude in Section 5.

2. Related Work

In [2], the Associativity-Based Routing (ABR) protocol was proposed. In ABR, link stability is measured using “associativity ticks”. The author claimed that links that are stable for

at least a threshold amount of time are likely to continue being stable. In [3], an on-demand routing protocol called Power and Mobility Aware Routing (PMAR) was proposed. It combines features from other routing protocols such as RREQ propagation control using geographical information (first used in location-aided routing protocols such as LAR [4] and PMLAR [5]), and estimating the lifetime of the link between two neighboring nodes using their locations and velocities. In PMAR, the estimated link lifetime is used in the following manner: when an intermediate node receives a RREQ, it calculates the expiry time of the link formed by itself and its previous hop by adding the estimated link lifetime to the previous hop timestamp recorded in the RREQ. If the node finds that the link is expired, it drops the RREQ. However, we do not agree to this method of using the link lifetime because a link is clearly not expired on broken if the node received a RREQ on the link. If the link was broken, the node would not have received the RREQ. In [6], Flow Oriented Routing Protocol (FORP) was proposed. It is extended from the DSR routing protocol. Using geographical and node mobility information, the amount of time before a link is broken known as the Link Expiration Time (LET) is estimated. The Route Expiration Time (RET) of a path is the minimum of the LETs of the links in the path. In FORP, the path with the highest RET is selected during route discoveries. In [7], a nameless extension of the DSR routing protocol was proposed. For ease of exposition, we refer to this protocol as DSR-stable. Similar to AODV-RRS [8], the concept of stable and caution zones is used in DSR-stable. However, as opposed to nodes only participating in route discoveries (forward RREQs) when they are inside stable zones in AODV-RRS, in DSR-stable, a node that receives a RREQ calculates the link availability of the link formed by it and its previous hop, and then modifies the aggregate path availability value on the RREQ before forwarding the RREQ. Several RREQs reach the destination through different paths and the destination selects the path with the highest availability, where the availability of a path is the minimum link availability among the availabilities of the links in the path. In [1], Link Stability Based AODV (LSB-AODV) was proposed. It tries to estimate the distance between two neighboring nodes using the received signal strengths of packets with the log-distance path loss radio propagation model. The Link Stability Factor (LSF) is used to measure the stability of a link based on its length. Its equation is given as $LSF = 1 - d/R$, where d is the estimated distance between a node and its previous hop, and R is the maximum node transmission range. The Path Stable Factor (PSF) of a path is given by the minimum of the LSFs of its links. The authors showed that LSB-AODV achieves higher packet delivery ratio, lower packet latency, and lower normalized routing load when compared to AODV. While LSB-AODV performs well, using the received powers of packets which can fluctuate greatly to estimate the distance between two nodes could be problem. Besides, the path loss exponent in the log-distance path loss radio propagation model also varies in different environments. The paper in [9] reviews some routing metrics available for wireless mesh networks such as Expected Transmission Time (ETT) [10], Weighted Cumulative Expected Transmission Time (WCETT) [10], and Metric of Interference and Channel Switching (MICS) [11]. However, these routing metrics are either designed for wireless mesh networks, or have a different objective (for example, to improve throughput or reduce interference) and are irrelevant to the problem that we seek to solve. In [12], a Distributed Compressive Sparse Sampling (DCSS) algorithm was proposed to solve the problem of recover the n -dimensional data values by querying only $m \ll n$ sensors based on some linear projection of sensor readings. Extensive experiments on both synthetic and real data were set and results show that by querying only the

minimum amount of m sensors using the DCSS algorithm, the compressive sampling recovery accuracy outperforms existing sparse random matrices and can be as good as those using random dense measurement matrices but using much less number of sensors. In [13][14], surveys on routing protocols for mobile ad-hoc networks were provided and authors discussed the advantages and disadvantages of these protocols and comparing their respective functionalities meanwhile some open issues and possible directions of future research are given. In [15], authors proposed a novel method to achieve k -barrier coverage of directional sensors (e.g., cameras) and found the solution of what is the minimum number of mobile sensors required to form (k) -barrier coverage. In [16], a hybrid CSMA/TDMA MAC called iQueue-MAC was proposed to adapt the variable/bursty traffic in both single and multi-channel modes. By using this iQueue-MAC, packet delay is reduced and throughput is increased.

3. Improving Route Stability

3.1 Estimating Link Remaining Lifetime

In mobile WSNs, it is desirable to discover and setup stable routes, i.e., routes consisting of links with long remaining lifetime. If nodes are equipped with GPS sensor, the remaining lifetime of a particular link in a network can be estimated based on geographical and node mobility information, i.e., the locations and velocities of the two nodes forming the link. **Fig. 1** shows two nodes A and B separated by distance d_0 . To derive the time link A-B remains up/available, we compute the amount of time node B remains within the maximum transmission range R from node A. Taking node A as the reference, the position of node B with respect to node A is given by $(x_{BA}, y_{BA}) = (x_B - x_A, y_B - y_A)$. The distance of node B from node A is initially $d_{t=0} = d_0 = \sqrt{x_{BA}^2 + y_{BA}^2}$. Subsequently, due to the motion of the two nodes, the position of node B with respect to node A after t seconds is given by $(x_{BA'}, y_{BA'})$ where:

$$x_{BA'} = x_{BA} + \left| \vec{v}_{BA} \right| t \cos \theta_{BA} \dots \dots \dots (1)$$

and

$$y_{BA'} = y_{BA} + \left| \vec{v}_{BA} \right| t \sin \theta_{BA}$$

We would like to know the amount of time t before node B goes out of range from node A. Assuming that the velocity of node B with respect to node A is constant, the critical time at which node B goes out of range from node A can be determined by solving the following equation:

$$d_{t^2} = (x_{BA'})^2 + (y_{BA'})^2 = R^2 \dots \dots \dots (2)$$

Substituting Eq. 1 into Eq. 2 and rearranging, we have the following equation:

$$\left| \vec{v}_{BA} \right|^2 t^2 + 2 \left| \vec{v}_{BA} \right| (x_{BA} \cos \theta_{BA} + y_{BA} \sin \theta_{BA})t + d_0^2 - R^2 = 0 \dots \dots \dots (3)$$

where:

$$\vec{v}_A = \left(\left| \vec{v}_A \right| \cos \theta_A \right)_i + \left(\left| \vec{v}_A \right| \sin \theta_A \right)_j$$

$$\vec{v}_B = \left(\left| \vec{v}_B \right| \cos \theta_B \right)_i + \left(\left| \vec{v}_B \right| \sin \theta_B \right)_j$$

$$\begin{aligned} \vec{v}_{BA} &= \vec{v}_B - \vec{v}_A = \left(\left| \vec{v}_B \right| \cos \theta_B - \left| \vec{v}_A \right| \cos \theta_A \right) \vec{i} + \left(\left| \vec{v}_B \right| \sin \theta_B - \left| \vec{v}_A \right| \sin \theta_A \right) \vec{j} \\ \left| \vec{v}_{BA} \right| &= \sqrt{\left(\left| \vec{v}_B \right| \cos \theta_B - \left| \vec{v}_A \right| \cos \theta_A \right)^2 + \left(\left| \vec{v}_B \right| \sin \theta_B - \left| \vec{v}_A \right| \sin \theta_A \right)^2} \\ \theta_{BA} &= \tan^{-1} \left(\frac{\left(\left| \vec{v}_B \right| \sin \theta_B - \left| \vec{v}_A \right| \sin \theta_A \right)}{\left(\left| \vec{v}_B \right| \cos \theta_B - \left| \vec{v}_A \right| \cos \theta_A \right)} \right) \end{aligned}$$

Note that Eq.3 is a quadratic equation of the form $ax^2 + bx + c = 0$. Where $x = t, a = \left| \vec{v}_{BA} \right|^2, b = 2 \left| \vec{v}_{BA} \right| (x_{BA} \cos \theta_{BA} + y_{BA} \sin \theta_{BA}),$ and $c = d_0^2 - R^2$. The root of a quadratic equation can be computed by using the method of completing the squares with the following equation:

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \dots \dots \dots (4)$$

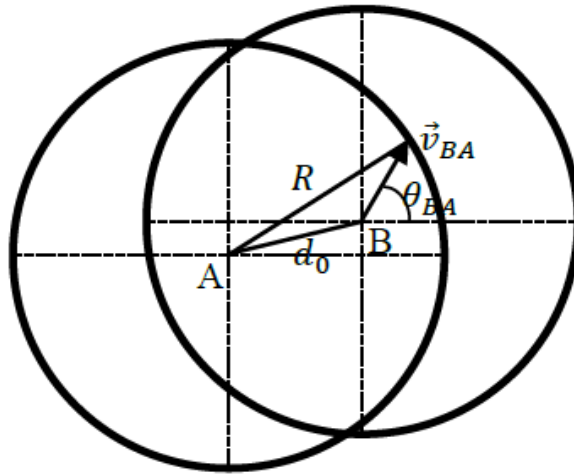


Fig. 1. Estimating the remaining lifetime of a link using nodes' locations and velocities

A path consists of one or more links. The remaining lifetime of a path is the minimum of the remaining lifetimes of the links in the path. Path remaining lifetime can be used as the routing metric in WSNs routing protocols to discover and setup stable routes. For example, the path with the longest remaining lifetime can be selected during route discoveries. In this paper, we refer to this method as the Maximum Path Lifetime (MPL) method. This method will serve as the benchmark method for developing new methods of improving route stability in mobile WSNs.

We now describe the deficiencies of the MPL method. Firstly, instantaneous node

velocities were used in deriving the estimated remaining lifetime of a link. However, nodes do not necessary have constant velocities. Nodes could change their velocities suddenly after their velocities are sampled. When that happens, the estimated remaining lifetime of a link or path becomes inaccurate. Secondly, nodes are required to use information that can be obtained only by using sensors when estimating the remaining lifetime of a link-location, speed and heading direction. As a result, nodes are to be equipped with sensors and this incurs additional cost. In Section 3.2, we propose a method to further improve upon the MPL method.

3.2 Improving Route Stability by Considering Link Lengths

The main problem with the MPL method is that nodes could suddenly change their velocities thus leading to inaccurate estimated link remaining lifetimes. To resolve this problem, besides link remaining lifetime, link length could be used as a secondary measure of link stability. Intuitively, shorter links have longer expected lifetime than longer links. Repeatedly solving Eq. 3 with random node velocities at different maximum speeds leads us to the graphs in Fig. 2. From the figure, we observed that in general shorter links are indeed more stable than longer links. Hence, it is better to setup longer (higher hop count but shorter link lengths) but potentially more stable routes.

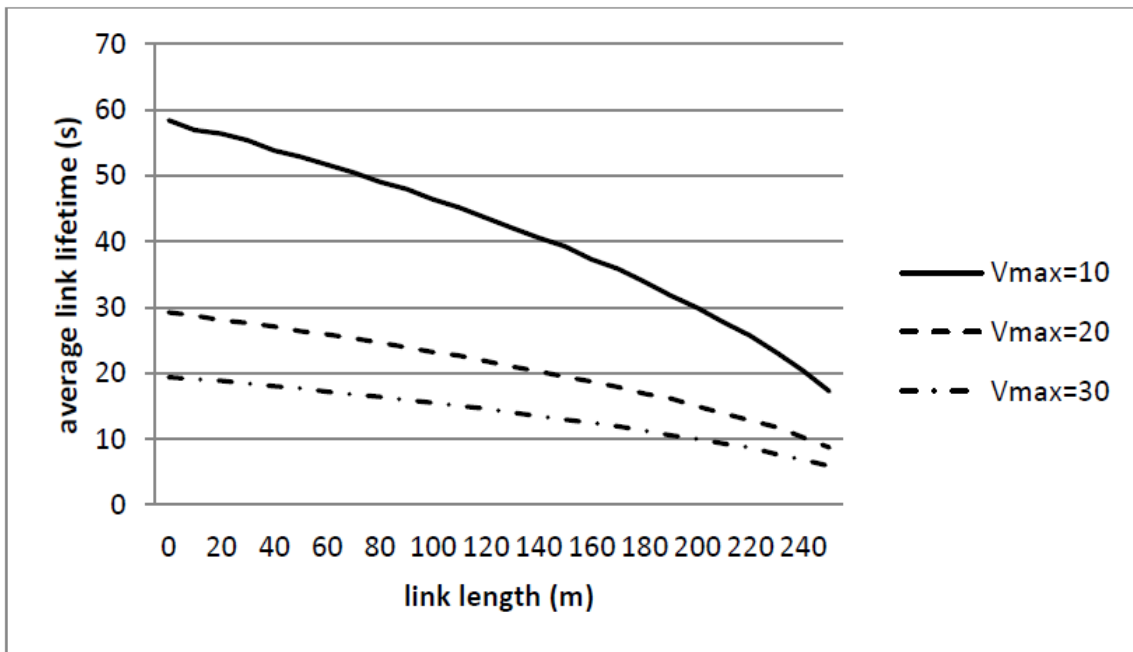


Fig. 2. Average link lifetime vs. link length

One question remains – How do we modify a routing protocol such that it discovers and setups routes consisting of short links? A way to do this in reactive routing protocols is by modifying the route discovery process such that paths consisting of long links are not discovered. This is the idea proposed in [8] with the safety region based route discovery method. To avoid the formation of routes consisting of long links, nodes that reside outside a threshold distance

from their previous hop ignore the RREQ that they receive during route discoveries. Combining the safety region based route discovery method with the MPL method, we have the safety region based route discovery with MPL (SR,MPL) method. This method selects the path with the highest estimated remaining lifetime among paths consisting of only links that are shorter than a threshold length. It guarantees a certain degree of route stability by using link length information and it performs well as can be seen in Section 4.

3.3 Proposed Routing Metric 1: Sum of Squared Differences of Link Lengths from Threshold

While the SR,MPL method performs well, we discovered several problems. Firstly, the use of the safety region based route discovery method could lead to a potential devastating problem, i.e., it could result in a scenario where a path from a source to a destination could not be found even if one or more path between the pair of nodes exist in the network. This happens because some RREQs are ignored at certain nodes. Secondly, link remaining lifetimes are computed using the locations and velocities of the nodes. This has the drawback that the locations and velocities of the nodes are required which means that specialized hardware such as GPS sensors must be used which incurs additional cost.

To resolve the first problem, instead of making certain nodes ignore RREQs, a routing metric could be used to select better paths. To resolve the second problem, the routing metric that we develop should not be computed using the locations and velocities of the nodes. With this in mind, we now propose the *Sum of Squared Differences of Link Lengths from the Threshold Length (SSDLLThresh)* routing metric. It satisfies both requirements discussed above, and its equation is given as follows:

$$SSDLLThresh_p = \sum_{l \in p} (length_l - \min(length_l, THRESHOLD_{LENGTH}))^2 (5)$$

Where $length_l$ is the length of link l , l is a link in path p , and $THRESHOLD_{LENGTH}$ is the threshold link length. The selected path p^* is defined as follows:

$$p^* = \operatorname{argmin}_{p \in P} (SSDLLThresh_p) \dots \dots \dots (6)$$

Where P is the set of discovered paths from a source to a destination.

The *SSDLLThresh* routing metric assigns a penalty to links that exceed the threshold length. Unlike ignoring of RREQs in the safety region based route discovery method, a routing metric does not reject paths with long links; it merely favors paths with shorter links over paths with longer links. Hence, the problem of not finding a path from a source to a destination is resolved. With the use of the threshold link length, we also avoided the problem of nodes selecting the shortest links leading to the formation of routes with very high hop count as highlighted in [17] and [18]. The use of the minimum sum of squares also has a hidden benefit, i.e., it is inherently hop count minimizing. This is because when there are more links in a path, there is a higher probability that more of the links will exceed the threshold link length assuming all other things equal, but if all link lengths along a path are less than $THRESHOLD_{LENGTH}$, then

hop count do not have an effect on the metric value for the path.

The *SSDLLThresh* routing metric specifies the use of a parameter: the threshold link length (*THRESHOLD_LENGTH*). Many heuristics could be used to set the value of this parameter. One of the heuristics is to set the threshold link length according to a desired minimum link remaining lifetime. From Fig. 1, it can be observed that the remaining lifetime of the link between nodes A and B is minimum when the velocity of node B relative to node A is in the direction of from node A to node B. In other words, if nodes A and B are moving directly away from each other, the remaining lifetime of the link between them is shortest. Assuming that nodes have maximum speed v_{max} , the desired link length threshold can be set to the value of $length_l^{max}$ using Eq.7, where $t_l^{desired_min}$ is the desired minimum remaining lifetime of a link l , and R is the maximum node transmission range.

$$t_l^{desired_min} = \frac{R - length_l^{max}}{2v_{max}} \rightarrow length_l^{max} = R - 2v_{max}t_l^{desired_min} \dots (7)$$

We now illustrate with an example how the routing metric value of a path is computed. Consider the network in Fig. 3. The lengths of the links are given next to the links in the figure. Suppose node 1 is the source, node 4 is the destination, and *THRESHOLD_LENGTH* is set to 200 m. three paths exist in the network: 1-2-3-4, 1-5-4, and 1-6-7-8-4. Path 1-2-3-4 has the lowest *SSDLLThresh* value and is selected.

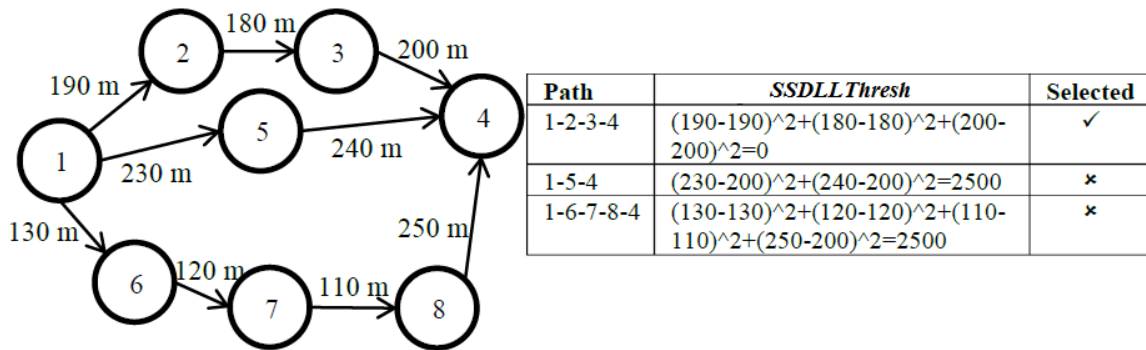


Fig. 3. An example illustrating how the first proposed routing metric is used

By favoring paths consisting of short links, we argue that the *SSDLLThresh* routing metric is able to handle the unpredictable nature of node mobility. For example, suppose two neighboring nodes forming a link and the link has a high expected remaining lifetime. Then, suppose the nodes change their mobility abruptly. Due to the change in node mobility, the expected remaining lifetime that was computed earlier becomes inaccurate. However, if the *SSDLLThresh* routing metric is used, due to the short link length, it takes some amount of time before the link can become broken. Hence, the *SSDLLThresh* routing metric is effective regardless of the node mobility model used. However, the same cannot be said in some other

routing metrics such as the one used in ABR [3] or routing metrics that are based on historical information.

3.4 Proposed Routing Metric 2: Sum of Squared Combined

In Section 3.3, we proposed the *SSDLLThresh* routing metric. It tries to select paths consisting of short links by assigning a penalty to links that exceeds a threshold link length. However, it does not consider node mobility. In this section, we extend it to also consider node mobility. We call the extended routing metric Sum of Squared Combined (SSC) and its equation is given as follows:

$$SSC_p = \sum_{l \in p, n \in p, n \neq src(p)} ((length_l - \min(length_l, THRESHOLD_{LENGTH})) + mobility_n)^2 \dots\dots\dots(8)$$

where $mobility_n$ is the measure of the mobility of node n (an intermediate or the destination node) in path p , l is a link in path p , and $src(p)$ is the source of path p . It is worth noting that we designed the routing metric to do the following things: a) penalize links exceeding $THRESHOLD_{LENGTH}$ by giving them “1 demerit point” for every meter they exceed $THRESHOLD_{LENGTH}$, and b) penalize nodes in the path by giving them “1 demerit point” for every unit of node mobility. Hence, we directly plus $mobility_n$ in Eq. 8. In reality, we can actually multiply with k for each of the factors (such as link length and/or node mobility). For example, we may give more weight to node mobility by multiplying with 2 or 3. Or we can give some powers to them. When node mobility information is available, $mobility_n$ is generally the speed at which the considered node is moving in unit of meters per second. The selected path p^* is defined as follows:

$$p^* = argmin_{p \in P}(SSC_p) \dots\dots\dots(9)$$

Compared to the *SSDLLThresh* routing metric, the *SSC* routing metric assigns penalty not only to paths consisting of long links but also paths that are formed through nodes with a high degree of node mobility.

3.5 Replacing the Use of True Link Length Information with Estimated Values

In the proposed routing metrics, geographical and node mobility information are required. A method of obtaining such information is through the use of sensors such as GPS. However, requiring the use of sensor incurs additional cost. For example, we found the price of the GPS sensor board MTS420 (circa 450 USD) to be about thrice the price of the IRIS mote XM2110 (circa 150 USD). Besides, there are also scenarios where sensors are inapplicable, for example, GPS sensors perform poorly inside buildings and are unsuitable for low power nodes. In this section, we look to substitute the use of actual link length information with their estimated values. In some WSN routing protocols, nodes periodically broadcast HELLO messages to allow other nodes to sense their presence. This feature could be exploited to estimate the distance between two nodes.

In **Fig. 4**, we show two contrasting scenarios of the distance between two nodes A and B. In **Fig. 4a**, nodes A and B are located close to each other while in **Fig. 4b** the nodes are at a

maximum distance from each other (subject to link A-B not broken). From this figure, it seems that the distance between any two neighboring nodes A and B can be estimated by evaluating the ratio of the number of nodes in the intersection of set U and V to the number of nodes in the union of sets U and V , where U is the set containing node A and its surrounding neighboring nodes, V is the set containing node B and its surrounding neighboring nodes. Therefore, $U = \{A\} \cup N_A, V = \{B\} \cup N_B$, and N_x is the current neighbor set of a particular node x . For simplicity, we refer to the ratio of the number of nodes in the intersection of sets U and V to the number of nodes in the union of sets U and V as *overlap_ratio*.

$$overlap_ratio = \frac{|U \cap V|}{|U \cup V|} \dots \dots \dots (10)$$

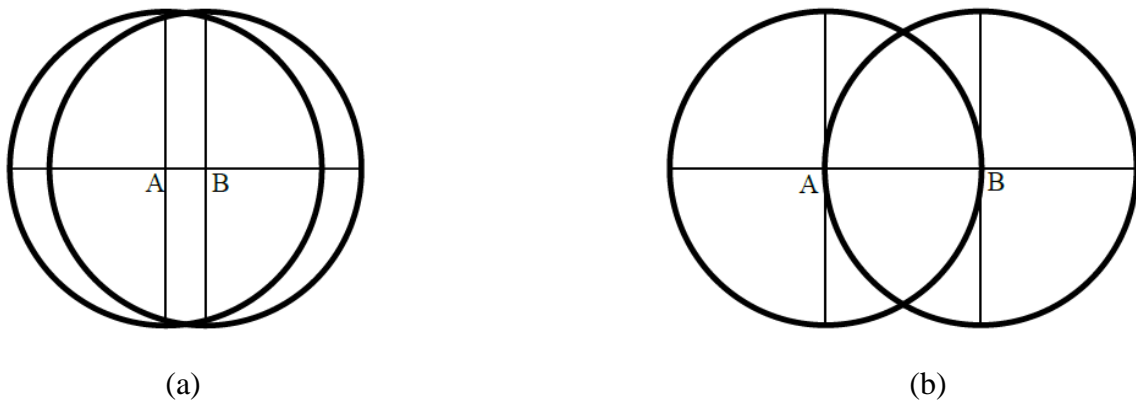


Fig. 4. Estimating the distance between two nodes:

(a) two nodes are close to each other, (b) two nodes are at maximum distance from each other

Assuming that nodes are uniformly distributed, as node density approaches infinity, *overlap_ratio* is approximately equal to the ratio of the area of the overlapping region of two equal circles to the total area covered by the two circles. In the following, we derive the area of the overlapping region of two equal circles as shown in **Fig. 5** From trigonometry, we have:

$$\cos \frac{\theta}{2} = \frac{d/2}{R} = \frac{d}{2R} \rightarrow \theta = 2 \cos^{-1} \frac{d}{2R} \dots \dots \dots (11)$$

The area of the sector with angle θ (ACFD) can be computed by using the concept of proportionality, i.e.:

$$area_{ACFD} = \frac{\theta}{2\pi} \times \pi R^2 = \frac{\theta R^2}{2} \dots \dots \dots (12)$$

Substituting Eq.11 into Eq.12, we have:

$$area_{ACFD} = \frac{(2 \cos^{-1} \frac{d}{2R}) R^2}{2} = R^2 \cos^{-1} \frac{d}{2R} \dots \dots \dots (13)$$

Using Pythagoras’ Theorem, we have:

$$\left(\frac{d}{2}\right)^2 + \left(\frac{h}{2}\right)^2 = R^2 \rightarrow h = \sqrt{4R^2 - d^2} \dots \dots \dots (14)$$

The area of the triangle ACD can be computed as follows:

$$area_{ACD} = \frac{1}{2} \times \frac{d}{2} \times h = \frac{d}{4} \sqrt{4R^2 - d^2} \dots \dots \dots (15)$$

Finally, the area of the overlapping region of the two circles is given by two times the area of the “D-shaped region” CDE or CFD, each of which is given by the area of the sector ACFD minus the area of the triangle ACD.

$$\begin{aligned} area_{overlapping_of_two_equal_circles} &= 2 \times (area_{ACFD} - area_{ACD}) \dots \dots \dots (16) \\ &= 2\left(R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{4} \sqrt{4R^2 - d^2}\right) \\ &= 2R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{2} \sqrt{4R^2 - d^2} \end{aligned}$$

From Eq. 16, the ratio of the area of the overlapping region of the two circles to the area jointly covered by the two circles is given as follows:

$$\begin{aligned} \frac{area_{overlapping_of_two_equal_circles}}{area_{union_of_two_equal_circles}} &= \frac{2R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{2} \sqrt{4R^2 - d^2}}{\pi R^2 + \left(\pi R^2 - \left(2R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{2} \sqrt{4R^2 - d^2}\right)\right)} \\ &\approx overlap_{ratio} \dots \dots \dots (17) \end{aligned}$$

From Eq. 17, we observed that *overlap_{ratio}* is approximately equal to the ratio of the area of the overlapping region of the two circles to the area of the union of the two circles, which is determined only by the distance between the centers of the two circles *d*, and the maximum node transmission range *R*. Hence, to determine the relationship between *overlap_{ratio}* and distance between the two node *d*, we need to express *d* in terms of *overlap_{ratio}*. Alas, we were unable to do that. Instead, we resorted to solving Eq. 18 using the bisection root finding numerical method by setting a constant value for *R* (250 m is assumed in our work) and varying the value of *k*.

$$\frac{2R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{2} \sqrt{4R^2 - d^2}}{\pi R^2 + \left(\pi R^2 - \left(2R^2 \cos^{-1} \frac{d}{2R} - \frac{d}{2} \sqrt{4R^2 - d^2}\right)\right)} - k = 0 \dots \dots \dots (18)$$

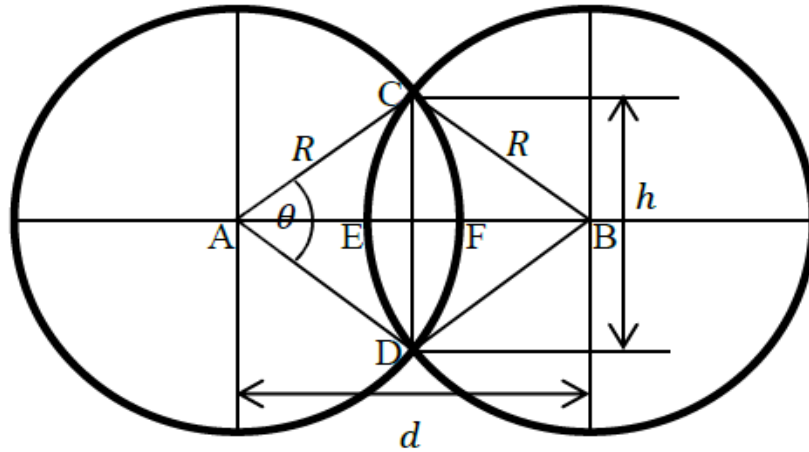


Fig. 5. Finding the area of the overlapping region of two equal circles

The graph of the relationship between link length d and overlap ratio ($overlap_{ratio}$) is plotted in **Fig. 6**. Using the curve fitting method with a polynomial of degree two, the relationship between link length and overlap ratio is as shown by the equation in **Fig. 6**. This equation can be used to estimate the length of a link given the overlap ratio as computed from Eq. 10. The value of overlap ratio is generally in the range of approximately 0.25 to 1.00. For values smaller than that, we simply set the estimated length of the considered link to $R = 250$ m.

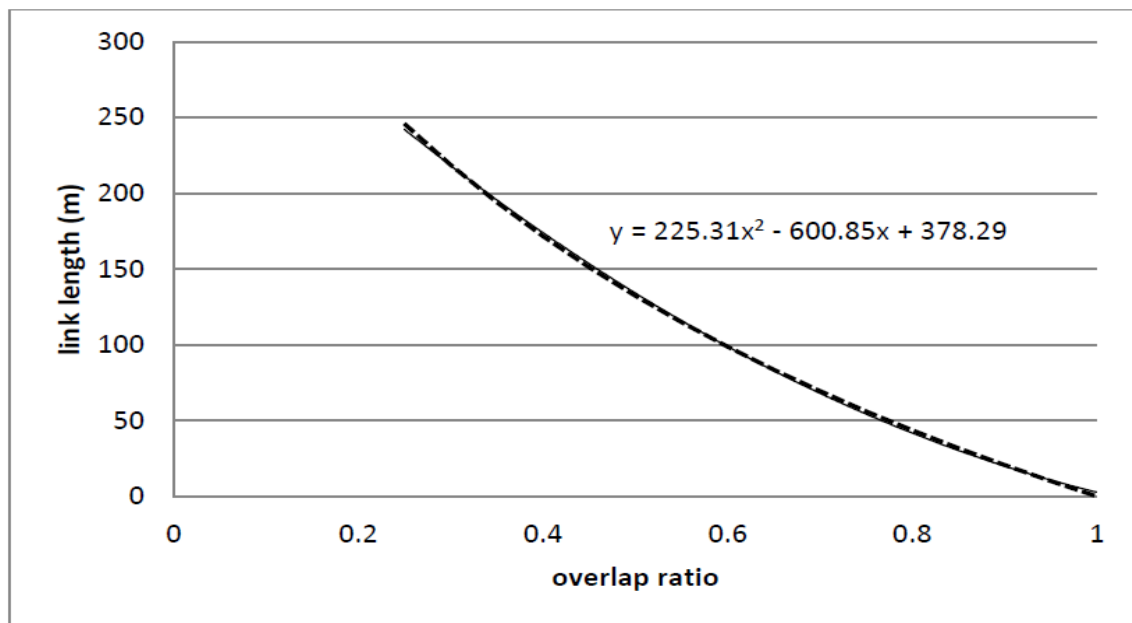


Fig. 6. The plot of the relationship between link length and overlap ratio

The accuracy of the estimated link lengths is governed by how closely the considered regions in a real scenario agree on the assumptions made when deriving the relationship between the overlap ratio and the length of a link, i.e., i) the uniformity of the node distribution in the considered regions, and ii) node density in the considered regions. When node distribution is more uniform or node density is higher, the accuracy of the estimations will be higher. This is because by using Eq. 17, we are assuming uniform node distribution and high/infinite node density. On the other hand, we try to use simulation to determine the relationship between $overlap_{ratio}$ and link lengths. However, we found that the relationship is only valid for the simulation settings used. It not be valid for other scenarios. In this paper, we estimate link length using overlap ratio. If the relationship between $overlap_{ratio}$ and link lengths is not accurate, then the estimated link length is also inaccurate. This can cause a negative effect on the route selection. In other words, less stable route may be selected due to wrong information. We also identified stale neighborhood information as another source of link lengths estimation inaccuracy. In this paper, HELLO beacon messages are broadcasted by nodes at fixed interval and asynchronously. Suppose there are three nodes A, B, and C. Node A send HELLO packets at time 1.0s, 2.0s, ..., node B sends HELLO packets at time 1.1s, 2.1s, ..., and node C sends HELLO packets at 1.2s, 2.2s, Also, let's assume at time 1 s, nodes B and C are not neighbors of A, but at time 2s, nodes B and C are neighbors of A. Because node B only send HELLO packet at 2.1 s, and node C only send HELLO packet at 2.2s, at time 2.0 s, nodes B and C are neighbors of node A, but node A do not realize this fact because it has not receive HELLO packets from nodes B and C (this is the inaccuracy). Hence, the current neighbor set may not be accurate. Since neighborhood information is used in estimating link length and node mobility, the link length and node mobility estimation ought to be less accurate then it should be. Therefore, stale neighborhood information can cause a negative effect on the route selection. Another example, suppose node y was previously a neighbor of node x but has moved out of transmission range from node x but the neighbor list entry of node y in node x 's neighbor list has not expired yet. Hence, node x still regards node y as its neighbor. A similar scenario is encountered when a new node z has moved within transmission range from node x but has not broadcast a new HELLO message. In this case, node x does not yet recognize node z as its neighbor. Fig.7 presents how sensitive the stale neighborhood information could affect link length via standard deviation of link length.

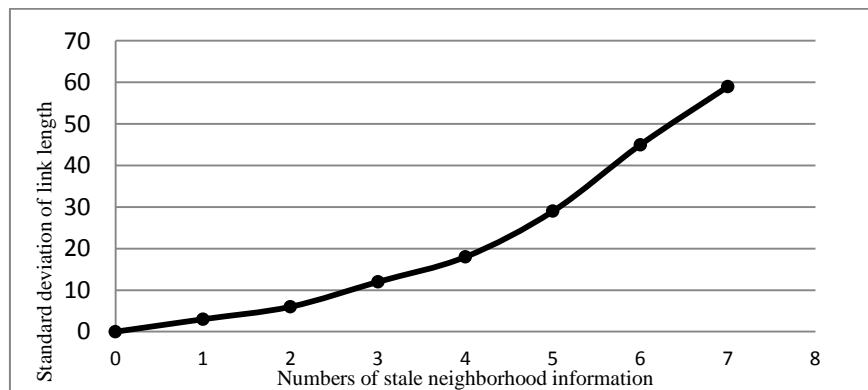


Fig. 7. Relationship between stale neighborhood information and standard deviation of link length

When estimated link lengths are used in place of actual link lengths when computing the routing metric value of links using the *SSDLLThresh* routing metric, we actually transformed the routing metric from one that considers link length to another that considers the ratio of the number of nodes in the intersection of the transmission coverage areas of two nodes forming a link to the number of nodes in the union of the transmission coverage areas of two nodes forming a link to the number of nodes in the union of the transmission coverage areas of the two nodes. As discussed above, a source of link length estimation inaccuracy is in how closely the assumptions made are followed in actual scenarios. However, we argue that our routing metrics continue to work even if the assumptions are only followed loosely, albeit with lower performance.

3.6 Estimating Node Mobility

In section 3.4, we proposed a routing metric that takes into consideration link length and node mobility. A method to estimate link length was established in Section 3.5. Without relying on sensors, it is difficult to measure node mobility. Fortunately, it turns out that the method proposed in Section 3.5 to estimate the distance between two neighboring nodes can also be used to estimate the amount of distance a node has moved from its previous location with little modification. Instead of computing *overlap_ratio* using a node’s current neighbor set and previous neighbor set as shown in Eq.19.

$$overlap_{ratio} = \frac{|N_x \cap N_x^p|}{|N_x \cup N_x^p|} \dots \dots \dots (19)$$

Where, N_x^p is the previous neighbor set of the node x.

Fig. 8 shows the difference between the two different operations: i) estimating the length of the link between two different nodes, and ii) estimating the amount of distance a node has moved from its previous location.

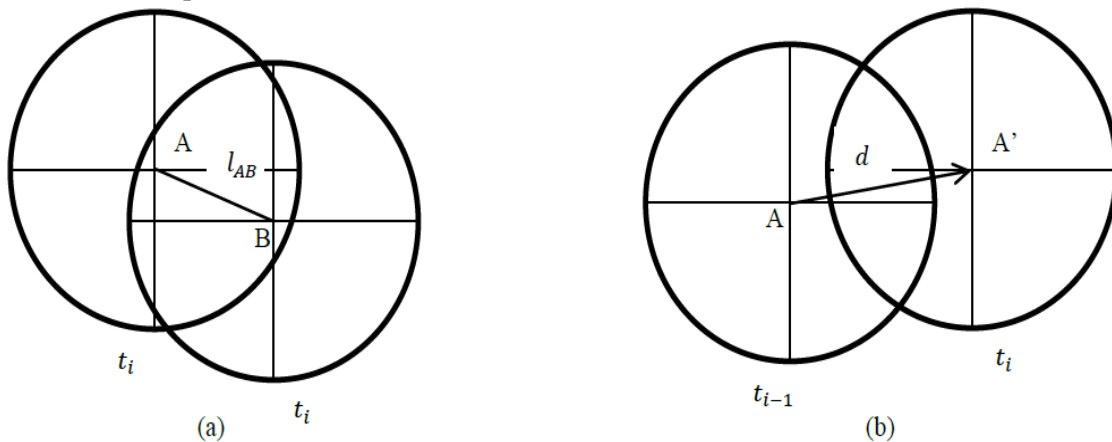


Fig. 8. (a) estimating the length of a link, vs. (b) estimating the distance a node has moved from its previous location

Generally, the overlap ratio in Eq.19 measures the degree of change in the neighbor set of a node between two successive sampling times. The change in the neighbor set of a node can be viewed as one of the following scenarios: i) the node is moving but its neighbors are static, ii) the node is static but its neighbors are moving, and iii) both the node and its neighbors are moving. In each scenario, a higher change in the neighbor set of a node signifies higher relative velocities between the node and its neighbors. The change in neighbor set of a node is also used in many other routing metrics to measure node mobility, for example, Mobility Factor (MF) [19] and Neighbor Change Ratio (NCR) [20]. The amount of distance a node has moved from its previous location is then used as the measure of its mobility ($mobility_n$) in Eq.8.

4. Results and Discussion

In this section, we provide the details of the evaluation work done and then discuss upon the results. We used the network simulator 2 (ns-2) [21] network simulator in our evaluation. We placed 75 nodes in a rectangular region of dimension 1500 m by 300 m (same dimension as used in [22]). Each scenario was repeated 20 times using different seed numbers in the range [1, 20] when generating the node mobility and network traffic patterns. The network traffic used in each instance of simulation consists of 5 pairs of CBR traffic flows each flowing at the rate of 40 Kbps (512 B packet size at the rate of 10 packets/s; $1Ki = 2^{10} = 1024$, $1K = 10^3=1000$), and (each flow) starting at a random time in [0, 20] s simulation time. The physical and MAC related configurations were set to emulate the IEEE 802.11 ERP-DSSS [23] [24] physical layer. Nodes have a maximum transmission range R of 250 m and a carrier sensing range of 550 m. The node mobility model used is the random waypoint model. We varied the degree of node mobility by varying the maximum node speed from 5 m/s to 25 m/s in increments of 5 m/s while the pause time is set to 0 s to that nodes are constantly moving.

The routing metrics used for performance evaluation are:

- **Packet delivery ratio (%)**: the total number of data packets that were successfully delivered divided by the total number of data packets sent by all sources
- **Normalized routing load**: the number of transmissions of all routing packets (RREQ, RREP, RERR, and HELLO) divided by the total number of data packets that were successfully delivered. It is a measure of the average number of transmissions required for routing packets for every data packet successfully delivered.
- **Average packet latency (ms)**: the average of the end-to-end delays of data packets that were successfully delivered.
- **Average hop count**: the average hop count of data packets that were successfully delivered.
- **Number of route discoveries**: In AODV, a route discovery is uniquely identified by a <source, broadcast ID> pair. The number of route discoveries is an indirect measure of the stability of the discovered routes as more route discoveries are needed when they are more route breakages.

4.1 Comparing AODV-minSSDLLThresh-true, AODV-SR,MPL and AODV- MPL

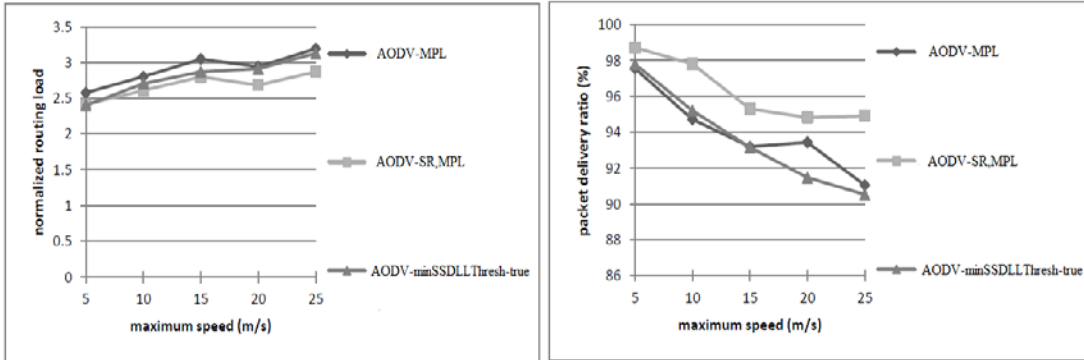
The purpose of this comparison is threefold: i) to show that the path with the highest

estimated remaining lifetime does not necessary performs best because link remaining lifetimes are computed based on instantaneous node velocities and nodes could change their velocities after their velocities have been sampled thus leading to inaccurate estimated link remaining lifetimes, ii) to show that paths formed by shorter links are more stable than paths formed by longer links, and iii) to show that the proposed *SSDLLThresh* routing metric performs well despite not being able to accurately select the most stable paths in terms of the estimated remaining lifetime due to not using node velocity information. Not using node velocity information is a constraint that we set when developing our routing metrics so that the developed routing metrics can be used without using sensors.

AODV-SR,MPL is a simple extension of AODV-MPL that tries to provide a certain level of guarantee of route stability by allowing only links that are less than the threshold length (set to 200 m for AODV-SR,MPL). By constraining to a subset of available paths, the selected path might be suboptimal in regard to maximizing the estimated link remaining lifetimes could be inaccurate. Indeed, we found that AODV-SR,MPL outperform AODV-MPL. AODV-SR,MPL obtained higher packet delivery ratios (Fig. 9a) and lower normalized routing loads (Fig. 9b). This is because routes setup by AODV-SR,MPL are considerably more stable as can be verified by the number of route discoveries produced (Fig. 9e). Compared to AODV-MPL, AODV-SR,MPL produced approximately 14.5%, 9%, 12.5%, 16.4% and 13.2% fewer route discoveries at 5, 10, 15, 20 and 25 m/s, respectively. This shows that routes break less frequently when AODV-SR,MPL is used than when AODV-MPL is used, which indicates that the routes found by AODV-SR,MPL are more stable than the routes found by AODV-MPL. The average hop counts are higher with AODV-SR,MPL (Fig. 9d) because the links of the routes have limited length. In terms of the average packet latency (Fig. 9c), there was no clear winner as AODV-SR,MPL outperformed AODV-MPL at 5, 10 and 25 m/s while the reverse is true at 15 and 20 m/s.

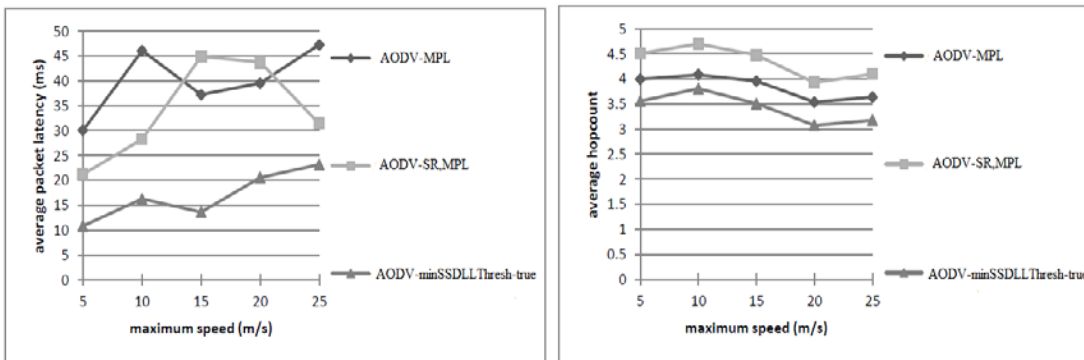
AODV-minAODV-SR,MPL-true is an extension of AODV that uses the *SSDLLThresh* routing metric. True link length information was used in computing the *SSDLLThresh* values (assuming that the necessary information is obtained from GPS sensors), hence the “-true” suffix at the end of the name of the method. The threshold link length value is set to $0.7R = 175$ m in order to balance the tradeoff between path remaining lifetime and hop count. From Fig. 2 it can be seen that links of 175 m long achieved an average lifetime of approximately 60% of the average lifetime of links of 0 m long. From our evaluations, we found that AODV-min *SSDLLThresh*-true performed quite closely to AODV-MPL but is outperformed by AODV-SR,MPL in terms of packet delivery ratio (Fig. 9a). When comparing the normalized routing loads (Fig. 9b), AODV-min *SSDLLThresh*-true performs closer to AODV-SR,MPL at lower degrees of node mobility (5-15 m/s) but closer to AODV-MPL at higher degrees of node mobility (20-25 m/s). AODV-SR,MPL produced the lowest normalized routing loads because some of the RREQs are ignored at certain nodes. AODV-min *SSDLLThresh*-true produced the lowest average packet latencies as can be seen in Fig. 9c. This could be due to the lower average hop counts of packets when AODV-min *SSDLLThresh*-true is used (Fig. 9d). The stability of the routes found by AODV-min *SSDLLThresh*-true is the best at lower degrees of node mobility (5-15 m/s). At higher degrees of node mobility, the routes found by AODV-SR,MPL were more stable than those found by AODV-min *SSDLLThresh*-true as can be seen in Fig. 9e. From this comparison, we found that the *SSDLLThresh* routing metric does a good job at approximating

and even outperform the path remaining lifetime routing metric (AODV-MPL) at lower degrees of node mobility (5-15 m/s) despite using less information (only link lengths are used in AODV-minSSDLLThresh-true as opposed to both link lengths and node velocities in AODV-MPL). From this, it also seems that the SSDLLThresh routing metric can be further improved by considering node mobility.



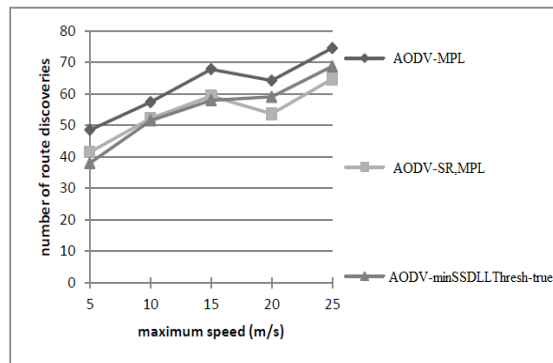
a. packet delivery ratio

b. normalized routing load



c. average packet latency

d. average hop count



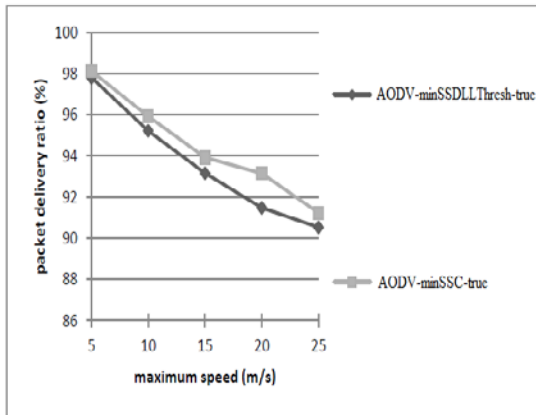
e. number of route discoveries

Fig. 9. Comparing AODV- minSSDLLThresh-true, AODV-SR,MPL and AODV-MPL

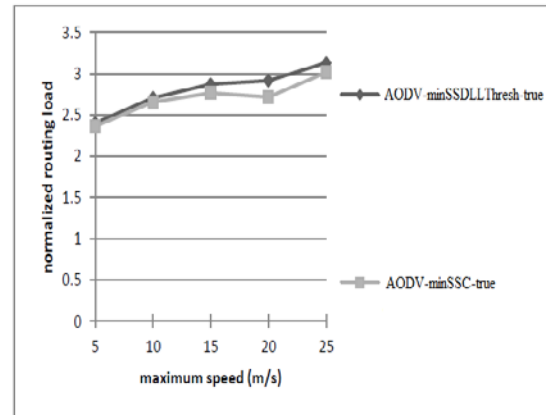
4.2 Comparing AODV-minSSDLLThresh-true and AODV-minSSC-true

AODV-minSSC-true is the direct extension of AODV-minSSDLLThresh-true. AODV-minSSC-true takes into account both link lengths and node mobility when evaluating paths as opposed to only considering link lengths in AODV-minSSDLLThresh-true. True link length information was used in both methods, and the threshold link length value used in both methods is $0.7R = 175\text{m}$. the purpose of this comparison is to show that by taking into account node mobility when selecting paths, paths that are more stable could be found.

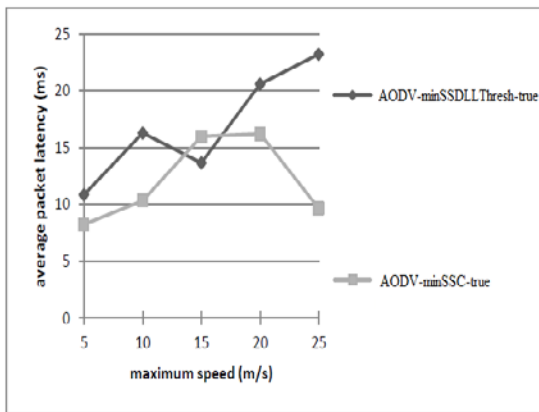
We found that AODV-minSSC-true outperforms AODV-minSSDLLThresh-true in most aspect due to the use of extra information – node mobility. AODV-minSumSqCombined-true obtained higher packet delivery ratios (Fig. 10a), lower normalized routing loads (Fig. 10b), lower average packet latencies (Fig. 10c), and lower number of route discoveries (Fig. 10e). However, the average hop counts of the two methods are almost identical as can be seen in Fig. 10d.



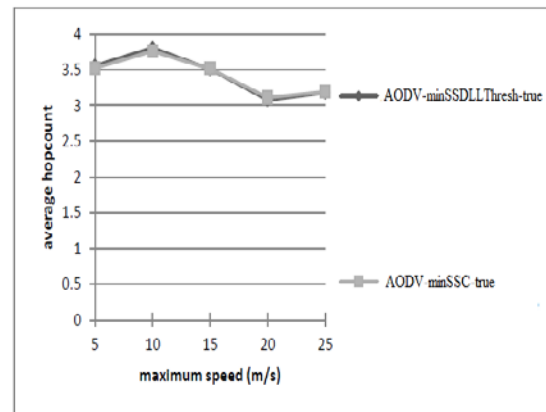
a. packet delivery ratio



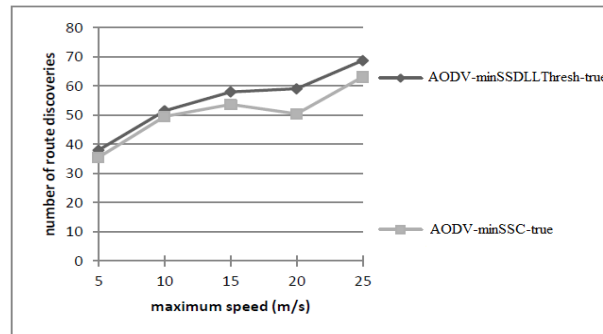
b. normalized routing load



c. average packet latency



d. average hop count



e. number of route discoveries

Fig. 10. Comparing AODV-minSSDLLThresh-true and AODV-minSSC-true

5. Conclusion

The hop count routing metric is commonly used in wireless sensor network routing protocols, but it does not perform well in mobile scenarios, the reason being the hop count routing metric favors lower hop count paths with long links over higher hop count paths with short links and longer links tend to break more easily than shorter links. In this paper, several methods for improving routing in mobile wireless sensor network were investigated. We first experimented with using link length information in conjunction with estimated path remaining lifetimes (AODV-SR,MPL) to improve network performance. Compared to purely using only the estimated path remaining lifetime as the routing metric (AODV-MPL), the AODV-SR,MPL method displayed better performance due to the higher stability of the discovered routes. However, this method may not find a path from a source to a destination even if many paths exist due to nodes dropping RREQs under certain conditions. To resolve this problem, another method is proposed - using a routing metric to assign penalty to long links that exceed a certain threshold length (SSDLLThresh-true). An extension of this routing metric that further takes into account node mobility was subsequently proposed (SSC). Both of these routing metric perform well and do not possess the deficiencies of the AODV-SR,MPL and AODV-MPL methods. When accurate information is available (by using GPS and other sensors), these routing metrics perform well. However, the true beauty of these routing metrics lies in the fact that they can even be used even without GPS and other sensor. From our investigation, we found that the proposed routing metrics are highly effective and they outperform existing routing metrics and the hop count routing metric even when used without accurate information from GPS and other sensors.

Acknowledgement

This project was supported by Introduction of high-level talents and overseas returnees scientific fund in Nanjing Forestry University (No.: GXL015) and 2015 University Students Practice Innovation Training Program (No.: 2015sjcx122).

References

- [1] J. Sun, Y. Liu, H. Hu, D. Yuan and A. L. Stability, "Link stability based routing in mobile ad hoc networks," in *Proc. of 2010 5th IEEE Conference on Industrial Electronics and Applications*, pp. 1821-1825, 2010. [Article \(CrossRef Link\)](#)
- [2] C.-K. Toh, "Associativeity-based routing for ad hoc mobile networks," *Wirel. Pers. Commun.*, vol. 4, no. 2, pp.103-139, 1997. [Article \(CrossRef Link\)](#)
- [3] W. C.-W. Tan, S. K. Bose, and T.-H. Cheng, "Power and mobility aware routing in wireless ad hoc networks," *IET Commun.*, vol.6, no.11, pp.1425, 2012. [Article \(CrossRef Link\)](#)
- [4] Y.Ko and N.Vaidya, "Location-Aided Routing (LAR) in mobile ad hoc networks," *Wirel. Networks*, vol. 6, no. 4, pp. 307-321, 2000. [Article \(CrossRef Link\)](#)
- [5] T. Lu and K. Feng, "Predictive mobility and location-aware routing protocol in mobile ad hoc networks," in *Proc. of GLOBECOM'05. IEEE Global Telecommunications Conference, 2005.*, p.5, 2005. [Article \(CrossRef Link\)](#)
- [6] M. Gerla, "IPv6 flow handoff in ad hoc wireless networks using mobility prediction," in *Proc. of seamless Interconnection for Universal Services. Global Telecommunications Conference. GLOBECOM'99. (Cat. No. 99CH37042)*, vol. 1a, pp. 271-275, 1999. [Article \(CrossRef Link\)](#)
- [7] H. Peng and L. Shao, "Energy Saving Routing Algorithm Based on Stable Zone in Mobile Ad Hoc Networks," in *Proc. of 2010 Third International Symposium on Information Processing*, no.1, pp. 181-185, 2010. [Article \(CrossRef Link\)](#)
- [8] W. Kim. "A reliable route selection algorithm using Global Positioning Systems in mobile ad-hoc networks," in *Proc. of ICC 2001. IEEE International Conference on Communications. Conference Record (Cat. No. 01ch37240)*, vol. 10, pp. 3191-3195, 2001. [Article \(CrossRef Link\)](#)
- [9] M. E. M. Campista, P. M. Esposito, I. M. Moraes, L. H. M. K. Costa, O. C. M. B. Duarte, D. G. Passps, C. V. N. de Albuquerque, D. C. M. Saade, and M. G. Rubinstein, "Routing metrics and protocols for wireless mesh networks," *IEEE Netw.*, vol. 22, no. 1, pp. 6-12, 2008. [Article \(CrossRef Link\)](#)
- [10] R. Draves, J. Padhye, and B. Zill, "Routing in multi-radio, multi-hop wireless mesh networks," in *Proc. of 10th Annu. Int. Conf. Mob. Comput. Netw. -MobiCom '04*, pp. 114, 2004. [Article \(CrossRef Link\)](#)
- [11] Y. Yang, J. Wang and R. Kravets, "Designing routing metrics for mesh networks," in *Proc. of IEEE Workshop on Wireless mesh Networks (WiMesh)*, 2005. [Article \(CrossRef Link\)](#)
- [12] S. Li and H. Qi, "Distributed Data Aggregation for Sparse Recovery in Wireless Sensor Networks," in *Proc. of IEEE International Conference on Distributed Computing in Sensor Systems, (DCOSS)*, 2013. [Article \(CrossRef Link\)](#)
- [13] Kuk-Hyun Cho and Min-Woo Ryu, "A Survey of Greedy Routing Protocols for Vehicular Ad Hoc Networks," *Smart Computing Review*, vol. 2, no. 2, pp. 125-137, 2012. [Article \(CrossRef Link\)](#)
- [14] Komal Zaman, Muhammad Shafiq, Jin-Ghoo Choi and Muddesar Iqbal, "The Life Cycle of Routing in Mobile Ad Hoc Networks: A survey," *Smart Computing Review*, vol. 5, no. 3, pp. 135-150, 2015. [Article \(CrossRef Link\)](#)
- [15] Zhibo Wang, Jilong Liao, Qing Cao, Hairong Qi, and Zhi Wang, "Achieve k-barrier Coverage in Hybrid Directional Sensor Networks," *IEEE Transactions on Mobile Computing*, vol. 13, issue.7, pp. 1443-1455. 2014. [Article \(CrossRef Link\)](#)
- [16] Shuguo Zhuo, Zhi Wang, Ye-Qiong Song, Zhibo Wang, and Luis Almeida, "A Traffic Adaptive Multi-channel MAC protocol with Dynamic Slot Allocation for WSNs," *IEEE Transactions on Mobile Computing*, vol. PP, issue .99, pp. 1-1. 2015. [Article \(CrossRef Link\)](#)

- [17] A. Moussaoui, F. Semchedine and A. Boukerram, "A link-state QoS routing protocol based on link stability for Mobile Ad hoc Networks," *J. Netw. Comput. Appl.*, vol. 39, pp. 117-125, 2014. [Article \(CrossRef Link\)](#)
- [18] C. Lal, V. Laxmi, and M. S. Gaur, "Performance analysis of MANET routing protocols for multimedia traffic," in *Proc. of 2011 2nd International Conference on Computer and Communication Technology (ICCT-2011)*, pp. 595-600, 2011. [Article \(CrossRef Link\)](#)
- [19] C. Wu, K. Kumekawa and T. Kato, "A MANET protocol considering link stability and bandwidth efficiency," in *Proc. of 2009 International Conference on Ultra Modern Telecommunications & Workshops*, pp. 1-8, 2009. [Article \(CrossRef Link\)](#)
- [20] E. Dutkiewicz, "A New Method of Selecting Stable Paths in Mobile Ad Hoc Networks." in *Proc. of IEEE International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing – Vol 2 – Workshops*, vol. 2, pp. 38 – 45, 2006. [Article \(CrossRef Link\)](#)
- [21] The Network Simulator – ns-2. [Online]. Available: [Article \(CrossRef Link\)](#)
- [22] T. T. Son, H. Le Minh, G. Sexton and N. Aslam, "A novel encounter-based metric for mobile ad-hoc networks routing," *Ad Hoc Networks*, vol. 14, no. 2, pp. 2-14, 2014. [Article \(CrossRef Link\)](#)
- [23] D.Vassis, G. Kormentzas, A. Rouskas, and I. Maglogiannis, "The IEEE 802.11g standard for high data rate WLANs," *IEEE Netw.*, vol. 19, no. 3, pp. 21-26, 2005. [Article \(CrossRef Link\)](#)
- [24] L. Villasenor-Gonzalez, "A Performance Study of the IEEE 802.11g PHY and MAC Layers over Heterogeneous and Homogeneous WLANs," *Ing. Investig. y Tecnol.*, vol. 8, no. 1, pp. 45-57, 2007. [Article \(CrossRef Link\)](#)



Yi-Han XU received his Bachelor degree in communications engineering from Huaihai Institute of Technology, China in 2007. He received his Master and Ph.D degrees in communications engineering from University of Malaya, Malaysia in 2010 and 2014, respectively. He is currently a associate professor in the College of Information Science and Technology, Nanjing Forestry University. His research field include multimedia applications, wireless sensor network, wireless and mobile communication and network programming.



Yin WU received his PhD degree from Nanjing University of Aeronautics and Astronautics, China, in 2013. He is currently serving as a Lecturer at the College of Information Science, Nanjing Forestry University, China. His research interests are wireless and mobile networks, MIMO wireless communication, design and test of electronic system.



Jun SONG was born in Jiangsu Province, China, in 1979. He received the B.S. degree and M.S. degree from the China University of Mining and Technology (CUMT), Xuzhou, in 2002 and in 2005 respectively. He received the Ph.D. degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, in 2014, all in electrical engineering. He is currently with the Department of information science and technology, in Nanjing Forestry University as an associate professor. His research interests include spectral estimation, array signal processing, and information theory.