(Ab-)Using DSR Route Information for Node Localization in MANETs

Boto Bako, Sergey Chapkin, Frank Kargl, Elmar Schoch

Ulm University, Institute of Media Informatics
{*givenname.surname*}@uni-ulm.de

Abstract. Unlike in conventional networks, nodes in mobile ad hoc networks (MANETs) usually take part in network maintenance and perform routing and diagnostics functions. Not only do they forward payload data but also send and receive topology information for routing purposes. This information may be used by malicious parties to intrude the location privacy of other participants.
In our previous work [1] we analyzed a scenario where an external party obtains complete topology information in a MANET. The results shown that such an attacker could achieve rather good results in localizing every node on the network. In this following work we analyze more realistic scenarios where attackers try to abuse the DSR routing protocol, to obtain topology information and derive position estimations of other nodes from that.
We present a localization approach that is based on a "hop to route length ratio" heuristic and show how these results compare to our previous findings. As a result we conclude that the accuracy that can be gained by only using DSR protocol information is rather restricted and only poses a minimal privacy threat.

1 Introduction

The emerging development of wireless communication technology has already brought up a new freedom in networking possibilities. Yet, the technology allows for even more application domains when wireless devices connect spontaneously and form a network dynamically. For instance, besides the concepts of mobile ad hoc networks (MANETs), wireless sensor networks (WSNs) are envisioned for monitoring purposes, and vehicle-to-vehicle networks (VANETs) are developed for safer and more comfortable driving.

However, wireless technology also introduces several security and privacy problems. One of these is location privacy. Due to the nature of wireless transmissions, a station can conclude that another node must be around in the technologically limited transmission range, as soon as it receives packets from that node. Therefore, the own position is revealed to any receiver in communication range with a certain degree of accuracy.

Basically, there are three types of potential location privacy violations that arise from immediate communication:

– *Identification and tracking in a mass of nodes:*
With a sectoral antenna, a receiver can locate the direction of the sender and can follow it without attracting attention.

– *Individual location profiles with a set of receivers:*
  When collecting communication samples at multiple receivers at different locations, individual traces of nodes passing by can be generated.
– *Social engineering:*
  Even if long-term identification of single nodes is not possible, one could use the information of communication samples in combination with time and location of the receiver to extrapolate data, e.g. on node density at certain times.

Such kind of data can be interesting to a number of people. As an example, stores could equip their premises with receivers to collect information about pedestrians using mobile communication devices. By collecting communication data of pedestrians, they would easily be able to extract where, when and how many pedestrians pass by and adapt prices accordingly. If mobile devices have a unique identifier, they could correlate this to a person using payment information. This could lead to extensive location profiles. Even if no accurate profiles could be gathered, simply the relation that a person has been at a certain location at a certain time might be interesting to someone. For instance, one could be suspected of a crime that happened in the vicinity, or the health insurance could raise the rates because they get to know about personal leisure time activities.

The scenarios presented so far all track persons in direct radio range and thus require a rather dense network of cooperating receivers, at least for collecting complete location profiles. However, when multi-hop ad-hoc routing is used and protocols like OLSR [2], AODV [3], or DSR [4] are employed, a form of indirect tracking might become possible. These routing protocols typically discover the complete or at least parts of the network topology to route packets from the sender to the destination.

By revealing topology information, the routing protocol introduces additional opportunities for attackers to generate location profiles. Having gathered information about the topology of the network, an attacker could use this as input for localization algorithms.

In [1], we investigated the potential accuracy of localization when powerful attackers know the received signal strengths of every direct link throughout the network. Under this precondition, using only a small number of anchor nodes ($< 5\%$) and under the assumption of low measurement errors, we proved that nodes can be localized with an accuracy mostly better than 20% of the radio transmission range.

In this paper, we alter the scenario to a more realistic one: the attackers can use only the information gathered through the DSR routing and forwarding process. So the attackers will only have information available which can be acquired by employing regular DSR mechanisms.

As the attackers only have a part of network topology, and no distance estimations, we analyzed existing approaches for range-free localization. In [5], Shang et al. propose using MDS-MAP approach in their localization algorithm and achieve good results, the error being $0.5 \cdot R$. However, they analyze networks with high node degree (12.5) and use the complete topology information. In our scenario only a part of the topology is known, and the node degree can be rather low (6). In [6] and [7] a propagation approach is presented, which runs in a distributed manner. Each node obtains an estimation of its position by assessing

the distance and the number of hops to anchor nodes, and then propagates this information to other nodes connected to it. The approach yields rather good results, although it also assumes knowing the complete topology (each node knows its direct neighbors). Our assumptions differ (not distributed calculation, partial topology information) substantially from those made in [5].

The next section gives a short overview on DSR. After that, we introduce our scenario and assumptions and esp. how DSR can be used to gather information about the network topology. Based on these – potentially incomplete – data, section 4 describes our the localization approach, before the analysis in section 5 shows, what accuracy an attacker can achieve. In section 6, a summary concludes the paper.

2 Dynamic Source Routing

DSR (Dynamic Source Routing Protocol) belongs to the class of reactive routing protocols designed specifically for use in multi-hop wireless ad hoc networks of mobile nodes [8]. The main idea of source routing is that each packet carries complete routing information (a list of intermediate nodes from source to destination). The source node determines the route of packets to the destination and intermediate nodes only relay the data. Moreover, the protocol allows for node mobility across the network as well as for the fact that the nodes can go off the network, moving out of the connectivity range of all their neighbors. Not only do nodes leave and join the network at random, but also paths between different nodes as well as hop-distances over those paths change constantly, leading to a very dynamic network topology. DSR also ensures that routes are loop-free, despite high topology volatility.

The main part of the protocol is route discovery, which runs reactively, i.e. when there is data to be sent. The source node floods a special Route Request packet (RREQ). In case a node that received a RREQ is not the destination node, it adds its own address to the RREQ's header and forwards the RREQ to all the nodes in its radio range. Thus, each RREQ packet carries a list of visited intermediate nodes. A soon the node with the destination address receives a corresponding RREQ packet, it sends a route reply packet (RREP) along the reverse path of visited nodes found in the RREQ. The RREP contains the complete path from source to destination. After receiving the RREP, the source node records it and sends the data along the discovered route.

In case a link along the route is broken due to changed topology, the forwarding node sends a Route Error (RERR) message back to the source node. The source node can then either use an alternative cached route or initiate a new route discovery.

3 Scenario and Assumptions

In contrary to the best-case scenario from our previous work [1], we are going to analyze how accurate node positions can be calculated in a realistic scenario where attackers gain topology information only from a routing protocol, DSR in this case. Therefore, the base of this scenario is a normally working MANET

with mobile nodes that may come and go, as well as move wherever they choose to. There are several cooperative attackers which behave as regular nodes. The attackers are mobile as well and can choose their positions in the network themselves. They are free to choose locations they deem to be most suitable for the fulfillment of their task. An additional assumption for this work is that attackers have the possibility to exchange information among each other. This allows them to share their positions and topology information they have with each other even if they become separated on the network because of broken links. This could be achieved e.g. by means of using a wireless access network like GPRS or by simply combining the information later during offline analysis.

In the beginning, attackers do not know anything about the network but gather their knowledge both by overhearing traffic such as data packets, RREQs, RREPs, and by actively sending RREQs to other nodes and collecting the following RREPs. They spread in that area to be covered, taking any strategic positions they may need. The further apart they are from each other, the more information about the network topology they might be able to gather.

There are two different ways attackers can gather topology information in this setup and also the purpose of the attack – localizing all nodes in an area or tracking only one single node – differs. Therefore three different scenario variations are investigated:

- Passive attackers: attackers only overhear packets (data, route requests and replies) from others. They don't actively interfere with network communication, but only passively collect information.
- Active attackers: in addition to passive overhearing, the attackers also actively send route requests to gather new information from the network. Route requests are sent to newly discovered nodes making it possible to gather more information about the network topology.
- Tracking: attackers explicitly track one known node and move themselves toward that tracked node, to improve accuracy.

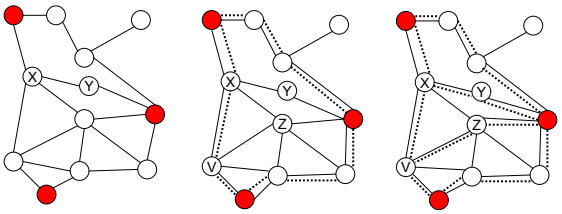

Fig. 1. 1: Sample network with attackers shown in red; 2: Dotted line is the current topology knowledge of the attackers; 3: Nodes Y and Z added to the known topology.

In the case of active attackers they start sending route requests to each other at the beginning (this applies also to the passive attackers case) and then to all nodes they have discovered in this way or which they have found by overhearing

other traffic. As the attackers exchange information all the time, they also know about nodes on all routes discovered by the other attackers, thus additional route requests to those nodes may result in even more nodes discovered. The actual sources for topology information are both received or overheard RREPs and overheard or forwarded data packets. They extract the route from such packets and examine it to enrich their own topology model and to find for new nodes to send RREQs to. Another source of passive information are route requests. They can provide the already filled up path of the route, which is then processed as above. The route requests may also provide new nodes for future RREQs. All attackers send RREQs to that node to discover additional topology information. Route replies, either overheard or forwarded, deliver similar information. In this case this is the complete route to some destination, which is in turn processed as described above.

Figure 1 illustrates how the attackers explore the initial network topology (left picture). First each attacker explores a route to each of his accomplices. The result is shown in the middle picture by dotted lines. Once the attackers have exchanged the information, the existence of all nodes located on dotted lines is known to the attackers. Nodes X and V are specially marked because they illustrate how even more nodes can be discovered. When the attacker on the right side sends a route request to nodes X and V respectively, he uncovers the nodes Z and Y. The resulting topology is shown in the right picture of Figure 1. There is only one node left uncovered. There is no more static information to extract from this configuration. The attackers then wait for network communications. It is worth mentioning, that the undiscovered node has little chance to hide. Should any node try to communicate with it, by first finding a suitable route (say it is node V) – the RREQ will be also received by an attacker. After this, the attackers learn about that standalone node and will send their own RREQs to it, thus annexing that last part of topology to their already acquired knowledge. This does not apply to links between nodes, however, as some of them can remain undiscovered.

The tracking attack scenario is the same as above, with the difference that all the attackers want to localize one special node on the network and they can move toward it to improve the evaluation. This scenario has its relevance as the following observation shows. Depending on different factors, the original scenario described could yield position evaluations (as will be shown in the analysis section) which are often not enough to obtain certain information not only about the exact location but also about visual identification of a node. This means that an attacker can localize the node and limit the search to some area but he or she can not tell which node exactly it is (from many others) if he or she would have a visual contact with the area.

In the next sections, an overview of the methods used is given and the localization algorithm used for these scenarios is described in detail.

4 Localization Approach

The input data for the localization algorithm are the information retrieved from the DSR protocol as described in the previous section. The algorithm uses heuris-

tics, based on predicting node positions with some probability, and using predictions which have high probability in iterative search. There is no upper limit of the runtime, as the algorithm tries to refine the estimation continuously (during the runtime it also processes any new information coming from analyzing DSR packets), until all the nodes receive rather good estimations. The algorithm can be aborted at any time, yielding a partial solution.

In order to perform well, it is important for the algorithm to start with "good" route paths between the anchor nodes. The higher the information quality of such a route path is, the more precise are location estimations. After the positions of some regular nodes could be estimated this way, these new positions are used for the positioning of other nodes. The question is how such route paths can be rated being "good" or "bad". The methods applied for rating routes is described in the next section.

4.1 "Hop to route length ratio" (HL) heuristics

There is a certain pattern how the number of hops on the route can be related to the geographical distance between the sender and receiver. From this relation, the path's quality can be derived, that is used by the positioning algorithm. This theoretical assumption has been validated by our simulations on obstacle free networks with uniformly distributed nodes. Besides, similar results were pointed out in the DV-hop method in [6]. There, a reverse value to "hop to length ratio" was used and also good results were achieved on networks with uniformly distributed nodes.

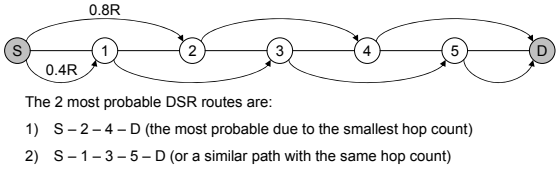

Fig. 2. Route from S to D will most likely include 3 or 4 hops, not 8.

First, we introduce the metric for route quality. When discovering a route, DSR chooses the route on which a node gets the fastest route response. As DSR RREQs are being flooded to all the nodes in the radio range R, it is most likely that the route with less number of hops will get the response faster than a route with more hops. Thus, the path with the lowest hop count is chosen with the highest probability. An example is shown in Fig. 2 with the two of the most likely routes from the source (S) to the destination (D). On networks with high node density (average hop being $< 0.5 \cdot R$), our simulations have shown that the quotient of number of hops and the route length is approximately $4/(3R)$, which means 4 hops for each 3 radio ranges. If this value is near 1, the nodes positions are very close to the direct geometric connection between source and destination (being attacker nodes when the algorithm starts, or already positioned nodes

later). This way, precise position estimations can be calculated. On the other hand, if this metric value increases (above $2/R$), the deviation of node's positions is high around the straight connection, thus no precise positioning is possible.

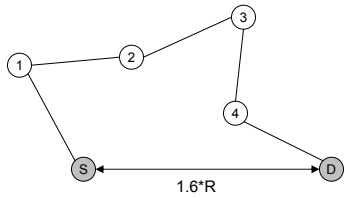

Fig. 3. With such hop to length ratio the geographical path of the route is unpredictable.

In Fig. 3, the anchor nodes reside at a distance of $1.6R$, and the route between them has 5 hops. Obviously, with $5/(1.6R)$ it is impossible to predict the positions of nodes with high accuracy. The only information accessible by the algorithm is that nodes are located around the line of the direct connection between the fixed anchor nodes (S and D). As the hop count is high, the spanning two dimensional space becomes huge, resulting in low positioning precision due to the many possibilities nodes can be placed around the direct connecting line. But approximately in the range between $4/(3 \cdot R)$ and $6/(3 \cdot R) = 2/R$ a good prediction about the geographical path of the route can be made. Independently, a similar result was obtained in [6]. For this work we experimentally estimated the $4/(3 \cdot R)$ - $2/R$ range to cut off the estimations which will most probably be inaccurate.

4.2 Derivation of node distribution along the route from the HL metric

The last section pointed out, how route path qualities can be computed. Now we describe how this path quality metrics are used to create a (partial) probability density function used by the algorithm. Our assumption was, that most nodes are located in a certain range from the direct line between the source and destination as shown in Fig. 4. This range obviously depends on the HL value. In this work we used the following relation for this range: $\sqrt[2]{R^2 - \frac{1}{HL^2}}$. With this relation, good results were achieved in our simulations. The probability distribution for this relation was quantified using Monte Carlo simulations. In these simulations, route paths with good HL metric were simulated and evaluated. As a result, we can say that each node lies within the range described in the formula above with a probability of 70%.

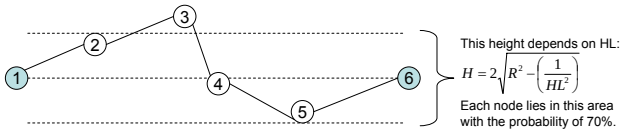

Fig. 4. Distribution of nodes along routs

4.3 Probability based position estimation

In this section, the core of our localization algorithm is presented. The idea is to determine for each node the area within which they lie with the highest probability. For this we use the following heuristics and probability distributions:

1. Probability distribution of nodes along routes with good HL metric, as described above.
2. Minimal distance heuristic: If there is no connection between two nodes, then it is more probable that the distance between them is higher than R.
3. Maximal distance heuristic: The length of one hop is <= R.

The algorithm first finds all routes with good HL metrics (HL in the range $(4/(3R), 2/R)$ and calculates the range of the most probable node distribution (1), defining the borders of this area. This area is further divided by additional lines according to maximal distance heuristic (3). This way, areas for a node that lie out of communication range get the probability 0, whereas areas within the range get higher probability. Generally spoken, by adding new borders, the areas become smaller and the probability values of overlapping areas are multiplied. In so doing, the ratio between probability values can be computed and is used to determine the area in which a node is located with the highest probability.

In Fig. 5 we see lines a and b defining the borders (1) for the route 1-5 and c and d the borders for the route 1-4 respectively. The circle represents the borders for the node 2 on those two routes (according to 3), and it is obvious that the maximal distance equals $R \cdot H$, where H is the number of hops to the respective node. For the node 2 H equals 1.

An example for the probability calculation for the introduced areas follows. For the node 2, the area in the circle gets a probability value of 100%, outside the circle 0% (according to 3). The area between a and b has an initial estimation of 70% inside and 30% outside, according to (1). The node 2 has additional borders c and d, according to (1), because it also belongs to the route 1-4. As we can see, the intersection of borders yields overlapping areas. For those areas the probabilities are being multiplied, this way we get the probability distribution for node 2. For example, the area with the highest probability (B) is in which the node 2 is placed in the picture. For this area the probability is computed as the product of three values. The first designates the maximum distance heuristic, the other two came from the HL-metric. So, the probability is calculated for this area as $P = 1 \cdot 0.7 \cdot 0.7 = 0.49$. For the other areas it is calculated similarly and the node is placed into the area with the highest probability value. Below a short pseudocode of the algorithm is introduced to give more details.

– 1. take one pair of attackers, A and B. Mark as used. If no unused pairs, go to 6.

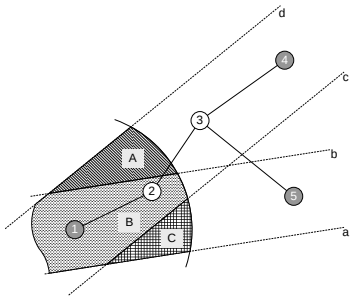

Fig. 5. Borders confining the areas on the plane where the node may be positioned.

- 2. calculate the hop to length ratio of the route AB. If it falls into the estimation range $(4/(3R), 2/R)$, obtain border estimations and apply them to all the nodes along the route.
- 3. for each node Z of the AB route: check all the routes between attackers which go through this node. For each route repeat step 2. This way some areas are excluded and other become smaller and get more exact probability values.
- 4. after each such check the position estimate of node Z improves due to new estimations from other routes on which this node lies. The minimal distance heuristic [1] is used to ignore the areas which lie very close to a node not connected with Z.
  After steps 2-4, any node which has only one area marked with a probability (all other areas marked with 0%) is considered resolved and is placed in the geometrical center of its area. Other nodes are placed into the area with the highest probability.
- 5. go to 1.
- 6. for each pair of nodes AB repeat the same steps as 1 through 4 for the attackers.
- 7. If not all the nodes are resolved then delete all "used" marks and go to 1, otherwise stop. Repeating the steps can improve the estimation because new information can become available over DSR, or nodes, resolved in the last run, can resolve the others. So the algorithm runs until all the nodes are resolved or is stopped.

5 Analysis

For this analysis, we concentrate on location accuracy in different networks, how it depends on the number of attackers and their placement. For simulations a network with 100 nodes was taken, the number of attackers varies as mentioned for each simulation. The nodes were placed randomly and uniformly, with mobility pattern "random waypoint". For assessing worst-case accuracy some non-uniform constellations have been taken. JiST/SWANS simulation package was used to run the simulation. The field size was set $15R * 15R$. The attackers are each time placed in the optimal way for them to achieve a good quality (distributed over the field, rather than concentrated in one place). This approach

has been chosen, because it was important to assess the threat to the location privacy of a normal node in the "worst case", which is of course the "best case" for the attackers.

Figure 6 gives an overview of a network with 100 nodes and a variable amount of attackers. The nodes kept their communication to the minimum during the simulation. The attackers were placed ideally to cover the most part of the network. There are three results presented for each case - the worst obtained accuracy, the best obtained accuracy, and the mean obtained accuracy. The worst and best accuracies are not peak values. These are mean values of accuracies near to the lower or the upper limit accordingly.

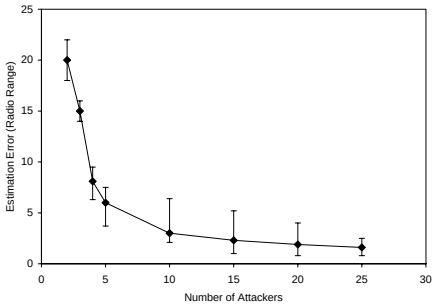

Fig. 6. Comparison of localization accuracy

We find that 5 attackers give a 6 time radio range mean accuracy, which can not be seen as useful localization result. At 10 attackers the estimation error drops to 3, and any further growth of the number of attackers does not bring any substantial improvements to the estimation. Moreover, if the number of attackers grows up to 30, which is 30% of the number of nodes, the mean error is still greater than 1.5 and closer to 2.0. This could be explained by the fact, that the more attackers are used, the less additional information is being obtained by adding extra attackers. Any further growth of the number of attackers will only bring small improvements to the estimation error, unless, of course the number of attackers goes up to 80% or 90%. As the realistic scenario could be 5% to 10% attackers (and even that can be considered as a lot) the numbers higher than 10% are solely of theoretical interest. The worst and the best accuracies differ little from the mean accuracy at small number of attackers, because quality of information is clearly not enough to obtain any useful results. However, with the growth of the number of attackers we see a greater deviation of best and worst accuracy values from the mean. The best accuracy value at 10 attackers is 2.1 and at 15 it is 1 radio range. This is explained by the fact that the actual distribution of nodes on the network plays a big role when the number of attackers is enough to collect the information. At some network constellations all heuristical predictions work almost perfectly. What has a negative influence on the accuracy is the fact that not all links can be obtained by attackers (although

this is also the case sometimes). The worst estimations are also explained by network constellations, in the first place by achieving constellations with unevenly distributed node degree.

In Figure 7 a diagram of dependency of the accuracy on the intensity of message exchange (traffic generation) by the nodes with silent attackers is shown. In this simulation which was run 10 minutes with low, medium, and high traffic volume, nodes were sending respectively 0.25, 1 and 4 messages per second to randomly selected nodes. The number of nodes was 100 and the number of attackers was 10, distributed in the optimal fashion. The attackers only send their route requests once in the beginning and then they rely only on overhearing the network traffic.

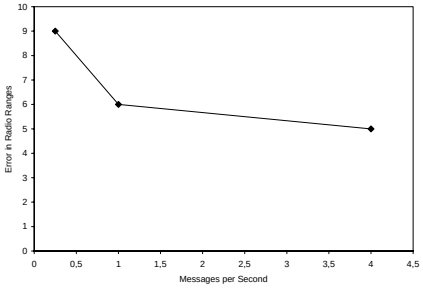

Fig. 7. Comparison of silent attack accuracy

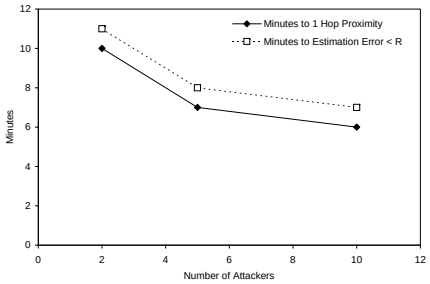

Fig. 8. DSR Localization with tracking

As it can be seen in Fig. 7, silent attack on a DSR network still yields some results, although they can hardly be used for individual tracking or location profiling. However, combined with other data about the position or moving patterns of a specific node, we could even use a result with an estimation error of 5 radio ranges. This use-case is important, as it shows that even if the attackers do not flood the network with route requests (which could be easily tracked down) and only rely on the overheard traffic, there are still results available (although of questionable quality).

The localization results of the third interesting use-case, which was briefly introduced in section 3, is shown in Figure 8. The test was conducted with 2 and 5 attackers, on a network of 100 nodes. Attackers maximum speed was limited to $2 \cdot R$ per minute. The node which was subject to tracking was placed 10 radio ranges away from the attackers, and the attackers this time were initially placed into the same corner of the network field, which is the worst possible constellation for them. The first diagram one can see in Figure 8 is the time in minutes, until the attackers reach 1 hop proximity of the tracked node. The second diagram is the number of minutes until the attackers obtain a position estimate with less than $1 \cdot R$ error. One can see that with these initial conditions the tracking time is decreasing rather slowly compared to the growth of the number of attackers. This could be explained by the fact that with this worst

possible attacker placement it does not play any significant role if there are 5 or more attackers. They will still have to move toward the victim. The time elapsed since reaching 1 hop proximity and locating the node with the estimate of less than one radio range is almost the same. This is explained by the fact that as soon as the node is in 1 hop proximity, 2 nodes are in most cases perfectly enough to track it.

6 Summary and Outlook

As our results show, a localization of nodes using only the information obtained from DSR routing is a cumbersome task. The accuracy is usually very bad, reaching a multitude of radio ranges. In case of IEEE 802.11 WLAN, this might be as bad as some hundreds of meters. On the other hand, there might be scenarios where this accuracy is enough, or where a higher accuracy can be reached due to special circumstances or geographic properties. Additional information about the nodes, like road maps in a car scenario might help as well as knowledge about movement patterns or personal habits of the node users.

In the general case however, attackers will probably have to invest additional effort to come up with a better tracking infrastructure that e.g. also measures signal strength or signal angles. To analyze how the accuracy of the localization will be influenced, if link quality gets communicated as a link metric in route replies could be an interesting work. Other future work is to investigate how accuracy increases when performing tracking over time (e.g. with the help of Kalman filter).

References

1. Chapkin, S., Bako, B., Kargl, F., Schoch, E.: Location tracking attack in ad hoc networks based on topology information. In: 2nd IEEE International Workshop on Wireless and Sensor Networks Security (WSNS'06). (2006)
2. Clausen, T., Jacquet, P.: Rfc 3626: Optimized link state routing protocol (olsr). `http://www.ietf.org/rfc/rfc3626.txt` (2003)
3. Perkins, C.E., Royer, E.M.: Ad-hoc on-demand distance vector routing. In: Proceedings of the 2nd IEEE Workshop on Mobile Computer Systems and Applications. (1999) 90–100
4. Johnson, D.B., Maltz, D.A.: Dynamic source routing in ad hoc wireless networks. In Imielinski, T., Korth, H., eds.: Mobile Computing. Volume 353. Kluwer Academic Publishers (1996) 153–181
5. Shang, Y., Ruml, W., Zhang, Y., Fromherz, M.P.J.: Localization from mere connectivity. In: MobiHoc '03: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing, New York, NY, USA, ACM Press (2003) 201–212
6. Niculescu, D., Nath, B.: Ad hoc positioning system (aps). In: Proceedings of GLOBECOM, San Antonio (2001)
7. Niculescu, D., Nath, B.: Ad hoc positioning system (aps) using aoa (2003)
8. Johnson, D.B., Maltz, D.A., Broch, J.: DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks. In Perkins, C., ed.: Ad Hoc Networking. Addison-Wesley (2001) 139–172