

# Link Stability Estimation Based on Link Connectivity Changes in Mobile Ad-hoc Networks

Qingyang Song<sup>1</sup>, Zhaolong Ning<sup>1</sup>, Shiqiang Wang<sup>1, 2</sup>, and Abbas Jamalipour<sup>3</sup>

<sup>1</sup> School of Information Science and Engineering, Northeastern University, Shenyang, 110819, P. R. China

<sup>2</sup> Dept. of Electrical and Electronic Engineering, Imperial College London, SW7 2AZ, United Kingdom

<sup>3</sup> School of Electrical and Information Engineering, University of Sydney, NSW, 2006, Australia

Emails: songqingyang@ise.neu.edu.cn, zhaolongning@gmail.com, shiqiang.wang11@imperial.ac.uk,  
a.jamalipour@ieee.org

A preliminary version of this paper has been presented in the 2010 IEEE International Conference on Wireless Communications, Networking and Information Security (IEEE WCNIS 2010) [1]. This paper provides a more detailed analysis of the proposed link stability estimation algorithm, extends the simulation results, and also proposes an application of the link stability estimation scheme to routing protocols of mobile ad hoc networks.

*Abstract* - Link stability issue is significant in many aspects, especially for the route selection process in mobile ad-hoc networks (MANETs). Most previous works focus on the link stability in static environments, with fixed sampling windows which are only suitable for certain network topologies. In this paper, we propose a scheme to estimate the link stability based on link connectivity changes, which can be performed on the network layer, without the need of peripheral devices or low layer data. We adopt a variable sized sampling window and propose a method to estimate the link transition rates. The estimation scheme is not restricted to specific network topologies or mobility models. After that, we propose a routing method which adjusts its operating mode based on the estimated link stability. Simulation results show that the proposed scheme can provide correct estimation in both stationary and non-stationary scenarios, and the presented routing protocol outperforms conventional routing schemes without link stability estimation.

*Keywords* - Link stability; mobile ad-hoc networks (MANETs); probabilistic model; routing protocol

## 1 Introduction

Many design issues in mobile ad-hoc networks (MANETs) are challenging due to the dynamic characteristics of the networks. Links can be unstable due to the random property of the wireless channel (e.g. fading, shadowing or noise) which affect the network performance significantly, especially for the routing process [2-5, 8]. Hence, link stability estimation methods are worth to be studied.

In [6] and [7], the notion of *associativity*, used as a metric to evaluate the link stability, was defined as the number of consecutive beacons. This method is simple; however it relies on some experimentally chosen parameters that have large influence on its performance, and is unstable in some scenarios. In [9], a network utility maximization problem was formulated by decomposing the input rate control and the scheduling problem under stability constraints. This is a rather complicated linear programming (LP) problem by introducing scheduling mechanism, after that it was relaxed to a simpler form under certain assumptions. However, fixed scheduling rates per link is not always true especially in MANETs.

In [10], a probabilistic model was used to evaluate the link stability by redefining the link availability as the conditional probability that a link remains connected throughout a specific time period, given that it is currently connected. This concept is extended in [11] and [12] for multicast scenarios; however, the link stability prediction schemes proposed in [10–12] are based on the random walk or random way point model and require the prior knowledge of some parameters of the mobility model. In [13], the authors presented a pattern matching based approach to predict link quality variation. This approach does not require the use of any external hardware, and relies simply on Signal to Noise Ratio (SNR) measurements. The nodes monitor and store the links' SNR values to their neighbors in order to obtain a time series of SNR measurements. When a prediction on the future state of a link is required, the node looks for similar SNR patterns to the current situation in the past (time series) using a cross-correlation function. The matches found are then used as a base for the prediction. Most link stability estimation schemes in the literatures [10–14] are based on low layer measurements, such as RSS; however, collecting and analyzing low layer data is generally more complex than only considering the link connectivity, which is accomplished on the network layer. Several literatures focus

on Global Positioning System (GPS) based mobility prediction schemes [15-16]. These methods predict the concrete geographical locations and velocity of nodes; however, GPS devices are not cost effective and restricted in many short range or indoor applications.

In MANETs, nodes communicate with each other by forming a multi-hop network in a decentralized manner without the aid of any pre-existing infrastructure. Ad-hoc networks have to face several challenges, such as dynamic topology, real-time communication, resource constraint, bandwidth management and packet broadcast overhead, thus making it complicated to design routing protocols [17-22]. There have been many routing protocols developed for MANET over the past few years, which can be generally classified into position-based and topology-based routing protocols. Position-based routing protocols select paths based on geographical information with geometrical algorithms. Topology-based routing protocols, which can be further divided into proactive, reactive and hybrid routing protocols [23, 24], select paths based on topological information. In proactive routing protocols such as optimized link state routing (OLSR) and destination sequenced distance vector (DSDV), every node knows a route to every other node all the time. There is no latency, however permanent maintenance of unused routes increases the control overhead. This type of protocols suits for the situation of low node mobility and high traffic load. Reactive protocols, such as ad-hoc on demand distance vector (AODV) and dynamic source routing (DSR), compute a route only when it is needed. This reduces the control overhead but introduces latency for the first packet to be sent due to the time needed for the on-demand route setup. This protocol is appropriate for high node mobility and low traffic load. Hybrid routing protocols, such as zone routing protocol (ZRP) and hybrid wireless mesh protocol (HWMP), try to combine the advantages of both the philosophies: proactive is used for near nodes or often used paths, while reactive routing is used for more distant nodes or less often used paths [25, 26]. Many of these routing protocols use shortest path algorithm with minimum hop count to reduce transmission latency and error rate. However, studies show that the shortest path is not necessarily the best path. Selecting a route based on the shortest hop-count metric without considering link stability leads to frequent route failures. Since link stability and

routing selection have some joint effects on the network performance, these two methods are worth to be studied.

In recent years, many literatures focus on routing protocols design. In [6], the authors proposed a Quality of Service (QoS) routing with throughput and delay constraints, and then the destination can select the link with the highest route stability value to reply to the source each time. In [18], the authors proposed a scheme for multipath multicast routing in MANETs to select neighbors with high reliability pair factor based on minimum value of reliability pair factor of a path. However, the link state varies with time, and it is rather complex to select paths for dynamic topology for [6] and [18]. In [22], the authors presented a greedy-based backup routing (GBR) protocol to improve route stability in mobile ad hoc networks. Route discovery for the primary path is mainly based on greedy forwarding; therefore, a primary path established in GBR approximately achieves the smallest hop count which turns back to the hop based routing strategy.

In this paper, we propose a scheme to estimate the link stability and apply the proposed estimation scheme to routing protocols. We first propose a mathematical model for link stability, and then propose a method to estimate the parameters of the link connectivity model. A variable sized sampling window is adopted to make the estimation scheme applicable for both stationary and non-stationary scenarios. Afterwards, we propose a routing method with link stability estimation and compare the packet transmission effectiveness among different routing protocols in MANETs. Different from previous literatures that focused on routing design, our routing protocol neither adopts the shortest hop count method nor greedily selects the links with highest route stability value, it dynamically switches among different routing protocols according to the channel state, which is easy for implementation. The proposed link stability estimation and routing schemes are not restricted to certain network topologies, and they do not require prior information about the mobility model of the network.

The remainder of this paper is organized as follows. Section 2 describes the link connectivity model (LCM) used for link stability estimation. Section 3 gives detailed description of the proposed link

stability estimation scheme and presents a routing method with link stability estimation. Section 4 shows the simulation results. Section 5 summarizes the paper.

## 2 Link Connectivity Model

In this section, we propose a probabilistic model to describe the connectivity of a link. Based on this model, the metrics to evaluate the link stability are outlined.

### 2.1 Model Construction

Let  $i$  and  $j$  denote arbitrary nodes in the network, a packet sending from Node  $i$  to Node  $j$  at an arbitrary time instant  $t$  can be successfully received by Node  $j$  if the following Signal-to-Interference-plus-Noise Ratio (SINR) condition is satisfied:

$$\frac{P_i(t)g_{ij}(t)}{N_j + \sum_{k \in V_I(t), k \neq i} P_k(t)g_{kj}(t)} \geq \beta_j \quad (1)$$

where  $N_j$  denotes the additive background noise at the receiver of Node  $j$ ,  $P_i(t)$  denotes the transmitting power of Node  $i$ ,  $g_{ij}(t)$  is the channel gain from Node  $i$  to Node  $j$ ,  $V_I(t)$  is the set of interfering nodes transmitting at the same time instant  $t$ , and  $\beta_j$  is the SINR requirement for the receiver of Node  $j$  to successfully decode a packet. When (1) is satisfied, the link from Node  $i$  to Node  $j$  (denoted by  $i \rightarrow j$ ) is connected at time  $t$ . If the two nodes' wireless channel characteristics are almost identical, Link  $i \rightarrow j$  can be considered as bi-directional link denoted by Link  $i \leftrightarrow j$ .

Due to the random properties of the wireless channels, the connectivity of Link  $i \leftrightarrow j$  can be denoted by the process

$$X_{ij}(t) = \begin{cases} 0, & \text{Link } i \leftrightarrow j \text{ is not connected at time } t \\ 1, & \text{Link } i \leftrightarrow j \text{ is connected at time } t \end{cases} \quad (2)$$

As in [27, 28], when only considering the radio channel characteristics,  $X_{ij}(t)$  has the Markov property. The link connectivity is mainly affected by the radio channel characteristics when both nodes are fixed or move at a fairly low speed. The probability of incorrect reception of a packet is generally indicated by the error probability, which is independent of whether the previous packet has been received successfully.

We regard  $X_{ij}(t)$  as the output of a two-state continuous-time Markov chain. We call this Markov chain the *continuous time link connectivity model (CTLCM)* for Link  $i \leftrightarrow j$ . The transition rate from State 0 to State 1 is denoted by  $\lambda_{ij}(t)$ , and the transition rate from State 1 to State 0 is denoted by  $\mu_{ij}(t)$ . When we sample  $X_{ij}(t)$  at a sampling interval  $\Delta t$ , the resulting random series  $\{X_{ij}(k\Delta t)\}$  can be regarded as the output of a two-state discrete-time Markov chain. This Markov chain is called the *discrete time link connectivity model (DTLCM)* for Link  $i \leftrightarrow j$ . The transition probability from State  $m$  to State  $n$  ( $m, n = 0, 1$ ) is denoted by  $p_{mn}(t)$ .

In the following context, the subscripts  $i$  and  $j$  are omitted where no ambiguity occurs and within a certain interval,  $\lambda(t)$ ,  $\mu(t)$  and  $p_{mn}(t)$  are independent of  $t$ , which can be denoted by  $\lambda$ ,  $\mu$  and  $p_{mn}$  respectively. Meanwhile, by sampling the output  $X_{ij}(t)$  of the CTLCM, we can obtain the transition probability matrix of the DTLCM:

$$P(\Delta t) = \frac{1}{\lambda + \mu} \begin{bmatrix} \mu + \lambda e^{-(\lambda + \mu)\Delta t} & \lambda - \lambda e^{-(\lambda + \mu)\Delta t} \\ \mu - \mu e^{-(\lambda + \mu)\Delta t} & \lambda + \mu e^{-(\lambda + \mu)\Delta t} \end{bmatrix} \quad (3)$$

Under the above described model, we can estimate the future link states based on the transition probabilities at present time, as long as the time we estimate lies in the same interval as the present time.

## 2.2 Link Stability Metrics

To define the link stability metrics, we start with the following definition.

**Definition 1:** A continuous time (or respectively, discrete time) Markov chain of finite states is *piecewise homogeneous* when its transition rates (or probabilities) are constant over partitioned time intervals, but not constant over the whole time axis. More precisely, given a Markov chain with  $K$  states, let  $\rho_{m'n'}(t)$  denote the transition rate (or probability) from State  $m'$  to State  $n'$  ( $m', n' = 0, 1, \dots, K-1$ ), and  $T_1, T_2, \dots, T_q, \dots, T_Q$  be discrete points on the time axis. If for  $\forall t_1, t_2 \in [T_q, T_{q+1}), q \in [1, Q-1]$ , we have  $\rho_{m'n'}(t_1) = \rho_{m'n'}(t_2)$ ; but for  $\forall t_1 \in [T_q, T_{q+1}), t_2 \in [T_{q-1}, T_q), q \in [2, Q-1]$ , there exists some  $m', n'$ , such that  $\rho_{m'n'}(t_1) \neq \rho_{m'n'}(t_2)$ , then the corresponding Markov chain is *piecewise homogeneous*, and the intervals  $[T_q, T_{q+1})$  are called the *homogeneous intervals* of the Markov chain.

Since the CTLCM is a continuous time Markov chain, the waiting time of a state transition has an exponential distribution [29]. Let  $\eta_{01}$  denote the waiting time of State 0 to State 1 transition, and  $\eta_{10}$  denote the waiting time of State 1 to State 0 transition. Within a specific homogeneous interval, we have

$$\begin{cases} P\{\eta_{01} > \tau\} = e^{-\lambda\tau} \\ P\{\eta_{10} > \tau\} = e^{-\mu\tau} \end{cases} \quad (4)$$

where  $\tau \geq 0$ .

Let  $t_p$  denote the present time, we evaluate the link stability probability when the link remains connected for time  $\tau$  (called the *remaining probability*):

$$p_{remain}(\tau) = P\{\eta_{10} > \tau\} = e^{-\mu(t_p)\tau} \quad (5)$$

and the probability the link recovers within time  $\tau$  after a link failure (called the *recovering probability*):

$$p_{recover}(\tau) = P\{\eta_{01} \leq \tau\} = 1 - e^{-\lambda(t_p)\tau} \quad (6)$$

$p_{remain}(\tau)$  is generally more significant than  $p_{recover}(\tau)$ , since the probability of the link remains connected for a given time  $\tau$  can be used to estimate how long the link is connected. However, in many scenarios of ad-hoc networks, especially when hidden terminals are present, collisions are likely to occur which may result in a temporal loss of packets. Therefore, introducing  $p_{recover}(\tau)$  is necessary as it can be adopted to determine whether a link failure is temporal.

### 3 Link Stability Estimation

The main task of link stability estimation is to estimate the probabilities denoted in (5) and (6). Since the remaining and recovering probabilities for different values of  $\tau$  have to be estimated, it is more convenient to estimate the transition rates  $\lambda(t_p)$  and  $\mu(t_p)$  in the CTLCM and evaluate  $p_{remain}(\tau)$  and  $p_{recover}(\tau)$  according to (5) and (6). In this section, we discuss methods to estimate  $\lambda(t_p)$  and  $\mu(t_p)$ . We first consider the case where all the link state samples are from the homogeneous interval, then discuss how to deal with samples from different homogeneous intervals, and finally outline the procedure of link stability estimation.

### 3.1 Transition Rate Estimation

To estimate the transition rates in the CTLCM, we firstly consider that all the link state samples are from the homogeneous interval  $t_p$ , and omit the variable  $t_p$  for simplicity. The time of link connection or disconnection has an exponential distribution. Hence, a straightforward idea is to estimate  $\lambda$  and  $\mu$  with the reciprocal value of the mean continuous connection or disconnection time of a link. However, since we can only obtain the link connectivity at discrete time instants by sending periodically beacons in practice, the connection or disconnection time is actually truncated and may increase estimation errors. In this paper, we propose an estimation scheme, and estimate the transition probabilities  $p_{mn}$  in the DTLCM. Based on the estimated transition probabilities, the transition rates  $\lambda$  and  $\mu$  in the CTLCM can be obtained.

#### 3.1.1 Estimating $p_{mn}$

Each node in the network broadcasts beacons periodically to notify its presence to other nodes within its transmission range. When we set the interval of beacons equal to  $\Delta t$ , we obtain samples of the connectivity of Link  $i \leftrightarrow j$  based on whether Node  $i$  has received a beacon from Node  $j$  at each sampling instant. And the sampled link connectivity is the output of the DTLCM. Hence, the transition probability  $p_{mn}$  can be estimated based on the link state samples at discrete time instants.

Since the DTLCM within the homogeneous interval containing  $t_p$  is a two-state discrete-time homogeneous Markov chain, its transition probability matrix can be expressed by:

$$\mathbf{P} = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} \quad (7)$$

where  $p_{m0} + p_{m1} = 1$ . Let

$$Y_m = \begin{cases} 0, & X_{k+1} \neq X_k \\ 1, & X_{k+1} = X_k \end{cases}, \text{ for } \forall k \in \mathbf{Z}, X_k = m \quad (8)$$

where  $X_k = X(k\Delta t)$  and  $\mathbf{Z}$  is the set of integers. The probability function (p.f.) of  $Y_m$  is

$$f(y; p_{mm}) = p_{mm}^y (1 - p_{mm})^{1-y} \quad (9)$$

Obviously,  $Y_m$  has a Bernoulli distribution with parameter  $p_{mm}$ . Its expectation  $\mu = p_{mm}$ , and its variance  $\sigma^2 = p_{mm} (1 - p_{mm})$ .

Since  $p_{m0} + p_{m1} = 1$ , we only have to estimate  $p_{00}$  and  $p_{11}$ , then  $p_{01}$  and  $p_{10}$  can be calculated easily. Let  $t_0$  be the latest sampling instant in the DTLCM prior to the present time  $t_p$ , and  $C(L) = \{c(t) : t = t_0 - L\Delta t, t_0 - (L-1)\Delta t, \dots, t_0\}$  be the set of sample values, where  $c(t) \in \{0, 1\}$  denotes the link connectivity at time  $t$ , and  $L$  is a positive integer which indicates how many samples we take into account in our estimation. In our preliminary discussion, we regard  $L$  as the number of all the stored samples which varies when the samples are from different homogeneous intervals, which will be described in Section 3.2.

Since the future state is determined by the current state in the Markov process, we need to consider the *transitions* between two consecutive samples rather than the individual sample values. We categorize the transitions into two groups: one group contains the transitions starting at State 0, and the other group contains the transitions starting at State 1. Let  $N_m(L)$  denote the number of transitions starting at State  $m$  in the set  $C(L)$ , then the frequency of State  $m$  to State  $m$  transitions in the set  $C(L)$  can be denoted by  $\bar{y}_m(L)$ . More precisely,

$$\bar{y}_0(L) = \frac{\sum_{i=1}^L [1 - c(t_0 - (i-1)\Delta t)] \cdot [1 - c(t_0 - i\Delta t)]}{\sum_{i=1}^L [1 - c(t_0 - i\Delta t)]} \quad (10)$$

and

$$\bar{y}_1(L) = \frac{\sum_{i=1}^L c(t_0 - (i-1)\Delta t) \cdot c(t_0 - i\Delta t)}{\sum_{i=1}^L c(t_0 - i\Delta t)} \quad (11)$$

The remaining problem is to perform an estimation of the parameter in the p.f. of a Bernoulli distribution.

Although the maximum likelihood estimator (MLE) of  $p_{mm}$  is  $\bar{y}_m(L)$  [30], some problems will arise when the sample size  $N_m(L)$  is small. This problem is more likely to occur shortly after a node initialization, since insufficient link state samples have been collected in that case. A solution to this problem is using the Bayes estimator which intends to minimize the expected mean squared error of the estimated parameter. The Bayes estimator of  $p_{mm}$  [30] is

$$\hat{p}_{mm} = \frac{N_m(L)\bar{y}_m(L)+1}{N_m(L)+2} \quad (12)$$

When  $N_m(L) \rightarrow \infty$ ,  $\hat{p}_{mm} = \bar{y}_m(L)$ , which is equal to the MLE value of  $p_{mm}$  and is unbiased.

### 3.1.2 Estimating $\lambda$ and $\mu$

After we estimated  $p_{mm}$  in the DTLCM, we propose a method to estimate the transition rates  $\lambda$  and  $\mu$  in the CTLCM. The estimation of the transition rates is based on the condition that the link break and reconnect (or vice versa) with a negligible probability  $\alpha_1$  within one sampling interval  $\Delta t$ . We make some restrictions on the value of the sampling interval  $\Delta t$  to satisfy the condition above, so that

$$p_{e1} = P\{\eta_{10} \leq \Delta t, X(\Delta t) = 1 | X(0) = 1\} \leq \alpha_1 \quad (13)$$

and

$$p_{e0} = P\{\eta_{01} \leq \Delta t, X(\Delta t) = 0 | X(0) = 0\} \leq \alpha_1 \quad (14)$$

From (3), we can obtain

$$\begin{aligned} p_{11}(\Delta t) &= P\{X(\Delta t) = 1 | X(0) = 1\} \\ &= p_{e1} + P\{\eta_{10} > \Delta t, X(\Delta t) = 1 | X(0) = 1\} \\ &= p_{e1} + P\{\eta_{10} > \Delta t | X(0) = 1\} \cdot P\{\eta_{10} > \Delta t \subseteq \{X(\Delta t) = 1\}\} \\ &= p_{e1} + e^{-\mu\Delta t} \end{aligned} \quad (15)$$

Hence,

$$p_{e1} = p_{11}(\Delta t) - e^{-\mu\Delta t} = \frac{\lambda + \mu e^{-(\lambda+\mu)\Delta t}}{\lambda + \mu} - e^{-\mu\Delta t} \leq \alpha_1 \quad (16)$$

Similarly,

$$p_{e0} = p_{00}(\Delta t) - e^{-\lambda\Delta t} = \frac{\mu + \lambda e^{-(\lambda+\mu)\Delta t}}{\lambda + \mu} - e^{-\lambda\Delta t} \leq \alpha_1 \quad (17)$$

Eqs. (16) and (17) have no analytical solution. For applications with strict requirements, they can be solved numerically and even in real time. While in ordinary cases,  $\Delta t$  can be set to a fixed value that is selected experimentally. Under the above condition, the transition probability matrix of the DTLCM can be simplified as

$$\mathbf{P}(\Delta t) = \begin{bmatrix} P\{\eta_{01} > \Delta t\} & P\{\eta_{01} \leq \Delta t\} \\ P\{\eta_{10} \leq \Delta t\} & P\{\eta_{10} > \Delta t\} \end{bmatrix} = \begin{bmatrix} e^{-\lambda\Delta t} & 1 - e^{-\lambda\Delta t} \\ 1 - e^{-\mu\Delta t} & e^{-\mu\Delta t} \end{bmatrix} \quad (18)$$

Then the transition rates in the CTLCM can be easily solved by

$$\begin{cases} \lambda = -(\ln p_{00}) / \Delta t \\ \mu = -(\ln p_{11}) / \Delta t \end{cases} \quad (19)$$

Eq. (19) indicates that  $\lambda$  is a function of  $p_{00}$  and  $\mu$  is a function of  $p_{11}$ . The MLE estimators of  $\lambda$  and  $\mu$  under the condition  $N_m(L) \rightarrow \infty$  can be obtained by

$$\begin{cases} \hat{\lambda} = -(\ln \hat{p}_{00}) / \Delta t \\ \hat{\mu} = -(\ln \hat{p}_{11}) / \Delta t \end{cases} \quad (20)$$

If  $\hat{p}_{00}$  and  $\hat{p}_{11}$  maximize the likelihood function,  $\hat{\lambda}$  and  $\hat{\mu}$  maximize the likelihood function. With ordinary values of  $N_m(L)$ ,  $\hat{\lambda}$  and  $\hat{\mu}$  are also likely equal to their actual values since they are evaluated from  $\hat{p}_{00}$  and  $\hat{p}_{11}$  that have minimal expected mean squared estimation errors.

By substituting (20) into (5) and (6),  $p_{remain}(\Delta t)$  and  $p_{recover}(\Delta t)$  can be directly evaluated from the estimated transition probabilities  $\hat{p}_{00}$  and  $\hat{p}_{11}$ , getting rid of the assumption that waiting time of link transitions are exponentially distributed in prior literatures. From the simulation results in Section 5, we will see that even with a non-Markov process, our proposed method can still make relatively correct estimation of  $p_{remain}(\tau)$  and  $p_{recover}(\tau)$  when  $\tau$  is small.

### 3.2 Sampling window size selection

Now we consider the case where the collected samples are from different homogeneous intervals. This may happen when the node changes its mobility within the time of its operation, which causes a change in the transition rates  $\lambda$  and  $\mu$  and the transition probabilities  $p_{mn}$  respectively.

Since the exact boundary points between homogeneous intervals are difficult to find, and in some cases the speed of a node is a continuous function of time or the variation of the actual transition probabilities are so slight that they are not observable from the samples. We can regard those samples that are *dramatically* different from the recent samples from the previous homogeneous interval. Therefore, the boundary point can be seen as the time instant when  $\lambda$ ,  $\mu$  and  $p_{mn}$  significantly vary from their previous values or the time instant when slight variations add up to a significant variation.

The problem of finding the latest boundary point can be regard as choosing an appropriate sampling size  $L$ . The time interval in which the link state is sampled called *sampling window* which is equal to  $L\Delta t$ , should be set appropriately to balance between the number of samples and the transient response of a dramatic change in link stability. In other words, when the channel state is relative stability,  $L$  should increase and the variance of the estimated parameters can be reduced. When the channel state changes dramatically,  $L$  should decrease to avoid the influence of past state.

We adopt the hypothesis testing method to find the optimal  $L$ . Its main idea is to expand the sampling window in the direction of past time, i.e. increase  $L$  by one at each step. After each expansion, the transition probabilities  $p_{mn}(t_p)$  are estimated. Then we test the hypothesis with  $H_0$ : {the transition probabilities throughout the sampling window are equal to the estimated values}, and continue the expansion until the hypothesis is rejected. The hypothesis value of  $L$  before the rejection is returned as the optimal sample size, and the transition probabilities  $p_{mn}(t_p)$  estimated with that value of  $L$  are used for further processing.

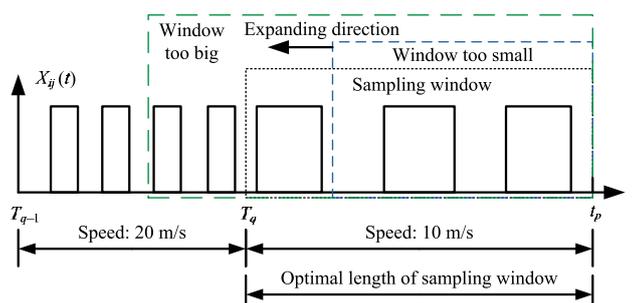


Fig. 1. Example of homogeneous intervals and sampling window.

Let  $Y_{m1}, Y_{m2}, \dots, Y_{mN}$  be the  $N$  samples from the sample space of  $Y_m$ . Since  $Y_m$  has a Bernoulli distribution as described in Section 3.1, from the Central Limit Theorem, we know that when  $N$  is sufficiently large, the distribution of

$$\frac{\bar{Y}_m - p_{mm}}{\sqrt{p_{mm}(1 - p_{mm})/N}} \quad (21)$$

is approximately a standard normal distribution. Hence, considering the samples from  $C(L)$ , the rejection region for  $H_0: \{p_{mm} = p_{mm0}\}$  with significant level  $\alpha_2$  is

$$\left\{ \mathbf{y} : \left| \frac{\bar{y}_m(L) - p_{mm0}}{\sqrt{p_{mm0}(1 - p_{mm0})/N_m(L)}} \right| > u_{1-\alpha_2/2} \right\} \quad (22)$$

where  $u_\alpha$  is the  $\alpha$ -quantile of the standard normal distribution, and  $p_{mm0}$  is the hypothesis testing probability for  $H_0$ .

To test whether the estimated values  $\hat{p}_{00}$  and  $\hat{p}_{11}$  are correct within the *entire* sampling window, we consider a *union set* of hypothesis. The rejection region for the hypothesis  $H_0: \{ p_{00} = \hat{p}_{00} \text{ and } p_{11} = \hat{p}_{11} \text{ within } C(L) \}$  is

$$\bigcup_{l=0}^L \bigcup_{m=0}^1 \left\{ \mathbf{y} : \left| \frac{\bar{y}_m(l) - \hat{p}_{mm}}{\sqrt{\hat{p}_{mm}(1 - \hat{p}_{mm})/N_m(l)}} \right| > u_{1-\alpha_2/2} \right\} \quad (23)$$

Eq. (23) indicates that we test whether the hypothesis is accepted for all possible intervals from the present time  $t_p$  on to the past direction, rather than only the entire sampling window. The reason is when the sampling window contains different homogeneous intervals, the estimation may be correct for the entire sampling window, however it is less likely to be correct for a subinterval containing  $t_p$ .

In order to make the Central Limit Theorem be applicable, we should assure a sufficiently large  $N_m(l)$ , and the rejection region for  $H_0: \{ p_{00} = \hat{p}_{00} \text{ and } p_{11} = \hat{p}_{11} \text{ within } C(L) \}$  can be modified as

$$\bigcup_{l=0}^L \bigcup_{m=0}^1 \left\{ \mathbf{y} : \left| \frac{\bar{y}_m(l) - \hat{p}_{mm}}{\sqrt{\hat{p}_{mm}(1 - \hat{p}_{mm})/N_m(l)}} \right| > u_{1-\alpha_2/2}, N_m(l) > N_{min} \right\} \quad (24)$$

where  $N_{min}$  is the minimum number of transitions (which is obtained from the samples as described in Section 3.1) starting from State 0 or State 1. When the sample size is too small, (22) and (23) make no sense since they are derived based on the Central Limit Theorem. Hence, the hypothesis can only be rejected when the amount of samples is sufficient.

### 3.3 The Estimation Process

Define that Node  $i$  is responsible for the stability of Link  $i \leftrightarrow j$ , according to the above discussions, the pseudo code of the estimation process can be formulated by Algorithm 1.

$L_{max}$  is the total number of stored samples. In addition, the samples outside the sampling window are freed from memory when they have never been included in the sampling window for a pre-defined time period. The hypothesis testing process contains one loop itself and the complexity of the above estimation process is  $O(L^2)$ .

---

Algorithm 1: Pseudo code of estimation process

---

```

for  $L = 1, 2, \dots, L_{max}$  do
    {Estimate  $\hat{p}_{00}$  and  $\hat{p}_{11}$  according to (12)};
    if ( $H_0: \{p_{00} = \hat{p}_{00}$  and  $p_{11} = \hat{p}_{11}$  within  $C(L)$ 
        is rejected according to (24)) break;
     $p_{00st} \leftarrow \hat{p}_{00}, p_{11st} \leftarrow \hat{p}_{11}$ ;
     $\hat{p}_{00} \leftarrow p_{00st}, \hat{p}_{11} \leftarrow p_{11st}$ ;
    {Estimate  $\hat{\lambda}$  and  $\hat{\mu}$  according to (20)};
    {Evaluate the stability according to (5) and (6)};

```

---

The estimation process can be simplified by solving an acceptance interval  $[p_{mm1}, p_{mm2}]$  for a specific set of samples, so that if the estimated transition probability  $\hat{p}_{mm} \in [p_{mm1}, p_{mm2}]$ ,  $H_0: \{p_{mm} = \hat{p}_{mm}\}$  is accepted. This interval can be easily solved from (22):

$$\begin{cases} p_{mm1} = (-b - \sqrt{b^2 - 4ac}) / (2a) \\ p_{mm2} = (-b + \sqrt{b^2 - 4ac}) / (2a) \end{cases} \quad (25)$$

where  $a = N_m(L) + u_{1-\alpha_2/2}^2$ ,  $b = -[2N_m(L)\bar{y}_m(L) + u_{1-\alpha_2/2}^2]$  and  $c = N_m(L)\bar{y}_m^2(L)$ .

To simplify the estimation process further, we regard samples with the same link state as samples from the same homogeneous interval, which is a general case since we have to sample at a much higher rate than the mobility variation in order to estimate the transition probabilities. With this consideration, the probabilities only need to be estimated when a change in link connectivity occurs at the left boundary point of the sampling window. Otherwise, the sampling window continues to expand. By this method, the computational time of the estimation process can be saved. The resulting simplified estimation process is listed in Algorithm 2. The complexity of the simplified estimation process is  $O(L)$ . The estimation results of these two schemes are the same.

---

Algorithm 2: Pseudo code of simplified estimation  
process

---

```


$[p_{001st}, p_{002st}] \leftarrow [0,1], [p_{111st}, p_{112st}] \leftarrow [0,1];$



for  $L = 1, 2, \dots, L_{max}$  do



if  $(c(t_0 - L\Delta t) = c(t_0 - (L - 1) \Delta t))$  continue;



{Estimate  $\hat{p}_{00}$  and  $\hat{p}_{11}$  according to (12)};



{Calculate  $p_{001}, p_{002}, p_{111}$  and  $p_{112}$  from (24)};



$[p_{001st}, p_{002st}] \leftarrow [p_{001st}, p_{002st}] \cap [p_{001}, p_{002}]$ ,



$[p_{111st}, p_{112st}] \leftarrow [p_{111st}, p_{112st}] \cap [p_{111}, p_{112}]$ ;



if  $((\hat{p}_{00} \notin [p_{001st}, p_{002st}] \text{ and } N_0(L) > N_{min})$



$\text{or } (\hat{p}_{11} \notin [p_{111st}, p_{112st}] \text{ and } N_1(L) > N_{min}))$



break;



$p_{00st} \leftarrow \hat{p}_{00}, p_{11st} \leftarrow \hat{p}_{11}$ ;



$\hat{p}_{00} \leftarrow p_{00st}, \hat{p}_{11} \leftarrow p_{11st}$ ;



{Estimate  $\hat{\lambda}$  and  $\hat{\mu}$  according to (20)};



{Evaluate the stability according to (5) and (6)};


```

---

### 3.4 Routing Selection Strategy

In this section, we first propose a method to classify the nodes based on the estimated link stability. Then we present a routing selection strategy with link stability estimation. A link can be considered as stable if the link stability estimation value is higher than a upper bound value  $\alpha_3$ , and the link can be considered as unstable if the link stability estimation value is below a lower bound  $\alpha_4$ . Otherwise, we consider the link is relatively stable.

In order to make full use of the advantages of different routing protocols under different link stability states, we propose a routing combination protocol containing different protocols, i.e. OLSR, AODV and ZRP. Our Routing Combination Protocol under Link Estimation (RCPLE) routing algorithm is carried out as follows: (1) we firstly conduct link stability estimation according to Section 3.1, and classified the nodes into fixed, relatively fixed and mobility nodes; (2) we adopt different sampling windows based on the estimated link stability according to Section 3.2; (3) we switch over different routing protocols dynamically according to the estimated link states; for example, if the link is

considered stable, we adopt the OLSR protocol; we choose the AODV protocol when the link is considered unstable, and we employ the ZRP protocol in other situations.

## 4 Simulation Results

The performance of the proposed estimation scheme is evaluated using OPNET simulations. A  $1000 \times 1000 \text{ m}^2$  rectangular network area is considered. The transmission range of the nodes in the network is 250 m, and the data rate is 11 Mbps. In the simulations, we take  $\alpha_2 = 0.02$ ,  $\Delta t = 3 \text{ s}$  and  $N_{min} = 100$ . The values of  $\alpha_2$  and  $N_{min}$  determine the length of homogeneous interval, if  $\alpha_2$  is too large or  $N_{min}$  is too small, the homogeneous interval will be very small, which results in inaccuracy of the estimation. On the contrary, if the homogeneous interval is too large, it is hard to track the real-time variations of the link. The value of  $\Delta t$  should be set small enough to guarantee the link break and reconnect (or vice versa) with a negligible probability, however  $\Delta t$  should not be too small, because the estimated value is the most accurate when  $\tau = \Delta t$ .

### 4.1 Performance of the Link Stability Estimation Scheme

We first evaluate the performance of the link stability estimation scheme by placing only two nodes in the network area, where one node is placed at the center of the network area and the other node moves according to a specific mobility model. This is a simplification of multiple-node networks. Each simulation was run with 20 different random seeds to evaluate the overall performance. The first mobility model we use is the random walk model with the epochs (i.e. the time the velocity remains unchanged) being exponentially distributed with mean 20 s, as suggested in [10]. We consider the case in non-stationary networks where the moving node moves at a fixed speed within a specific time interval, but changes its speed among different intervals. The simulation length is set to 50000 s. In each interval of 10000 s, the moving node moves at a fixed speed. The set of speed in different time intervals is {10, 30, 5, 20, 40} m/s. The average values of the estimated transition rates are compared with their actual values which are computed from the reciprocal mean connected and unconnected time of the link. It can be observed from Fig. 2 that the estimated values follow the actual values despite of some slight deviation. One reason for this deviation is due to the assumption that  $p_{e0}$  and  $p_{e1}$  are small.

Another reason is that the random process describing the link connectivity has the Markov property, is not exactly true in real scenarios. Hence, the transition rates computed are not exactly equal to the values estimated by (20).

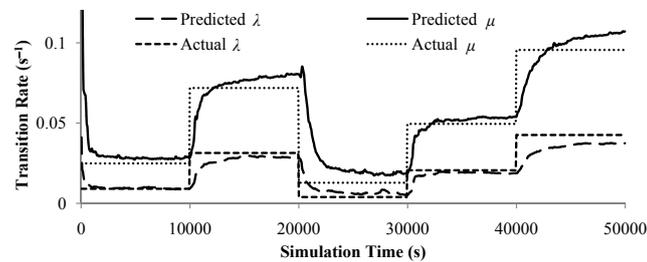


Fig. 2. Transition rates.

To further study the selection of the sampling window, we examine the sampling window length  $L\Delta t$  at different simulation time. Fig. 3 shows that after a change in speed, the sampling window length greatly decreases, while it generally increases with time when the speed remains constant. This tendency is in our expectation as discussed in Section 3.2. The reason is that the link connectivity may change frequently when the distance between the two nodes fluctuates around the transmission range; and when the distance is significantly smaller or larger than the transmission range, the link may remain connected or unconnected for a long time since the speed is low. This may lead the link stability estimation scheme to decrease the sampling window length in order to cover the most recent trend.

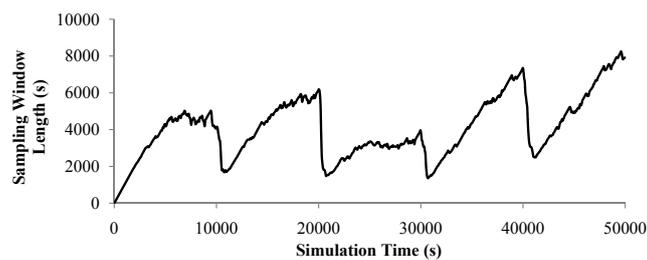


Fig. 3. Sampling window length.

Another important issue we have to study is whether the estimated remaining probability  $p_{remain}(\tau)$  and recovering probability  $p_{recover}(\tau)$  are equal to their actual values. We consider a stationary network and set the simulation length to 10000 s. The estimated probabilities are compared with the actual

frequency the link remains connected for time  $\tau$  or recovers within time  $\tau$  during simulation. Fig. 4 (a) shows the results when the moving node moves at a fixed speed  $v = 10$  m/s, Fig. 4 (b) shows the results when  $v$  is uniformly distributed between 10 m/s and 19 m/s, and in Fig. 4 (c)  $v = 20$  m/s. We also consider the scenario where two nodes move together in the network area, one at speed  $v_1 = 10$  m/s and the other at  $v_2 = 20$  m/s as shown in Fig. 5. Then we use the random waypoint model as the mobility model for the moving node and examine the performance with different pause time, as shown in Fig. 6. It can be observed that all the estimation results are approximately equal to the actual situations. Deviations are also due to  $p_{e0}$  and  $p_{e1}$  as discussed above.

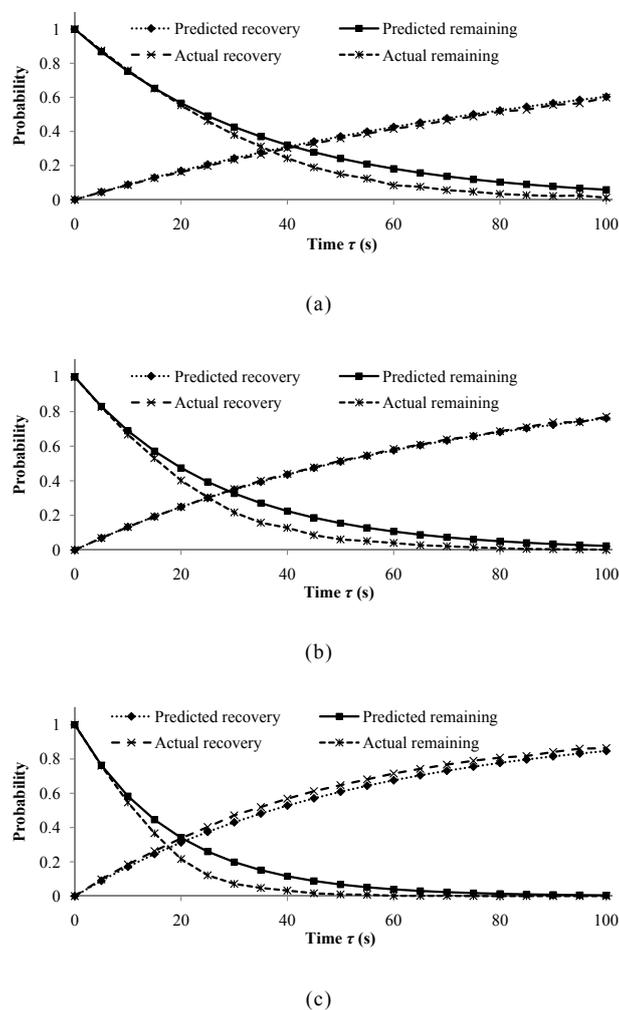


Fig. 4. Remaining and recovering probabilities with random walk model, one node moving: (a)  $v = 10$  m/s, (b)  $v \in [10, 19]$  m/s, (c)  $v = 20$  m/s.

The above results are obtained with the random walk or random waypoint model. The next question is whether the proposed scheme can be generalized to other mobility patterns, especially when the nodal mobility is not random. We still place one node fixed at the center. But we set the moving node to move in a “X” formed trajectory. In this scenario, the actual link connectivity is a periodic function of time and is no longer random. Fig. 6 shows the results for this scenario when the moving node moves at speed  $v = 10$  m/s and  $v = 20$  m/s respectively. The deviations are large at some values of  $\tau$ , since the Markov process does not hold in this scenario. However, the estimated values are relatively correct when  $\tau$  is small. This makes sense in real applications since, on one hand, the flow duration of applications is generally short; on the other hand, the exact remaining probability is not quite important when the probability is lower than a certain threshold, because we consider the link as unstable anyway. And remaining probabilities are more significant in link stability evaluation.

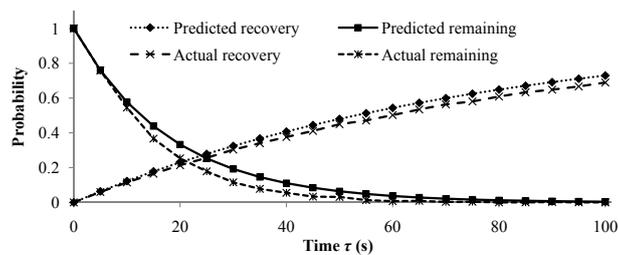
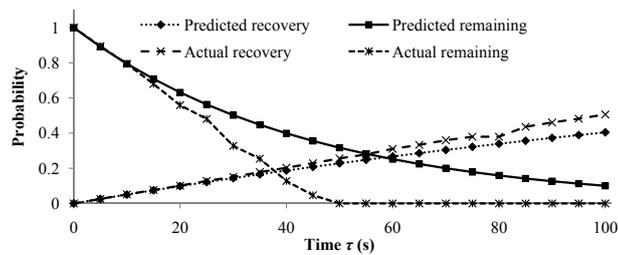


Fig. 5. Remaining and recovering probabilities with random walk model, two nodes moving,  $v_1 = 10$  m/s,  $v_2 = 20$  m/s.



(a)

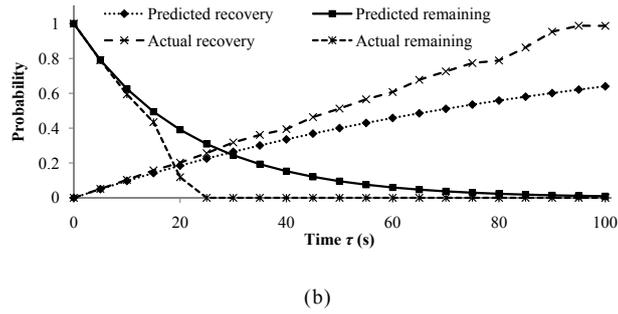


Fig. 6. Remaining and recovering probabilities with non-random movement, one node moving: (a)  $v = 10$  m/s, (b)  $v = 20$  m/s.

#### 4.2 Performance of the Routing Selection Scheme

To evaluate the performance of the routing selection scheme, we place 40 nodes in the network area. We divide the nodes into two groups. Group I contains 20 nodes. These 20 nodes are selected as fixed nodes, and formed a  $4 \times 5$  grid with 175 m separation between neighboring nodes. The nodes from Group I represent a fixed backbone network and remain in their locations throughout the whole simulation. Group II contains the other 20 nodes. These are initially placed randomly in the network area. They move according to the random walk model at an average speed which is set to  $\{0.5, 1, 2, 5, 10, 20, 40\}$  m/s respectively in these routing protocols. Each simulation was run with 100 different random seeds to obtain the overall performance. Two nodes (one from Group I and one from Group II) are selected to generate packets and send them to random destinations, at a packet rate 0.5 pk/s. The length of each packet is 1024 bit.

We first compare the packet delivery rate among three different routing protocols, namely, OLSR, AODV and ZRP protocols, which are representations of proactive, reactive and hybrid routing protocols respectively. Packet delivery rate (or success rate) is defined as the ratio of the number of packets received to the number of packets sent [7]. It can be observed from Fig. 7 that performances of the OLSR protocol are the best when nodes are static, however, as the node speed increases, the packet transmission rate decreases. This is because OLSR is a proactive routing protocol, and each node constructs and maintains routing information of all the other nodes no matter whether there exists communication demands or not. For the AODV protocol, when the nodes are static, the performance is not better than the OLSR protocol, when the speed increases to 15m/sec, the packet transmission rate

reaches the maximum value. That is because AODV transmits data according to whether there are demands or not, and the routing table is construct and maintained on demand. For the ZRP protocol, we can draw the conclusion that when the nodes are slight mobile, the packet transmission rate achieves the max value. This is because it combines proactive and reactive protocols in an organic way and adopt different routing protocols in different scenarios. As the speed extends to some extent, the delivery rates of these three single protocols descend quickly due to the highly varying topology of the 40-node network.

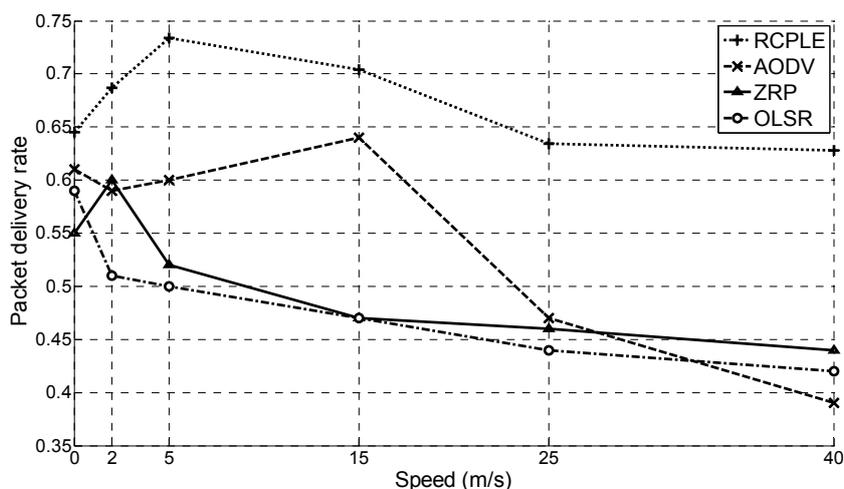


Fig. 7. Packet delivery rate for different random walk speeds.

Since our proposed routing strategy RCPLE is a combination of these three different routing protocols and one of them is selected according to the estimated link stability, the choice of the upper and lower bounds affects the performance of the network. We have run simulations to compare the performance of the network for different upper and lower bounds, and found that for the network we simulated, good packet delivery performance is provided when the upper bound  $\alpha_3$  equals to 0.7 and the lower bound  $\alpha_4$  equals to 0.4 . Although ZRP is a hybrid routing protocol combining proactive and reactive routing protocols, it only considers regional radius (measured by the hop number) and neglects the link connection or stability within each hop. Our proposed routing algorithm selects paths according to the estimated link stability and changes the sampling window size according to the packet

delivery rate, which is more dynamic than the ZRP protocol. Therefore, the proposed protocol outperforms ZRP as well.

## 5 Conclusions

The link stability estimation scheme we proposed in this paper is based on link connectivity changes and is easy to implement. Compared with link *associativity* based estimation schemes, the proposed scheme focus on a probabilistic model and the estimation results have explicit meanings both in theory and practice. Meanwhile, the proposed scheme is simpler than the methods using GPS or low layer measurements, and it is not restricted to a specific network topology. We adopted a variable sized sampling window which is more flexible for the dynamic link state, and is also a major contribution in our work.

Since link state has a pervasive impact on routing process, we proposed a routing method which adjusts its operating mode based on the estimated link stability. Simulation results show that the proposed stability estimation scheme is able to estimate the link stability in both stationary and non-stationary scenarios and the proposed routing method enhances packet delivery rate effectively in ad-hoc networks. Future work can be focused on reducing the fluctuation of the sampling window length.

## References

- [1] S. Wang, Q. Song, J. Feng, and X. Wang, "Predicting the link stability based on link connectivity changes in mobile ad hoc networks," in Proc. IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS), pp.409-414, 2010.
- [2] E. Alotaibi and B. Mukherjee, "A survey on routing algorithms for wireless ad-Hoc and mesh networks", Computer Networks, vol. 56, no.2, pp. 940–965, 2012.
- [3] L. Guo, J. Cao, H Yu, L. Li, "Path-based routing provisioning with mixed shared protection in WDM mesh networks", Journal of Lightwave Technology, vol. 24, no. 3, pp. 1129-1141, 2006.
- [4] L. Guo. "LSSP: A novel local segment-shared protection for multi-domain optical mesh networks", Computer Communications, vol. 30, no. 8, pp. 1794-1801, 2007.

- [5] P. Lopez, R. Tinedo, and J. Alsina, "Moving routing protocols to the user space in MANET middleware", *Journal of Network and Computer Applications*, vol. 33, no. 5, pp. 588-602, 2010.
- [6]. N. Sarma and S. Nandi, "Route stability based QoS routing in mobile ad-hoc networks", *Wireless Personal Communications*, vol. 54, no.1, pp. 203–224, 2010.
- [7] A. Bamis, A. Boukerche, I. Chatzigiannakis and S. Nikolettseas, "A mobility aware protocol synthesis for efficient routing in ad-hoc mobile networks", *Computer Networks*, vol. 52, no.1, pp. 130–154, 2008.
- [8] D. Floriano, F. Guerriero, and P. Fazio, "Link-stability and energy aware routing protocol in distributed wireless networks", *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 4, pp. 713-726, 2012.
- [9] A. Giovanidis and S. Stanczak, "Stability and Distributed Power Control in MANETs with Per Hop Retransmissions", *IEEE Transactions on Communications*, vol. 59, no. 6, pp. 1632-1643, 2011.
- [10] S. Jiang, D. He and J. Rao, "A prediction-based link availability estimation for routing metrics in MANETs", *IEEE/ACM Trans. on Networking*, vol. 13, no. 6, pp. 1302–1312, 2005.
- [11] J. Torkestani and M. Meybodi, "A link stability-based multicast routing protocol for wireless mobile ad-hoc networks", *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1429-1440, 2011.
- [12]. R. Biradar, S. Manvi and M. Reddy, "Link stability based multicast routing scheme in MANET", *Computer Networks*, vol. 54, no. 7, pp. 1183-1196, 2010.
- [13] K. Farkas, T. Hossmann, F. Legendre, B. Plattner and S. Das, "Link quality prediction in mesh networks", *Computer Communications*, vol. 31, no. 8, pp. 1497–1512, 2008.
- [14] R.Vadivel and V. Bhaskaran, "Adaptive reliable routing protocol using combined link stability estimation for mobile ad-hoc networks", in *Proc. AIP Conference*, pp. 625-632, 2010.

- [15] T. Anagnostopoulos, C. Anagnostopoulos and S. Hadjiefthymiades, “An adaptive location prediction model based on fuzzy control”, *Computer Communications*, vol. 34, no. 7, pp. 816–834, 2011.
- [16] R. Huang and G. Záruba, “Location tracking in mobile ad-hoc networks using particle filters”, *Journal of Discrete Algorithms*, vol. 5, no. 3, pp. 455-470, 2007.
- [17] N. Weng and J. Yang, “A cross-layer stability-based routing mechanism for ultra wideband networks”, *Computer Communication*, vol. 33, no. 18, pp. 2185-2194, 2010.
- [18] R. Biradar and S. Manvi, “Neighbor supported reliable multipath multicast routing in MANETs”, *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 1074-1085, 2012.
- [19]. Z. Ning, L. Guo, Y. Peng, and X. Wang, “Joint scheduling and routing algorithm with load balancing in wireless mesh networks”, *Computers and Electrical Engineering*, vol. 38, no. 3, pp. 533-550, 2012.
- [20] L. Guo, L. Zhang, Y. Peng, J. Wu, X. Zhang, W. Hou and J. Zhao, “Multi-path routing in spatial wireless ad hoc networks”, *Computers and Electrical Engineering*, vol. 38, no. 3, pp. 473–491, 2012.
- [21] J. Wang, Y. Liu and J. Yu, “Building a trusted route in a mobile ad hoc network considering communication reliability and path length”, *Journal of Network and Computer Applications*, vol. 34, no. 4, pp. 1138-1149, 2011.
- [22] W. Yang, X. Yang, S. Yang, and D. Yang, “A greedy-based stable multi-path routing protocol in mobile ad hoc networks”, vol. 9, no. 4, pp. 662-674, 2011.
- [23] Y. Zhang, J. Luo and H. Hu, “Wireless mesh networking: architectures, protocols and standards”, Auerbach Publications, 2007.
- [24] M. Kharraz, S. Hamid, and Z. Albert, “On-demand multicast routing protocol with efficient route discovery”, *Journal of Network and Computer Applications*, vol. 35, no. 3, pp. 942-950, 2012.
- [25] F. Li and Y. Wang, “Routing in vehicular ad hoc networks: A survey”, *IEEE Vehicular Technology Magazine*, vol. 2, no. 2, pp. 12-22, 2007.

- [26] L. Guo, Y. Peng, X. Wang, D. Jiang and Y. Yu, "Performance evaluation for on-demand routing protocols based on OPNET modules in wireless mesh networks", *Computers and Electrical Engineering*, vol. 37, no. 1, pp. 106-114, 2011.
- [27] S. Hwang and D. Kim, "Markov model of link connectivity in mobile ad-hoc networks", *Telecommunication Systems*, vol. 34, no. 1, pp. 51-58, 2007.
- [28] Z. Ye and A. Abouzeid, "Optimal stochastic location updates in mobile ad hoc networks", *IEEE Transactions on Mobile Computing*, vol. 10, no. 5, pp. 638-652, 2011.
- [29] A. Papoulis and S. Pillai, "Probability, Random Variables and Stochastic Processes", New York: McGraw-Hill Companies, Inc., 2002.
- [30] G. Casella and R. L. Berger, "Statistical Interference, 2nd Edition", Singapore: Thomson Learning Asia, 2002.