Annals of Telecommunications (2018) 73:787–802
https://doi.org/10.1007/s12243-018-0666-z

Metric anticipation to manage mobility in mobile mesh and ad hoc wireless networks

Sabrine Naimi[1,2] · Anthony Busson[3] · Véronique Vèque[1] · Ridha Bouallegue[2]

Received: 23 November 2017 / Accepted: 28 September 2018 / Published online: 11 October 2018

Abstract

In mobile ad hoc networks (MANETs), node mobility management is performed by the routing protocol. It may use metrics to reflect link state/quality. But, the delay between measures of the link quality and its integration in the route computation is very detrimental to the mobility management. Consequently, routing protocols may use lossy links for a few seconds leading to a significant performance deterioration. In this paper, we propose a new routing metric technique calculation which aims at anticipating link quality. Basically, the idea is to predict metric values a few seconds in advance, in order to compensate the delay involved by the link quality measurement and their dissemination by the routing protocol. Our technique is based on measurements of signal strength and is integrated in two classical routing metrics: ETX (expected transmission count) and ETT (expected transmission time). Validations are performed through both simulations and a testbed experimentation with OLSR as routing protocol. NS-3 simulations show that our metric may lead to a perfect mobility management with a packet delivery ratio of 100%. Experiments on a testbed prove the feasibility of our approach and show that this technique reduces the packet error rate by a factor of 3 in an indoor environment compared to the classical metrics calculation.

Keywords MANET · ETX · Metric · OLSR · Mobility · Anticipation · Testbed

1 Introduction

The recent evolution of wireless communication technologies and the emergence of mobile devices (laptops, smartphones, etc.) have allowed access to the network from anywhere, at anytime, without having to connect communicating devices to infrastructure. An undeniable advantage of these wireless technologies is the ability to be mobile while staying connected. Without any dedicated infrastructure, mobile ad hoc networks (MANETs) [1] allow communications among fixed and mobile devices. In MANETs, the nodes are connected among each other using direct wireless links. The node radio interface sets up a wireless link with a node according to the highest received signal strength it measures in its vicinity. The MANET is composed of nodes directly connected to each other by wireless connections and to reach a given node, messages are relayed by intermediate nodes. The underlying topology of such a network is changing with mobility of nodes. The basic idea in MANETs is that each node is both a router and a source/sink of data. Hence, the routing protocol [2] has to adapt itself to this new topology to find a new route across the network.

As all wireless communications, MANETs suffer not only from multiple impairments like bit errors and collisions but also link breakage due to interference or mobility. A link failure impacts directly the routing protocol as every route passing through this link is then broken.

In [3], an experimental study performed through a real 802.11-based MANET testbed shows that human mobility and interference lead to a very different link and route lifetime distributions. Thus, if no mobility, links break

✉ Sabrine Naimi
sabrine.naimi@isi.utm.tn

Anthony Busson
anthony.busson@ens-lyon.fr

Véronique Vèque
veronique.veque@l2s.centralesupelec.fr

Ridha Bouallegue
ridha.bouallegue@gmail.com

[1] Laboratory of Signals and Systems, Université Paris Sud - CentraleSupelec - CNRS, Orsay, France

[2] Innov'COM Laboratory Higher School of Communication - Tunisia, Tunis, Tunisia

[3] University Claude Bernard Lyon 1 - LIP (ENS Lyon - INRIA - CNRS - UCBL), Villeurbanne, France

quickly due to a noisy environment or remain stable for a long period. If link failures are only due to mobility, 90% of all links would still be available after five minutes, but in a long-term perspective, only 3% of the links would still be available with mobility interruptions only [4]. Vehicular ad hoc networks (VANETS) are an extreme case of mobility, as the vehicle speed is high and may vary a lot from one vehicle to another. In VANETs, it has been shown that routing protocols based on topology control are unefficient [5] because these protocols suffer from large end-to-end delay and low packet delivery ratio.

A routing protocol makes cooperation between all nodes in a distributed manner and is able to self-adapt to topology changes [6]: it consists in detecting link or node failures and to inform the other nodes of this topology change.

Mobility management is not so efficient in routing protocols [7], because the delay between the link failure event and the real consideration by the routing protocol and topology or routing table updating can be very long [8].

Let us detail how the routing protocol detects a link failure and propagates this event in the entire network. As mentioned, a node sets up a wireless link with its neighbor node having the highest received signal strength. The signal strength can decrease due to variable propagation condition or mobility.

In the proactive MANET routing protocols standardized by IETF [9, 10], the link management is performed at the network layer and is based only on the proper or bad reception of "hello" messages coming from the neighbors. Hello messages are sent periodically among neighbors and allow to perform the 1-hop and 2-hop neighbor discovery. This mechanism is employed for populating the local link information base and the neighborhood information base [9].

This limitation is very detrimental to the mobility management because the complexity and volatility of the radio layer make mechanisms implemented at the IP level unadapted to reflect link qualities in real time. This is caused by several fundamental reasons.

The radio environment may fluctuate at a different time scale (faster) than the frequency of control messages/probing messages allowing to sense the link. For instance, in OLSR [9], Hello messages are sent every 2 s by default. Statistics on the link quality requires several of these messages to be consistent. Link quality measures are then performed on a period that is unable to capture fast changes of quality. Moreover, these local measures need to be disseminated in the whole network to be taken into account (for link states approaches). These disseminations are also performed at a frequency that may introduce an important delay (e.g., topology control (TC) messages are sent every 5 s by default in OLSR) between the time at which the link state or the metric are estimated and its inclusion in the route calculation. Moreover, routing protocols tend to react to link breakages and do not try to prevent them. Nevertheless, specific mechanisms may be integrated in classical routing protocols to improve their performance in the presence of mobility. For instance, an efficient link detection algorithm (link sensing) can prevent failed links and disable them before being unusable [11, 12]. This technique has proved to be efficient but it has the disadvantage to disable the lowest quality links even if they are the only available links. A more accurate solution consists in associating specific metrics to each link to reflect their quality or their degradation. When a node is mobile, the degradation of the link quality leads to an increase of the associated metric and, thus, to a change of route with more stable links (with smaller metric values). When a link is designated as lost (route error in AODV [10] or the LOST_LINK state in OLSR, for instance), the routing process computes a new route but this broken link has been used for a few seconds and a certain number of losses have been necessarily experienced.

To reduce the different delays between the real quality of a link and its estimation or inclusion in the routing process, we think that one of the major problems is the quantity considered by the routing protocols to compute the shortest path from the source to destination. A metric associated with each edge/link is a numerical value that aims to express the throughput, the delay, or the quality of a link. The routing protocol uses these metrics to determine the best path for a source-destination pair. The simplest but imperfect metric is the hop count one where each existing link has a value of 1. The performance of a path is calculated from a function that takes into account the metric of the links forming the path [13]. A more accurate solution consists in associating specific metrics [14, 15] to each link that reflect their quality. When a node is mobile, degradation of the link quality leads to an increase of the associated metrics and, thus, to a change of route with more stable links (with smaller metric values).

In this paper, we present a method to anticipate the values of some routing metrics. The method has been partially published in [16]. Our main idea consists in using a prediction algorithm applied to received signal strengths to estimate future values of the metrics in order to compensate the delay involved by the link quality measurements and their dissemination by the routing protocol. Such anticipation techniques are used in infrastructure-based networks, but it has not been proposed to anticipate link metrics in ad hoc networks. To the best of our knowledge, metric anticipation to manage mobility in ad hoc networks has never been addressed in the literature. Our contribution is manifold:

1. We propose to perform routing in a MANET using a new metric calculation method to manage the mobility problem. Our metric is derived from the classical

metrics, in particular ETX [17] (expected transmission count) and ETT [18] (expected transmission time), but we combine their computation to our prediction algorithm.

2. In our simulation results, we compare the performance of this combined routing and prediction approach with the existing metric-based routing protocols from the literature.
3. We demonstrate the feasibility of our approach within an implementation of our solution deployed in a testbed. It allows us to validate our propositions in a real context.

The paper is organized as follows. In Section 2, we present related work on routing metrics used in ad hoc networks and prediction algorithm in infrastructure networks. Section 3 introduces our prediction method to anticipate routing metric values from measured signal strengths. We detail the method used to predict the signal strength, and a simple experiment to estimate its accuracy. Simulation scenarios and results are described in Section 4. Section 5 presents the implementation of our solution and results of our experimentations. We conclude in Section 6.

2 Related work

Usually, a network and hence, a MANET, is represented by a graph topology where vertices are the nodes and edges connecting two nodes represent the wireless links existing between these two nodes. The routing task is generally performed in a distributed manner using a routing protocol including several components. The first component consists of the choice and the measurement of edge cost or metric for each link. The second one is the route computation which consists in an algorithm to find the "best" route between a source and a destination. The third component specifies the way to exchange the topology information between nodes in order to update the graph topology and to maintain a global and consistent view of the network.

In this section, we introduce the notion of routing metrics, and present the ones considered in this paper. Then, we review several mobility prediction algorithms and metrics sensitive to mobility. We do not present works regarding routing protocols in mobile ad hoc networks as it is a well-known subject, and because our contributions focus on routing metrics. Nevertheless, the reader can refer to [2] for a survey of routing protocols.

2.1 Routing metrics

A metric associated with each edge/link of the MANET is a numerical value which aims to express the cost or the performance of a link. The routing protocol uses these metrics to determine the best path for a source-destination pair. The performance of a path is calculated from a function that takes into account the metric of the links forming the path [13]. The simplest function is the sum of the metrics, thus called additive metrics such as delay, or hop count. As we said, the hop count metric costs 1 for each link. The sum is then the number of hops in the path. More complex functions can also be the multiplication, or the minimum/maximum, of the metrics on the path. For instance, when the link metrics are the delivery ratio, its product on the path leads to the end-to-end delivery ratio. Throughput/bandwidth is an example where the minimum of the metrics is considered to evaluate a path (the metric is then the throughput of a link, and the minimum on the path is the end-to-end throughput).

The classical hop count metric is not able to reflect the link quality for which performances are rather characterized by the loss rate or the signal strength. Hop count tends to favor long-distance links that may offer poor received signal strengths, packet delivery ratio, and are more sensitive to mobility and link breakages as the involved nodes will be at the limit of the radio range of each other. Richer metrics have thus been proposed. They are generally not specifically defined to manage mobility but there are a few exceptions. Some metrics try to reflect the mobility of the nodes, for instance the average number of link breaks [19], or the link duration (LD) [20] that corresponds to the time where two nodes are within the transmission range of each other. These metrics aim at capturing the node mobility, and favor stable routes. But they are strongly dependent on the mobility patterns, useless when all nodes are mobile, and may be counterproductive when all nodes are fixed. Other metrics, not particularly related to mobility could nevertheless be efficient to manage mobility. These metrics are supposed to reflect the link quality and increase when a node recedes from a neighbor. The routing protocol may find better routes and switch from receding to more stable links. Therefore, these metrics have two important benefits: to manage mobility efficiently and to favor efficient link according to a performance criterion like the bandwidth, the loss rate, or the delay. Other metrics have been adapted to the wireless context, with cross-layered link interference estimation to select optimal paths [21], adapted to multi-channel wireless networks [22, 23] or cognitive radio [24].

2.2 ETX, ETT, and LD metrics

In this paper, we focus on some of the most popular metrics, ETX (expected transmission count) [17] and ETT (expected transmission time) [18], which have been shown among the most efficient [25]. We also consider these two metrics because the simplicity of their implementation is adapted to our proposal. Moreover, we present the link duration (LD) metric in detail as it will be compared to the other metrics in our simulations.

ETX Calculation of ETX uses the forward delivery ratio (**df**) and the reverse delivery ratio (**dr**) of the link. **df** is the probability that a data packet reached the destination (the extremity of the wireless link) successfully. **dr** is the probability that a data packet is received successfully. ETX is formally defined as follows:

$$ETX = \frac{1}{df * dr} \tag{1}$$

ETX evaluates the mean number of transmissions and retransmissions needed to send a data packet, as Eq. 1 is the mean of a geometric distribution with parameter $df * dr$. Consequently, routing protocols using the ETX metric tend to promote links with low loss rates. But, ETX does not distinguish links with different capacities.

ETT To solve this problem, the ETT metric was proposed. The ETT calculation is based on ETX but takes into account the data packet length L and the transmission rate B:

$$ETT = ETX * \frac{L}{B} \tag{2}$$

ETT can be seen as the time expected to transmit a packet of length L. It multiplies the number of attempts to send a packet (the ETX metric) by the time to transmit the packet physically on the link ($\frac{L}{B}$).

LD The link duration metric [7, 26] is calculated by measuring the life duration of a link between two nodes. For a given source-destination pair, the path with the maximum path duration is selected as the best route. The path duration is the minimum LD of all the links forming the path. The idea is to consider that routes with long path duration are more stable. In our simulations, we shall show that it is very dependent on mobility models and wrong in practice.

Metric classification Table 1 shows an overview of these metrics: input that allows their computation (input), how a route is chosen according to the link metrics (route selection), and the objectives.

2.3 Prediction algorithms

In MANET, the objective of prediction techniques is to better route packets and to avoid packet losses. A survey on the different prediction techniques can be found in [27]. They are classified as a function of the available information (geographic, at the link or physical layer, etc.). A prediction algorithm consists of a time series of data in one hand and, in the other, of an estimation method to analyze the data. The time series is composed of n samples taken at regular intervals. The nature of sampled data varies a lot not only depending on the intended applications [28], like node positions [29, 30], speed, and movement direction [31] but also signal strength [32], transmission power [33], and link expiration time [34]. The main estimation or prediction techniques are autoregressive (AR) modeling [30], regularity-pattern recognition algorithms, motion prediction algorithms based on a model of human movement behavior, Kalman Filters, Learning Machine, genetic algorithms, neural networks [35], or fuzzy methods [32]. The methods based on movement behavior models and pattern recognition fail in the case that nodes change in an unpredictable manner (a driver can change from a vehicular mobility to a pedestrian one) or the node behavior is not known or mixed. In [34], a mobility prediction method is presented for estimating the expiration time of the wireless link between two adjacent ad hoc nodes as a way to enhance various unicast and multicast routing protocols. The route expiration time is estimated as the least of the link expiration time values of all links on the route. Based on this prediction, routes are reconfigured before they disconnect. The estimation of the link expiration time is done using the node positions, their speed, and their moving direction.

The paper [29] uses extreme learning machines (ELMs) to predict not only the future location of nodes based on node coordinates but also the speed and movement direction angle. Another deep learning scheme is used in [36] to classify users into clusters as a function of their mobility

Table 1 Metrics properties

Metrics	Input	Route selection	Objectives
ETX	Packet error rate	Route that minimizes the sum of the metrics	Minimize the number of transmissions
ETT	Packet error rate Link capacity Mean packet length	Route that minimizes the sum of the metrics	Minimize end-to-end delay
LD	Link lifetime	Route that maximizes the minimum of the metrics	Choose oldest lifetime path

pattern and to optimize handovers. In [30], the authors focus on neighbor discovery and its hello protocol and propose ARH (autoregressive hello protocol) where each node predicts its neighbor mobility and position using an autoregressive model, based on historical location reports; it also predicts its own position using position samples in the same way. This protocol adaptively adjusts "hello" message rate to the optimal value according to time-varying node mobility. The effectiveness of this protocol is shown by simulations.

In [31], the proposed technique uses movement history and existing concepts of genetic algorithm to remove outliers. In a second phase, an adjacency matrix is obtained that calculates the predicted direction of each node using vector calculations and force directed graphs.

In [37], an adaptive mobility prediction method is proposed. It predicts the future distance of two neighboring nodes using automaton where a node predicts its own future position given its current position, speed, and direction.

All these techniques have been designed for ad hoc networks to predict future node location or the distance between them. But generally, nodes do not know their positions as they are not systematically equipped with localization devices or localization may be lost for a while. Moreover, mobility is often unpredictable (due to abrupt direction change for instance). But, the main problem is due to the fact that the node locations and movements do not directly give or predict link quality. Link quality is not only linked to the distance between nodes but is also affected by the environment where the node evolves (city, building, etc.) and complex radio phenomena (propagation, fading, shadowing, etc.). Therefore, the topology or the future topology (as given by prediction algorithms) does not give sufficient information to perform efficient route calculations as it should depend on the link qualities. Mobility prediction algorithms are thus a solution to potentially detect future link breakages, but must be necessarily combined with a link-sensing algorithm able to assess link quality through a given metric. But, we shall show that sensing algorithms are also able to predict future link quality from a local history of measures. Link qualities and their respective metrics can then be anticipated and directly taken into account by the routing protocol. Therefore, we think that it is more appropriate to predict future link quality through local measures, for instance from regular measures of the signal strength, than predicting geographical node location. We develop algorithms based on these assumptions in the next section.

3 Metric anticipation algorithm

The basic idea of our algorithm to anticipate the value of routing metrics is to forecast future signal strengths from local measures and to deduce future frame error rate (FER). FER is defined here as the probability that the transmission of a frame fails. It does not take into account retransmissions. While focusing on ETX and ETT metrics, the above method can however be applied to any routing metrics based on FER. The main algorithm of our method is followed by a presentation of its different components/functions: mapping between the signal strength and the frame error rate, the anticipation technique, and insights about the choice of the algorithm parameters. Our algorithm relies on existing routing control packets and potentially data packets. It does not introduce any protocol overhead. An array with the last signal measurements is just required at each node to anticipate the signal values.

3.1 Metric anticipation

ETX and ETT metrics are used in their classical forms when the link quality is good and are increased or decreased according to our signal prediction when the link quality is lower than a predetermined threshold. The algorithm is presented in algorithm 1. It considers the ETX metric, but we could consider ETT as well. The new metric is denoted ETX_ANT (respectively ETT_ANT). The algorithm is applied at each control packet reception. Upon a reception, the receiver measures the *LINK_QUALITY*, either RSSI, SINR (signal to interference plus noise ratio), or any quantity reflecting the link quality.

Algorithm 1 ETX_ANT ALGORITHM

```
void ETXcalculation()
begin
    /* The different variables are updated
    at each HELLO reception                         */
    Update $LINK_Q$ with the last Hello measurement;

    if $LINK_Q > TH_Q$ then
        /* The link quality is good or is
        improving: we do not anticipate             */
        $ETX\_ANT = COMPUTE\_ETX()$;
    else
        /* The quality is under the $TH_Q$
        threshold: we anticipate                    */
        $SIGNAL\_PRED =$
        $SIGNAL\_PREDICTION(TIME)$;
        $FER\_PRED = FER(SIGNAL\_PRED)$;
        $ETX\_ANT = 1/(FER\_PRED * d\_f)$;
```

The LINK_QUALITY variable is compared to a predefined threshold denoted TH_Q. If it is greater, we compute the ETX metric with the classical algorithm through the COMPUTE_ETX() function. If the link quality is less than TH_Q, we anticipate the value of ETX. First, the signal strength is predicted with the SIGNAL_PREDICTION(.) function using a past measurement history of signal strength values. The argument of this function, the parameter TIME (expressed in seconds), corresponds to the prediction time (the number of seconds we anticipate the signal strength/metric).

Then, we map this future signal strength to an expected FER with the FER(.) function. It allows the node to estimate the d_r parameter of the ETX metric (see Eq. 1). d_f is computed on the other extremity of the link (predicted or not depending on the link quality), and obtained through routing control packets. The ETX_ANT metric is then computed from these anticipated values.

3.2 Relationship between FER and signal strength

Anticipation is made within a prediction mechanism based on the history of link quality measurements. ETX and ETT metrics are mainly based on the FER of the wireless link. Therefore, it is this quantity that we try to anticipate. But, we do not use previous FER measures since it behaves as a step function. Indeed, wireless cards adapts their modulation and coding rate to keep a reasonable FER. If we observe FER when the signal is degrading, it is generally kept at a low level until the link reaches a quality threshold from which FER increases suddenly. It is thus difficult to anticipate FER with such a behavior, whereas the received signal strength is strictly decreasing/increasing as the nodes are moving and is directly related to FER. It is thus easier to forecast a trend for signal strengths than for FER.

A node collects and stores signal strengths of the packets exchanged with its neighbors. This quantity is generally available through the RSSI with 802.11 and 3G/LTE cards. In our implementations, these measures are collected only for control packets as the routing daemon may not be informed of the reception of data packets. But obviously, measures from data packets can be utilized as well when available.

We assume that each node has a table that maps signal strength to FER. This table can be obtained from the network card manufacturer or inferred through a set of measures performed before the ad hoc network deployment. This function, used to map a signal strength to a FER, has been denoted $FER(.)$ in our algorithm.

3.3 Signal strength prediction algorithm

Several methods have been proposed to predict signal strength [38]. Most of these methods assume a particular radio environment, for instance through a known path-loss or fading. We consider instead the linear regression method proposed in [39]. This choice has been motivated by the fact that this mechanism has a low complexity and does not rely on a particular radio context. It is worth noting that it is the anticipation mechanism that we have implemented in our simulations and experiments. But our proposition does not rely specifically on this mechanism, and any other prediction algorithms can be considered.

3.3.1 Linear regression method

With a linear regression method, a history of signal measurements, denoted $S = \{s_1, s_2, .., s_n\}$, is stored in a table. It contains the last n measurements updated at each new reception. The different times at which they are collected are stored in the vector $T = \{t_1, t_2, ..., t_n\}$. A linear regression model with the form $S = a + bT$ is used to fit the data. The future signal strength is then estimated as:

$$\hat{s}_{n+p} = a + bp \tag{3}$$

where p is the step of the prediction. It is related to the parameter TIME in our algorithm (given in algorithm 1). a and b are given by:

$$b = \frac{\sum_i s_i t_i - \bar{S}.\bar{T}}{\sum_i t_i^2 - n\bar{T}^2} \tag{4}$$

$$a = \bar{S} - b\bar{T} \tag{5}$$

$\bar{S}$ (respectively $\bar{T}$) is the mean of vector S (respectively T).

The authors of [39] used a dynamic window (dynamic size of the vectors S and T) to adapt to eventual abrupt changes. During these sudden changes, “old” data may not reflect the current trend. The reduction of the window size is thus necessary to discard measurements that are no more pertinent.

The size of the history window is initially set to a default value and changes according to errors observed in the prediction process. If the prediction error is greater than a threshold, the size of the window is immediately reduced. However, if the error does not exceed the threshold, the size of the window is increased until it reaches its maximum size.

3.3.2 Experimentations to evaluate the prediction algorithm

In our first set of experiments, we have just validated and evaluated the prediction signal mechanism presented above [39]. We will now test our own implementation. We configured two laptops in ad hoc mode under Linux (Ubuntu 12.04 LTS). They were equipped with two external USB wireless cards. We performed this evaluation in the outdoor garden of our laboratory where the two laptops were in line of sight of each other. The mobility of the two mobiles was as follows. First, both computers moved in opposite directions at a pedestrian constant speed (for 20 s); then, they stay more or less at the same distance (between seconds 20 and 90) before approaching a few seconds and moving

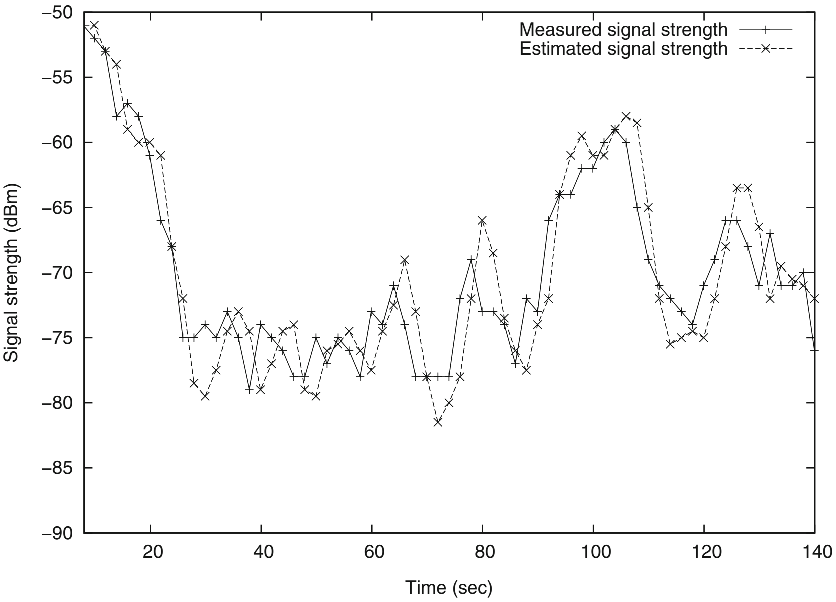


Fig. 1 Comparison of the measured signal strength and the prediction (made 2 s in advance)

away for a second time (between seconds 103 and 120). This particular mobility pattern aims at evaluating the prediction algorithm when there are clear movements in the signal strength (nodes approaching or moving away) and when the signal oscillates (the distance between the two PCs does not vary much).

In Fig. 1, we plotted the measured signal and the estimated ones (2 s in advance, i.e., TIME = 2 s). In the figure, the real signal is shifted of 2 s in order to compare easily the prediction and the measured signal. The figure clearly shows that this mechanism successes to predict signal strengths when nodes are approaching and moving away. As expected, when the signal strength oscillates, the prediction is less accurate. Several similar experiments have been performed and led to the same behavior.

3.4 Design parameters

Before performing the evaluation of our proposal, we need to tune the parameters TIME and TH_Q.

$TIME$ must be set according to the time needed to disseminate a new metric values in the network and the time to process it (routes computing, etc.). Therefore, it strongly depends on the used routing protocol. In the simulation setting, it is set as the frequency where control messages, that contains the metrics, are generated by the nodes. The other parameter, TH_Q, must be set in such a way that the prediction begins at least TIME seconds before a link breakage. Therefore, a fine tuning of this parameter assumes the knowledge of the radio environment and the maximum relative speed of the mobile nodes (denoted S_R hereafter). In our simulations, this information is known. In case of a real deployment, it can be inferred beforehand through experimentations. We give here a method to tune this parameter when the path-loss function is known. The path-loss function PL(distance) gives the expected receiving power with regard to the distance. But, any other method, more adapted to the deployment context can be used instead.

One way to tune our anticipation algorithm is to guarantee that a link breakage will be detected and taken into account by the routing process before it actually breaks. A link with a signal strength that would reach the physical threshold of reception, denoted TH_R, should be thus anticipated TIME seconds in advance. In other words, the threshold TH_Q must be set in such a way that when the node speed is S_R, it takes TIME seconds to pass from a signal strength of TH_Q to TH_R. We get:

$$TH\_R \geq PL\left(PL^{-1}(TH\_Q) + S\_R \times TIME\right) \tag{6}$$

where $PL^{-1}(TH\_Q)$, and $PL^{-1}(TH\_Q) + S\_R \times TIME$ are the distances at which the threshold TH_Q and TH_R are reached respectively. We inverse this equation to obtain the minimum value of TH_Q:

$$TH\_Q \geq PL\left(PL^{-1}(TH\_R) - S\_R \times TIME\right) \tag{7}$$

In this section, we have introduced the main algorithms we propose to anticipate the metric values, the prediction algorithm we use, and some results obtained from experiments.

4 Simulation

In order to validate and evaluate the performances of our proposition, we have conducted extensive simulations carried out under different scenarios.

4.1 Simulation scenarios

We performed simulations using NS-3 (Network Simulator version 3 (http://www.nsnam.org/)). We used the optimized link state routing protocol (OLSR) [9] to integrate our solution. OLSR is a proactive link-state protocol for which it was easy to take into account different routing metrics. Also, the new version of OLSR (OLSRv2 [40]) has been designed to take into account metrics more elaborated than the hop count. In OLSR, there are mainly two types of control messages: (i) Hello messages that are exchanged periodically (with a period of HELLO_INTERVAL) at one hop and contain local information on the neighborhood, (ii) topology control messages (TC) which are disseminated periodically in the whole network (with a period of TC_INTERVAL), and contain links states information (link, nodes, and metrics). We changed the format of these messages to include our metric value. Moreover, we changed the algorithm used to update the routing table. We implemented the Dijkstra algorithm instead of the default algorithm of OLSR, optimized for hop count metric. The rest of the OLSR routing protocol has not been modified.

We compare our approach (ETX_ANT and ETT_ANT) with the classical metrics ETX, ETT, and LD. LD metric is defined by the difference between the last time that the link was symmetric as managed by OLSR and the time at which the TC is sent. The fixed parameter values used for the simulations are given in Table 2. The Minstrel Wi-Fi manager is the algorithm that allows a node to compute an appropriate physical transmission rate (modulation and coding) with a neighbor according to the current radio environment, e.g., FER or signal over noise ratio.

In all simulated scenarios, we use the log distance path loss model [41] as the radio propagation model. We combine this function with a random variable S modeling the channel randomness. S follows a log-normal distribution with zero mean and different standard deviations σ (σ^2 ranges from 0 to 25). The received signal strength is then $PL(d) + S$ (expressed in dBm). The parameter TH_Q is set accordingly using Eq. 7 for a speed of 60 km/h for the chain scenario. The parameter TIME is equal to 2 s. It corresponds exactly to the TC period.

Three quantities are considered to evaluate the performance of our anticipated metrics and to compare it with the other metrics: (i) packet delivery ratio defined as the number of received packets divided by the number of sent packets, (ii) throughput defined as the mean number of received bits per second, (iii) the end-to-end delay. In all curves, each point is the average of 20 simulations and is shown with a confidence interval of 95%. It may appear that the confidence interval lengths are smaller than the symbols used in the plots, making them barely visible. We have simulated three topologies, that corresponds to different possible applications of ad hoc networks:

- **Chain scenario (Fig. 2a).** The first topology considered is a chain of 11 static nodes which are separated from a distance of 100 meters. A mobile node moves at constant speed along this chain. This topology is a trivial case that allowed us to test the efficiency of our algorithm, and where results are easy to interpret. The source is the mobile node. It transmits data packets at regular intervals (CBR traffic). The destination is the first node in the chain. We vary the speed of the mobile node from $10km/h$ to $70km/h$. The source moves and the application starts after a few seconds to let the time to OLSR to fill the routing tables. The simulation stops when the mobile node arrives at the end of the chain. This topology may emulate a network where the mobile is a train, a metro, or a vehicle and where it has been more convenient to deploy a mesh network than a wired one along the road or railway.
- **Mesh Network Scenario (Fig. 2b).** In this scenario, we consider a mesh network modeled through a grid topology of nine fixed nodes. Six nodes are mobile following the random waypoint model (RWP) [42] with constant speed (the speed is the same for all nodes and for each trajectory of the RWP). In the simulations, we select one source and destination chosen randomly among the mobile nodes.
- **Ad hoc network scenario (Fig. 2c).** In the third scenario, we considered an ad-hoc network where 25 nodes are mobile and are randomly distributed in an area with a size 300 × 300 m^2. Mobility model, speed, and selection of the (source, destination) pair is the same as in the mesh scenario.

4.2 Simulation results

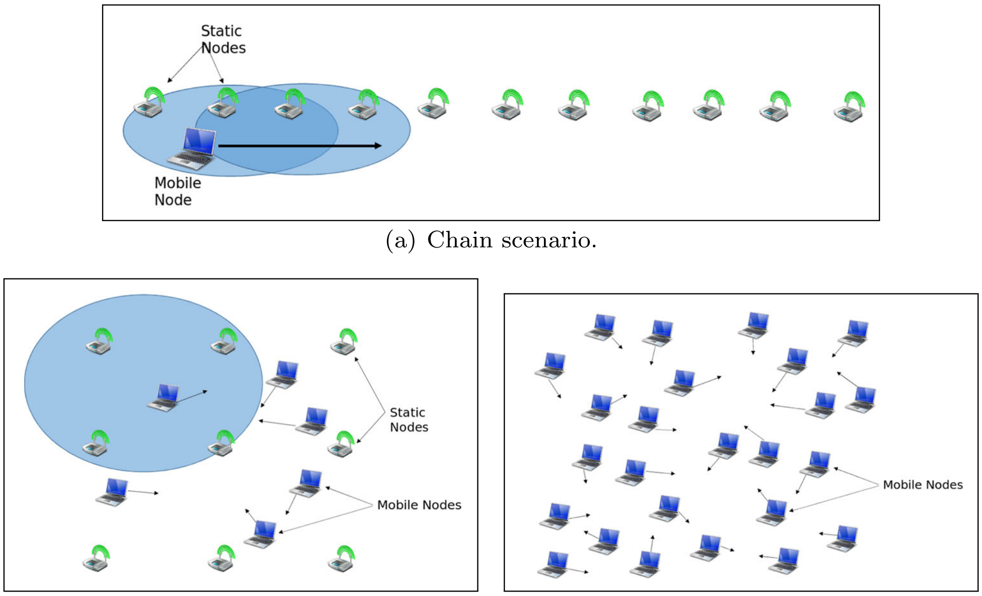
Results on the ETX prediction The quality of the prediction of the ETX metric has an important impact on the performance of the routing protocol. The anticipation occurs approximately for 19%, 23%, and 26% of the times in the chain, mesh, and ad hoc scenarios respectively. This proportion is quite insensitive to the mobile speed as it is more related to the ratio between the size of the region where the metric is anticipated and the whole radio range.

In Fig. 3, we plot the difference between the anticipated ETX value and the real one obtained $TIME$ seconds later. The speed of the mobile stations is 10 km/h in average.

Table 2 Simulation parameters

Parameters	Numerical values
Simulation time	62 [s]
Window size (chain scenario)	1200 × 200 [m^2]
Window size (mesh and ad hoc scenario)	300 × 300 [m^2]
Number of nodes (chain scenario)	12
Number of nodes (mesh scenario)	15
Number of nodes (ad hoc scenario)	25
Propagation model	Log distance propagation loss model
Fading (log-normal)	$\sigma^2 = 0, 4, 25$ mean = 0
Wireless technology	IEEE 802.11a
Traffic type	Constant bit rate (CBR)
Packet size	1024 bytes
Wi-Fi manager	Minstrel_Wifi_Manager
HELLO_INTERVAL (OLSR)	0.25 s
Anticipation time (TIME)	2 s

Fig. 2 The three different scenarios. Blue circles represent the radio ranges of the static nodes

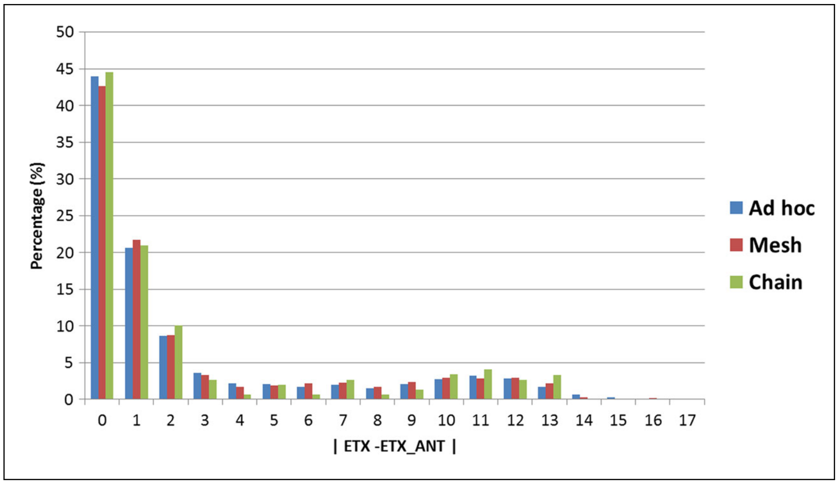


(a) Chain scenario.

(b) Mesh scenario.

(c) Ad-hoc scenario.

Fig. 3 Distribution of the errors computed as the difference between the anticipated ETX value and the real value: $|ETX - ETX|$. The speed for the three scenarios is 10km/h



It corresponds to the worst case as for greater speed, the error decreases. Indeed, the distance traveled by the mobile station between the reception of two consecutive control packets increases with the speed, and the trend of the signal may be easily determined. Also, the anticipated ETX leads more frequently to the ETX limit values (equal to 1 for instance when the signal is improving). The ETX reaching also this limit value $TIME$ seconds later the error is almost nil.

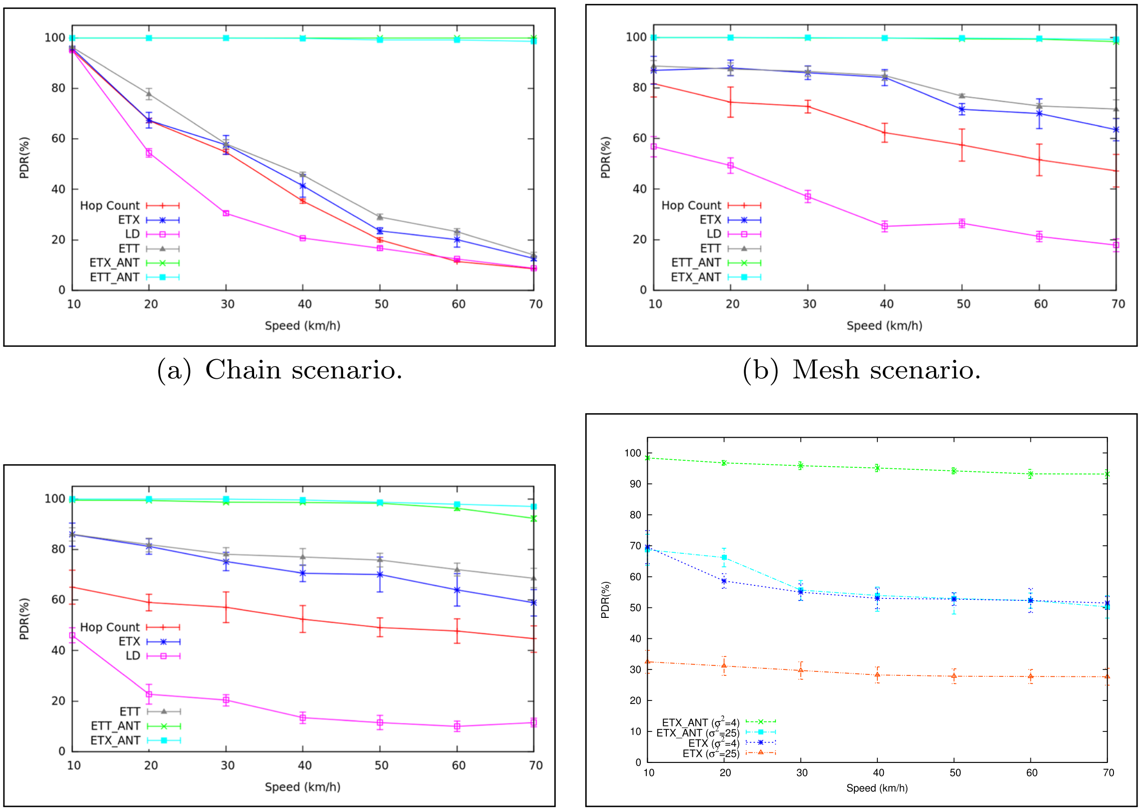
In the figure, we observe that the error is very small with approximately 45% of errors less than 1 and 80% less than 5. Nevertheless, the radio environment is well known and stable as it corresponds to the one of the simulator. The parameterization of the anticipation is then perfect.

Results on packet delivery ratio PDR is the quantity that reflects the most the ability of our proposition to manage mobility as loss of connectivity for a mobile node necessarily leads to packet losses. Figure 4a–c shows the PDR when using different metrics (hop count, LD, ETX, ETT, ETX_ANT, and ETT_ANT), varying the speed of the mobile node from 10 to 70 km/h, and without fading ($\sigma^2 = 0$). In these figures, the PDR is computed as the ratio between the number of packets received by the destination and the total sent. The source rate is 10 packets per second. It has been chosen in order to avoid any congestion and to evaluate the capacity of our proposal to find routes with a low error rate in presence of mobility.

For the chain and mesh scenarios, the PDR with ETX_ANT is 100% for all speeds. It stays greater than 97% for the ad hoc scenario. ETT_ANT presents a smaller percentage, but it is still greater than 92% in the worst case (ad hoc scenario with a speed of 70 km/h). The lower performance of ETT_ANT is due to the Wi-Fi rate, managed by the Wi-Fi manager (parameter B in Eq. 2), that is not predicted in our algorithm, and may lead to inaccurate predictions. These results empirically prove that node mobility is perfectly managed with our anticipated metrics for these scenarios.

The other metrics, and in particular the LD metric, have a PDR that drops with the node speed. LD selects paths with the longest duration which explains why it is the worst metric. Indeed, rather than privileging new links, the LD metric chooses the oldest.

To show the impact of fading, we plot in Fig. 4d the metrics ETX and ETX_ANT for the ad hoc scenario and for different fading value (we vary the standard deviation σ^2). We observe that performances degrade when the variance



(a) Chain scenario.

(b) Mesh scenario.

(c) Ad-hoc scenario.

(d) Ad-hoc scenario with fading.

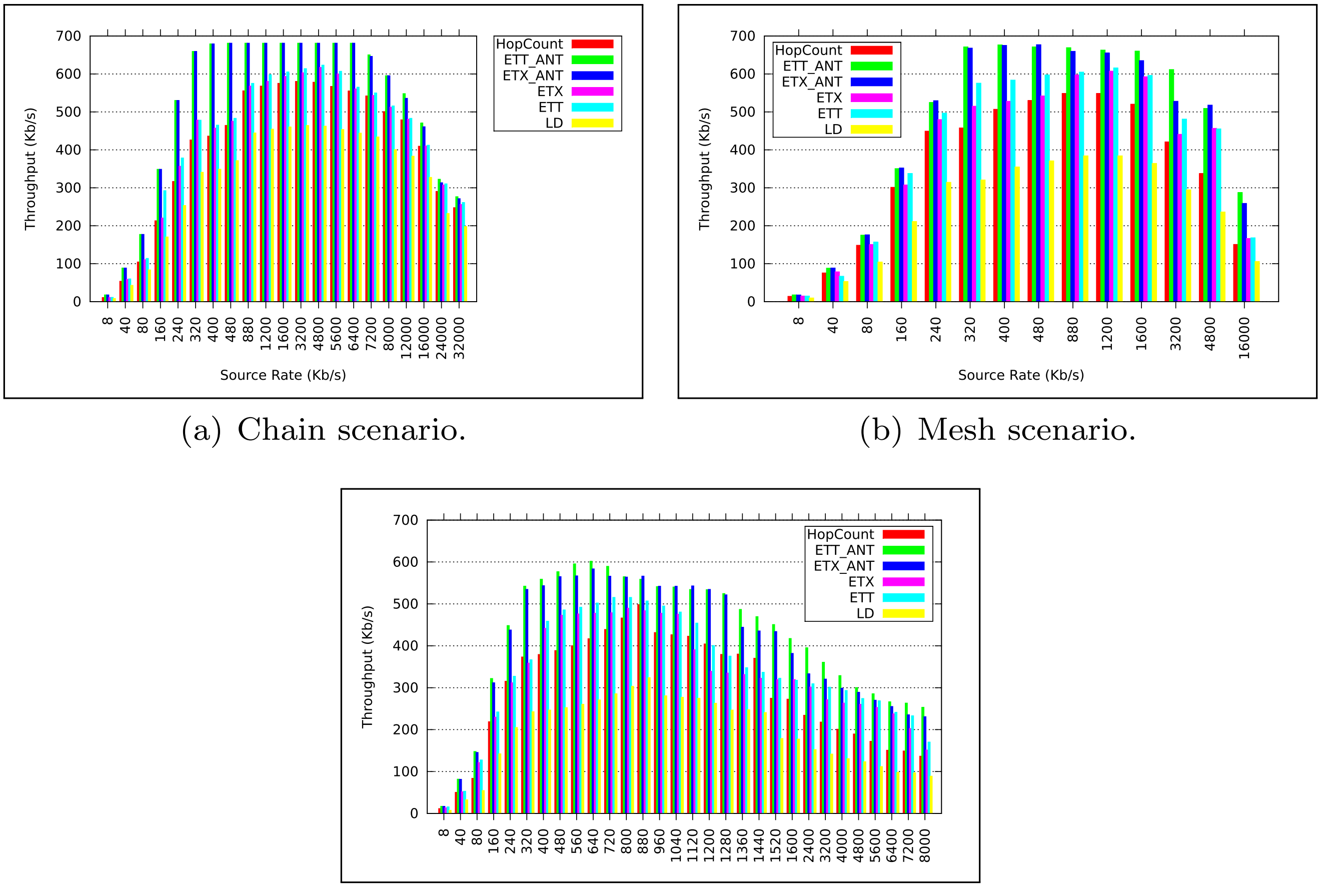
Fig. 4 Packet delivery atio for the three scenarios as function of the speed of the mobile stations. For the ad hoc scenario, PDR with and without fading is considered

increases whatever the metric. With small fading ($\sigma^2 = 4$), PDR with ETX_ANT remains greater than 95% showing that the mobility is well managed. For $\sigma^2 = 25$, the PDR is at most 70%. But, approximately 25% of the packet losses are due to the fading itself and not to mobility (it is the percentage that we observed when nodes were not mobile). This corresponds to a worst case scenario where fading is very important, and where by definition such randomness is unpredictable. But, PDR for the ETX_ANT metric stays more or less constant with the mobile speeds showing that even with high fading it still predicts properly the metrics. By lack of space, these results have not been shown for the previous scenarios (chain and mesh), but they present exactly the same behavior.

Results on the throughput The throughput is computed as the number of bits received by the destination over the simulation time. The speed of the mobile nodes is fixed, equal to $25km/h$.

Figure 5a–c show the throughput as the source transmission rate increases. In all scenarios, the measured throughput for ETX_ANT and ETT_ANT outperforms the one given by the other metrics. As ETT_ANT takes into account link capacities, it gives better results, but stays close to the throughput of ETX_ANT. The other metrics do not reach the maximum given by our anticipated metrics since they always experienced an important amount of packet losses. For the three scenarios, we observe a degradation of the throughput when the source rate reaches a certain threshold. It happens when the ad hoc network becomes saturated leading to losses of routing control packets and routing errors. Nevertheless, even in this congested state, our anticipated metrics offer better performance than the classical ones where routing errors caused by the nodes mobility and congestion cumulates.

Results on the end-to-end delay The mean end-to-end delay is shown in Fig. 6a–c for the three scenarios. In chain and mesh scenarios, the end-to-end delay when using the anticipated metrics is insensitive to the mobility. For the ad hoc scenario, it increases, but stays significantly lower than the delay obtained with the other metrics. For the classical metrics, the explosion of the delays is caused by the use of deteriorating links by the routing protocol. When these

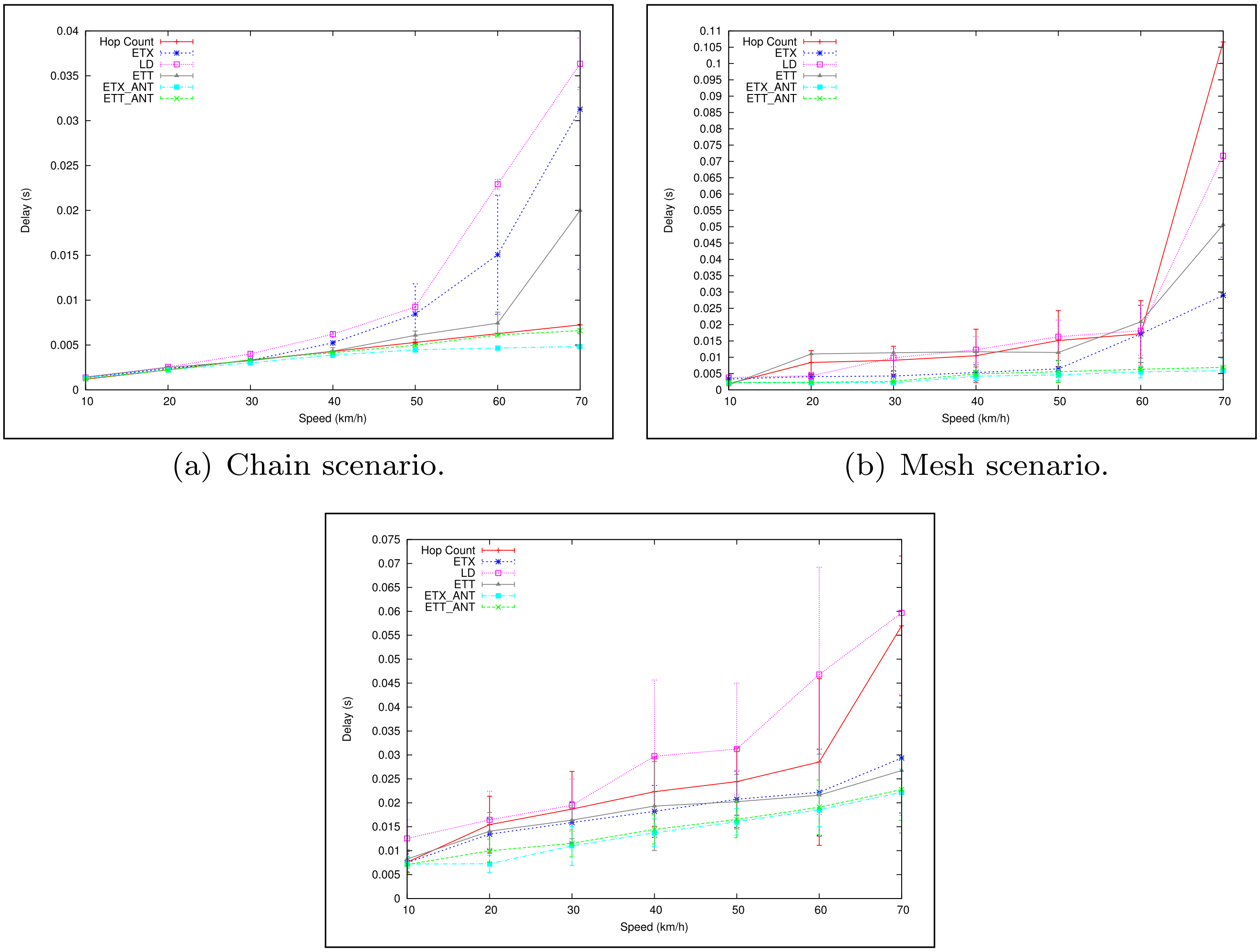


(a) Chain scenario.

(b) Mesh scenario.

(c) Capacity of mobile ad-hoc scenario.

Fig. 5 Throughput for the three scenarios



(a) Chain scenario.

(b) Mesh scenario.

(c) End-to-end delay of mobile ad-hoc scenario.

Fig. 6 End-to-end delay for the three scenarios

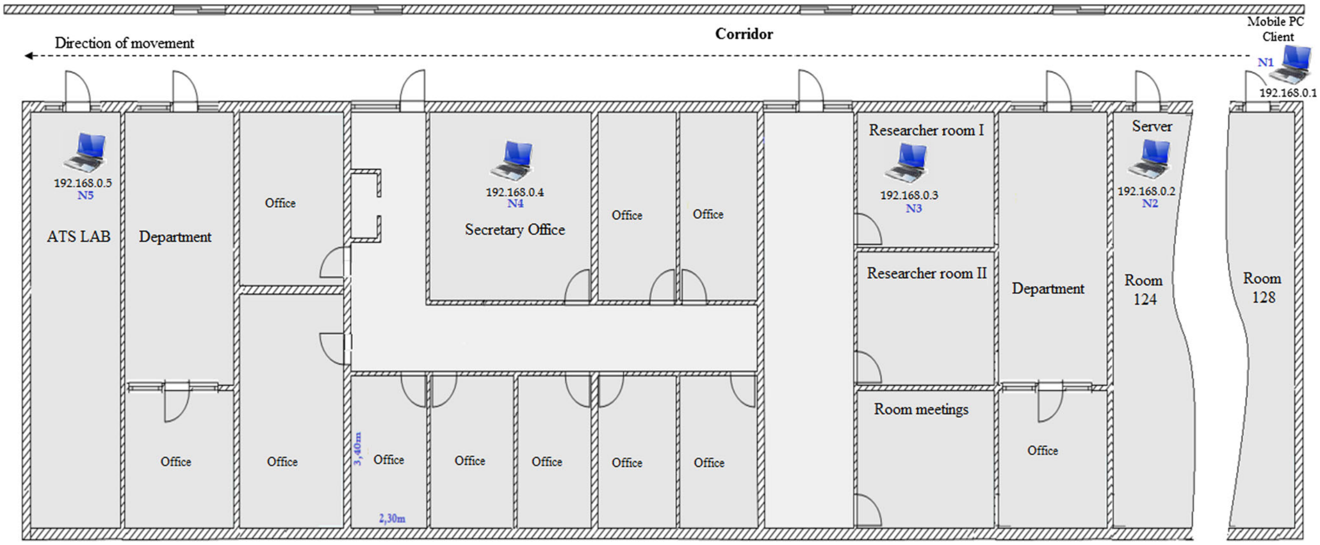


Fig. 7 Environment of the testbed

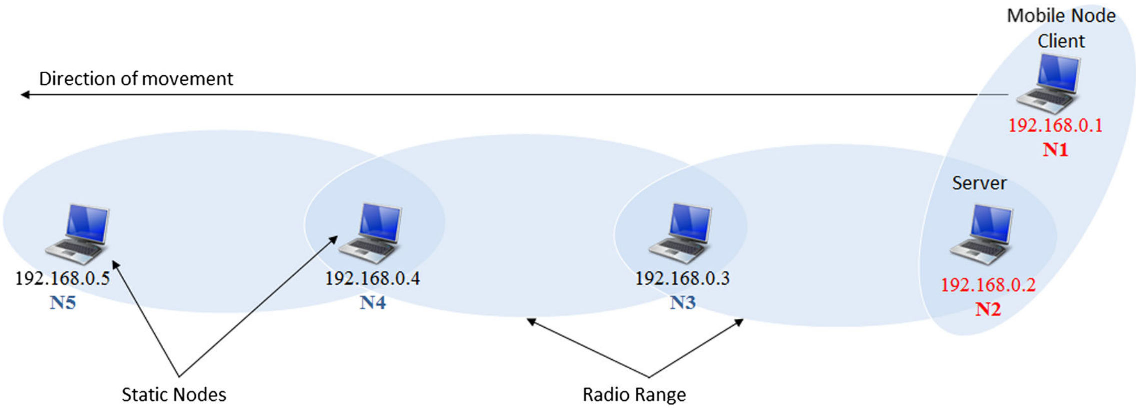


Fig. 8 Chain and wireless links in the testbed

links are used, several transmissions for the same packet are performed at the Wi-Fi level since transmissions fail. It involves a delay due to these different attempts that is also increased by greater value of the back-off (in average). Packets in the node buffer may also be delayed even if they are destined to be transmitted on a link with a good quality (transmitted on the same Wi-Fi card but for a different neighbor).

5 Testbed

We implemented our anticipated metrics on a testbed. It allows us to test our proposed algorithm in a real radio environment, discover potential implementation issues, and evaluate the feasibility of the approach.

Implementation We implemented the ETX_ANT metric. Our implementation has been integrated in the Olsrd daemon. Olsrd is an open source implementation of olsr, coded in C. It is the most famous instance of OLSR. It has been shown stable and has allowed an easy integration of our metrics. The version that we modified is Olsrd-0.6.6.2 [43]. We did not need to add a new field in TC and HELLO messages as ETX metric is already integrated in Olsrd. It was sufficient to add and call functions that obtain signal strength at the reception of HELLO messages, and that anticipate FER accordingly. Our code may be found at the following url [44].

Table 3 Mapping between FER and signal strength

Signal strength	Packet loss
From −70 to −75 dBm	P = 5.52%
From −76 to −80dBm	P = 9.1%
From −81 to −85dBm	P = 32%
at −87 dBm	P = 46%
at −88 dBm	P = 75%
at −89 dBm	P = 94%
< −89dBm	P = 100%

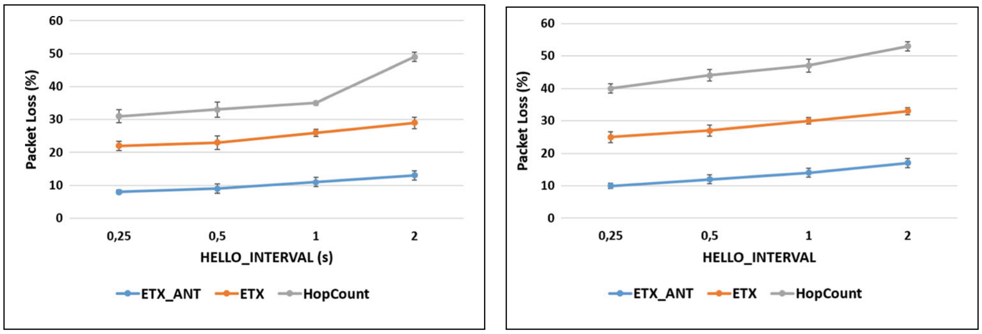
Configuration and testbed setup For the test, we use five laptops with Linux Ubuntu LTS 14.04. These computers are equipped with different wireless cards configured in ad hoc mode (Intel Centrino Wireless-N 1000, Qualcomm Atheros AR9285 Wireless Network Adapter, and Intel Centrino Wireless-N 2230).

We run the experiments inside the laboratory of Communications Systems (Sys'Com) in Tunis. The ad hoc network is formed of five laptops distributed on one floor as shown in Fig. 7. The rooms on that floor are separated by walls (bricks or aluminum walls). It is also important to mention that there are multiple wireless networks in the building that are interfering among each other. We choose a chain topology (Fig. 8) because it presents a simple case that allows us to test the ability of our algorithm to anticipate link breakages and to choose appropriate paths. The mobile PC moved along the corridor and passed through the entire chain of fixed nodes. Figure 8 depicts the wireless links between the fixed nodes. The mobile node (N1) is the

Table 4 Testbed parameters

Parameters	Numerical values
Wireless technology	IEEE 802.11bgn
Number of nodes	5
Routing protocol	OLSRD 0.6.6.2
Frequency band	2.412 GHz
Number of packet for low speed	100 packets
Number of packet for medium speed	60 packets
ANTICIPATION_TIME (OLSRD)	5.0 s
TH_Q (OLSRD)	−64 dBm
HELLO_INTERVAL (OLSRD)	0.25, 0.5, 1.0, 2.0 s
TC_INTERVAL (OLSRD)	2.0, 5.0 s

Fig. 9 Packet losses for different HELLO_INTERVAL and two speeds. TC_INTERVAL = 2 s



(a) Low speed. (b) Medium speed.

client and communicates with the server (N2). We consider different HELLO_INTERVAL, TC_INTERVAL, and two different speeds (approximately 3 and 7 km/h).

Mapping between FER and signal strength To calculate the mapping between FER and signal strength, required for the anticipation, we proceed as follows. We perform a sequence of measures that maps FER and signal strength. With two laptops in the experimental area, we measure the FER as a function of the distance. As the measure varies from one test to another, we repeat this method several times and compute the average. The results are shown in Table 3.

We set the TIME parameter at 5 s. It corresponds to one of the two TC frequency (TC_INTERVAL). We set the TH_Q threshold in order to begin the anticipation 5 s before that losses appear. With the considered speeds, it led to TH_Q=−64 dBm. The signal/FER mapping and the different parameters have been configured in Olsrd. The parameters are summarized in Table 4.

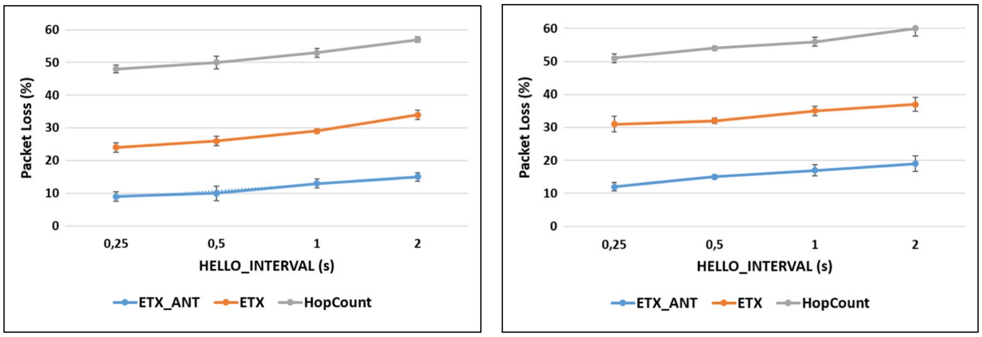
Performance measurement and analysis We implemented and compare the performance of three metrics: hop count, ETX, and ETX_ANT. We vary the frequency of the OLSR control messages to measure their impact on the mobility management (TC_INTERVAL and HELLO_INTERVAL). We measure the loss rate for each experimentation.

Experimentations were repeated five times for each set of configurations (metric, TC_INTERVAL, HELLO_INTERVAL, speed).

Figures 9 (TC_INTERVAL = 2 s) and 10 (TC_INTERVAL = 5 s) show the average for the five experiments for different HELLO_INTERVAL and for both speeds. Results are shown with a confidence interval at 95%. We observe that for the classical metrics (ETX and hop count) losses increase when the control messages frequency decreases (as the routes are updated at a lower frequency). When HELLO_INTERVAL is 0.25 (i.e., each node generates four Hellos per second), they present a high loss rate (20% for TC_INTERVAL = 2 s, and 30% for TC_INTERVAL = 5 s). For a lower frequency of Hellos, they even reach 60% for HOP COUNT, and 37% for ETX. In the other hand, our metric allows to keep a loss rate around 10% and is varying from 8 to 19% according to the different configurations. At low speed (Fig. 9), it reaches 8% for HELLO_INTERVAL = 0.25 s, and does not exceed 13% with HELLO_INTERVAL = 2 s. The two other metrics are more sensitive to HELLO frequency, between 22 and 29% for ETX, and between 31 and 49% for HOP COUNT. It seems thus, that anticipating the metrics makes the routing more robust with regard to the frequency of control messages.

Basically, our experiments show that our anticipation technique improves significantly the PDR and helps to manage node mobility. Our metrics outperform the classical

Fig. 10 Packet losses for different HELLO_INTERVAL and two speeds. TC_INTERVAL = 5 s



(a) Low speed. (b) Medium speed.

ones, which experience a very important loss rate (between 22 and 33% for ETX, and between 31 and 53% for HOP COUNT). Even if the results are very different from those obtained by simulations, we can see that the hierarchy is respected. Unfortunately, our metric does not give a PDR of 100% even with low speeds. This is mainly due to the unfavorable environment of our experiments (indoor) where (i) there were some locations in the trajectory of the mobile nodes where the PDR was not 100% (even when it was static), (ii) we use different computers and cards for which the parameterization (tuning of TH_Q) was difficult, and (iii) mapping between loss rates and signal strength is not uniform in the corridor where the experiments were performed whereas the configuration was the same on all PCs. By involving more human resources and nodes/PCs, it would be interesting to perform tests on another area (outdoor environment) in order to prove that a PDR of 100% is achievable with our metrics.

6 Conclusion

Several techniques of mobility prediction have been designed for ad hoc networks to predict future locations of nodes or the distance between them. But in general, nodes are unable to locate their positions because they are not equipped with location-based devices, and mobility is often unpredictable (because of changes in direction, for example). Furthermore, link qualities are not only related to the distance between nodes, but in function of complex radio phenomena (fading/shadowing) and to the environment in which nodes operate (city, road/highway, etc.). Therefore, we think that a prediction method of link quality based on the signal strength is more appropriate. In this paper, we have presented our routing metrics ETX_ANT and ETT_ANT.

These metrics anticipate the value of classical routing metrics ETX and ETT based on signal strength prediction. When nodes are mobile, this anticipation allows the routing protocol to compensate the delay due to the computation and dissemination of the metrics.

Through a large set of simulations and scenarios, we have shown that our method improves significantly the main performance criteria such as PDR, throughput, and end-to-end delay. Simulations show that the packet delivery ratio is close to 100% for all scenarios and speeds. It empirically proves the ability of our technique to efficiently manage the nodes mobility, even to be optimal (PDR of 100%), when the radio environment is stable and known.

The proposed approach has also been integrated in the OLSR routing protocol, and a set of experiments have been conducted in an indoor environment with different configurations. Losses are approximately three and four times less important with our approach compared to the classical ETX and hop count respectively. But, we still experienced some losses that are mainly explained by the unfavorable experiment area that we have chosen (indoor with a very complex/heterogeneous radio environment). Also, it points out that in a real environment, an important protocol parameterization is crucial to allow an accurate signal prediction and metric anticipation to obtain the targeted performances.

There are several ways to improve or complete this work. Experiments in outdoor environment should be performed to evaluate the performance of our metrics in a more predictive radio environment. Also, other routing protocols could be considered.

The proposed solution has been designed for ad hoc and mesh networks. But these metrics could be used in other contexts. In particular, it would be interesting to adapt such metrics for wireless mobile devices (e.g. smartphone, etc.) in order to help to perform vertical handovers. The idea is to associate these metrics to the different radio interfaces (4G/Wi-Fi/etc.), allowing the device OS to anticipate link breakages and switch seamlessly from one technology to another.

References

1. Siva Ram Murthy C, Manoj BS (2004) Ad hoc wireless networks: architectures and protocols, portable documents. Pearson Education
2. Boukerche A, Turgut B, Aydin N, Ahmad MZ, Bölöni L, Turgut D (2011) Routing protocols in ad hoc networks: a survey. Comput Netw 55(13):3032–3080
3. Lenders V, Wagner J, Heimlicher S, May M, Plattner B (2008) An empirical study of the impact of mobility on link failures in an 802.11 ad hoc network. IEEE Wirel Commun 15(6):16–21
4. Alasmary Waleed, Zhuang Weihua (2012) Mobility impact in ieee 802.11 p infrastructureless vehicular networks. Ad Hoc Netw 10(2):222–230
5. Husain A, Raw RS, Kumar B, Doegar A (2011) Performance comparison of different routing protocols in vehicular network environments. In: Wyld DC, Wozniak M, Chaki N, Meghanathan N, Nagamalai D (eds) Advances in computing and information technology. Springer, Berlin, pp 427–436
6. Gite P, Sharma M (2012) Performance evaluation of ad-hoc network routing protocols using ns2 simulation. ACEEE Int J Netw Secur 3(1):6
7. Bai F, Sadagopan N, Helmy A (2003) Important: a framework to systematically analyze the impact of mobility on performance of routing protocols for adhoc networks. In: INFOCOM (2003), vol 2, pp 825–835
8. Naimi S, Busson A, Vèque V, Bouallegue R, Slama LBH (2014) Mobility management in ad hoc networks using routing metrics. In: 2014 International conference on communications and networking (ComNet). Hammamet, Tunisia, pp 1–6
9. Clausen T, Jacquet P (2003) Optimized link state routing protocol (OLSR), RFC3626
10. Perkins C, Belding-Royer E, Das S (2003) Ad hoc on-demand distance vector (AODV) routing RFC3561

11. Ali HM, Naimi AM, Busson A, Vèque V (2007) An efficient link management algorithm for high mobility mesh networks. In: MobiWac '07, pp 42–49
12. Ali HM, Naimi AM, Busson A, Vèque V (2009) Signal strength based link sensing for mobile ad-hoc networks. Telecommun Syst 42(3-4):201–212
13. Parissidis G, Karaliopoulos M, Baumann R, Spyropoulos T, Plattner B (2009) Routing metrics for wireless mesh networks. Guide Wireless Mesh Netw, 199–230
14. Rondinone M, Ansari J, Riihijärvi J, Mähönen P (2008) Designing a reliable and stable link quality metric for wireless sensor networks. In: Proceedings of the workshop on real-world wireless sensor networks, REALWSN '08. ACM, pp 6–10
15. Houaidia C, van den Bossche A, Idoudi H, Val T, Saïdane LA (2013) Link availability aware routing metric for wireless mesh networks. In: AICCSA. IEEE Computer Society, pp 1–4
16. Naimi S, Busson A, Vèque V, Slama LBH, Bouallegue R (2014) Anticipation of etx metric to manage mobility in ad hoc wireless networks. In: Guo S, Lloret J, Manzoni P, Ruehrup S (eds) Ad-hoc, mobile, and wireless networks. Springer International Publishing, Cham, pp 29–42
17. De Couto DSJ, Aguayo D, Bicket J, Morris R (2005) A high-throughput path metric for multi-hop wireless routing. Wirel Netw 11(4):419–434
18. Esposito PM, Elias M, Campista M, Moraes IM, Henrique L, Costa MK, Duarte OCMB, Rubinstein MG (2008) Implementing the expected transmission time metric for OLSR wireless mesh networks. In: 1st IFIP Wireless days WD '08
19. Qin L, Kunz T (2006) Mobility metrics to enable adaptive routing in manet. In: Wireless and mobile computing, networking and communications (WiMob'2006), pp 1–8
20. Sadagopan N, Bai F, Krishnamachari B, Helmy A (2003) Paths: analysis of path duration statistics and their impact on reactive manet routing protocols. In: Proceedings of MobiHoc '03. ACM, pp 245–256
21. Ashraf U, Abdellatif S, Juanole G (2008) An interference and link-quality aware routing metric for wireless mesh networks. In: IEEE Vehicular technology conference. VTC 2008-Fall., pp 1–5
22. Draves R, Padhye J, Zill B (2004) Routing in multi-radio, multi-hop wireless mesh networks. In: ACM MobiCom '04. ACM, pp 114–128
23. Borges VCM, Pereira D, Curado M, Monteiro E (2009) Routing metric for interference and channel diversity in multi-radio wireless mesh networks. In: Ad-Hoc, mobile and wireless networks. Springer, pp 55–68
24. Youssef M, Ibrahim M, Abdelatif M, Chen L, Vasilakos AV (2014) Routing metrics of cognitive radio networks: a survey. IEEE Commun Surv Tutor 16(1):92–109. First
25. Campista MEM, Esposito PM, Moraes IM, Costa LHMK, Duarte OCMB, Passos DG, de Albuquerque CVN, Saade DCM, Rubinstein MG (2008) Routing metrics and protocols for wireless mesh networks. IEEE Netw 22(1):6–12
26. Gerharz M, de Waal C, Frank M, Martini P (2002) Link stability in mobile wireless ad hoc networks. In: 27th Annual IEEE conference on local computer networks. LCN 2002., pp 30–39
27. Bui N, Cesana M, Hosseini SA, Liao Q, Malanchini I, Widmer J (2017) A survey of anticipatory mobile networking context-based classification, prediction methodologies, and optimization techniques. IEEE Commun Surv Tutor 19(3):1790–1821
28. Gavalas C, Konstantopoulos D, Pantziou G (2010) Mobility prediction in mobile ad-hoc networks. In: Next generation mobile networks and ubiquitous computing, IGI Global, pp 226–240
29. Ghouti L, Sheltami TR, Alutaibi KS (2013) Mobility prediction in mobile ad hoc networks using extreme learning machines. Procedia Comput Sci 19:305–312
30. Li X, Mitton N, Simplot-Ryl D (2011) Mobility prediction based neighborhood discovery in mobile ad hoc networks. In: Networking, pp 147–159
31. Suraj R, Tapaswi S, Yousef S, Pattanaik KK, Cole M (2016) Mobility prediction in mobile ad hoc networks using a lightweight genetic algorithm. Wirel Netw 22(6):1797–1806
32. Ali SS, Kamalrulnizam AB, Kayhan ZG, Gonzalez AJ (2011) Mobility and signal strength- aware handover decision in mobile IPv6 based wireless LAN. In: Proceeding of the international multiconference of engineers and computer scientists, IMECS'2011, Hong Kong
33. Su W, Lee SJ, Gerla M (2000) Mobility prediction in wireless networks. In: MILCOM 2000. 21st Century military communications conference proceedings, vol 1, pp 491–495
34. Su William, Lee S-J, Gerla M (2001) Mobility prediction and routing in ad hoc wireless networks. Int J Netw Manag 11(1):3–30
35. Capka J, Boutaba R (2004) Mobility prediction in wireless networks using neural networks. In: Vicente J, Hutchison D (eds) Management of multimedia networks and services, volume 3271 of lecture notes in computer science. Springer, Berlin, pp 320–333
36. Wang Z, Xu Y, Li L, Tian H, Cui S (2018) Handover control in wireless systems via asynchronous multi-user deep reinforcement learning. arXiv:1801:02077
37. Mousavi SM, Rabiee HR, Moshref M, Dabirmoghaddam A (2007) Model based adaptive mobility prediction in mobile ad-hoc networks. In: International conference on wireless communications, networking and mobile computing. WiCom 2007., pp 1713–1716
38. Duel-Hallen A, Hu S, Hallen S (2000) Long range prediction of fading signals Enabling adaptive transmission for mobile radio channels. IEEE Signal Process Mag 17:62–75
39. Long X, Sikdar B (2008) A real-time algorithm for long range signal strength prediction in wireless networks. In: IEEE Wireless communications and networking conference. WCNC'08., pp 1120–1125
40. Clausen T, Dearlove C, Jacquet P, Herberg U (2014) The optimized link state routing protocol version 2, RFC7181 apr
41. Erceg V, Greenstein LJ, Tjandra SY, Parkoff SR, Gupta A, Kulic B, Julius AA, Bianchi R (1999) An empirically based path loss model for wireless channels in suburban environments. IEEE J Select Areas Commun 17(7):1205–1211
42. Hyytiä E, Virtamo J (2007) Random waypoint mobility model in cellular networks. Wirel Netw 13(2):177–188
43. http://www.olsr.org. [Online]
44. http://www.anthonybusson.fr/index.php/2-uncategorised/13-metric-etx-ant. [Online]